QUALCOMM®
CDMA Technologies

MSM6550/MSM6150™ Mobile Station Modem

Software Interface

80-V7196-2 Rev. A

June 24, 2004

QUALCOMM® Proprietary

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

Send technical questions to:
https://support.cdmatech.com

MSM6550 / MSM6150™ Software Interface

80-V7196-2 Rev. A

Contents

11 Receive Front-End

14 Deinterleaver

17 Modulator

18 Sleep Controller

Registers by Name

Figures

Tables

1 Introduction

1.1 Overview

This book details the MSM6550 device ARM-accessible registers. This chapter provides a register overview, along with the definitions and assumptions used. The remaining chapters are organized by core and provide detailed register information.

Each core's chapter includes one or two major sections listing the core's ARM registers, as well as the address for each register, its bit assignments and bit settings. The chapter also provides an overview and detailed information on the functionality of each register.

NOTE **Addresses in this document are offsets from zero for each core**, unlike register addresses in previous MSM documentation, which have typically included actual BASE+offsets sequentially throughout the book.

1.2 Definitions/guidelines

Table 1-1 offers guidance in interpreting the register definitions and covers some of the issues and assumptions made throughout this book.

Table 1-1 Register definitions

	Guidelines
Endian byte order	Endian is a term that defines the byte order to store a sequence of bytes in memory, either MSB first (big-endian) or LSB first (little endian). Hardware coding defines MSM6500's registers; in each 32-bit word, MSB = 31, LSB = 0. The MSM6500 chip supports only little endian addressing. In little endian mode, the first byte is bits 7:0, and the fourth byte is bits 31:24.
Word addressing	All registers are word-accessible. Assume the ARM convention: a byte, 8 bits; a half-word, 16 bits; a word, 32 bits. All accesses are expected to be word size. The ARM can do byte and half-word reads, since hardware treats a read as a word access. Individual bytes accesses to a read register that triggers a hardware event will trigger an event for each read performed.
Active bit state	Unless stated otherwise, all boolean flags are active-high in register descriptions. Use a value of "1" to enable a function or indicate an event and a value of "0" do the opposite. Active-low flags are explicitly specified in the register descriptions. For example, if the BOOGA_ENA bit is set '1,' it enables BOOGA mode; if cleared '0,' it disables the mode.
Unused bits/ words	In the register map many bits are marked "RESERVED" or unused. Ignore undefined bits in readable registers, software cannot assume a '0' upon a read. Write undefined bits with '0's' in writable registers. Bits marked "IGNORED" can be written with any desired value; in some cases this is convenient to software. MSM6500's memory and register addressing simplifies implementation and allows expansion. Many holes exist in the address space. Avoid accesses to undefined memory locations, they can cause unpredictable behavior.
Reset state	A register's reset state is 'unknown' unless stated otherwise. It's better not to use a reset in hardware, it conserves area and simplifies coding and testing. Bits with a reset will be noted in the register descriptions. Assume that when a core comes out of reset, all its programmed register bits are either in their reset state (where defined) or unknown. Do not expect a previously programmed value to be maintained.

1.3 Register categories

Most registers can be categorized using one of the types listed in Table 1-2. These categories are defined here to simplify the software interface.

Table 1-2 Register categories

	Guidelines
Write command	Command registers are always write-only, typically there is no meaningful read-back value. Use command registers to trigger events in the hardware. Often these registers serve a single purpose. You write a bit pattern to specify the desired event. Multiple events can be combined; a register may have multiple 1-bit fields, each field triggering a hardware action, such as in an interrupt clear register. A command register can also contain encoded command fields and other data fields. The bit pattern written to it can be transient, triggering an event and then gone. No need to first write a value to trigger the event and another value to stop the trigger.
Read command	These registers return the desired data, but may also trigger a hardware event every time they are read. Often they are used to read out memory locations where each read triggers a read of the next memory location.
Control	Control registers configure the hardware to operate in a specific mode. These registers are either static or take effect at a specific time boundary (such as the next 256 chip boundary). Control bits are registered in hardware and may be readable.
Status	Status registers are read-only and reflect the current state of the hardware. There may be restrictions on when the status words can be reliably read, this ensures a self-consistent value is available. Any restrictions will be indicated in the register descriptions.
State	State registers are read/writable by the processor but also change state due to hardware events. For software to reliably read or update a state register, special “rules of engagement” may be needed to read out a self-consistent value or avoid collisions for updates by software and hardware. Each state register has these rules described in detail.
Abbreviations	
W	write register
R	read register
C	command register
X	unknown after power-on

1.4 ARM address map summary

Figure 1-1 shows the global memory address map for MSM6550/6150 from the ARM's point of view.

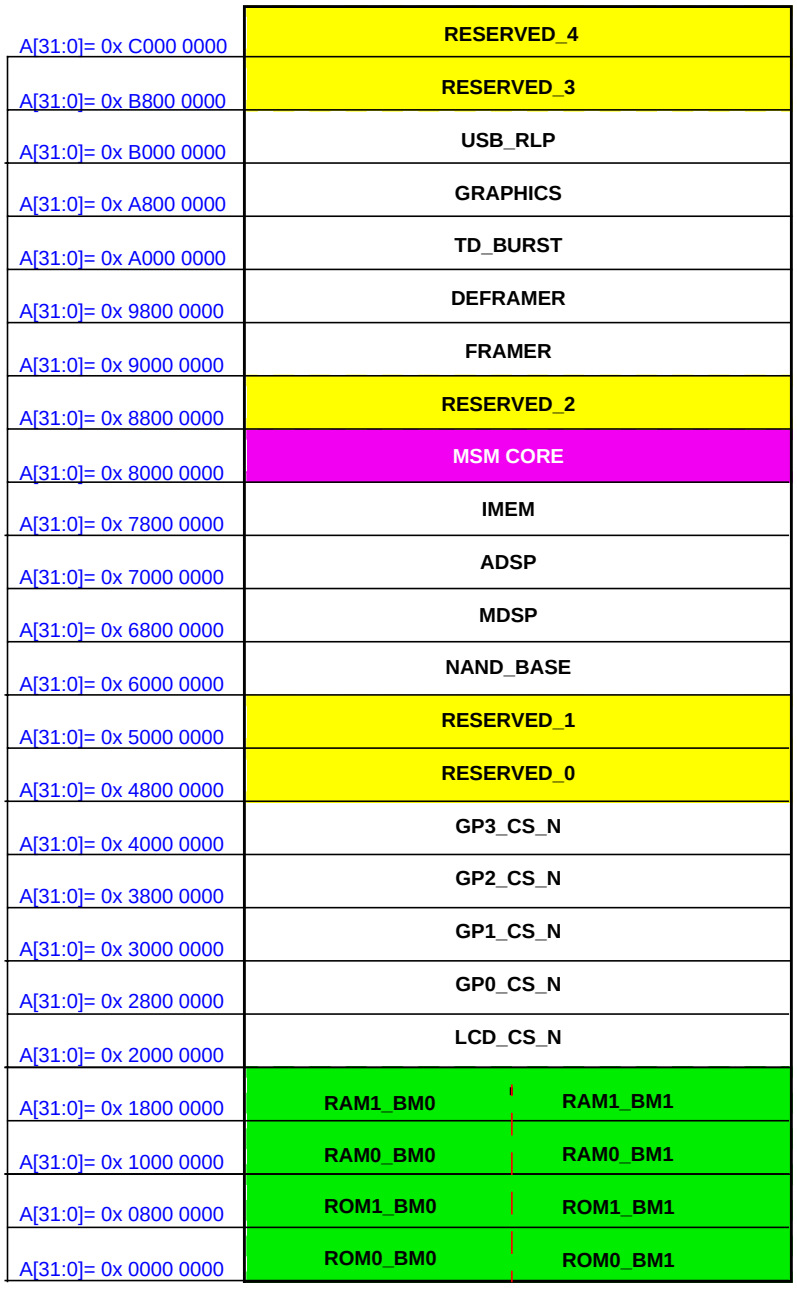
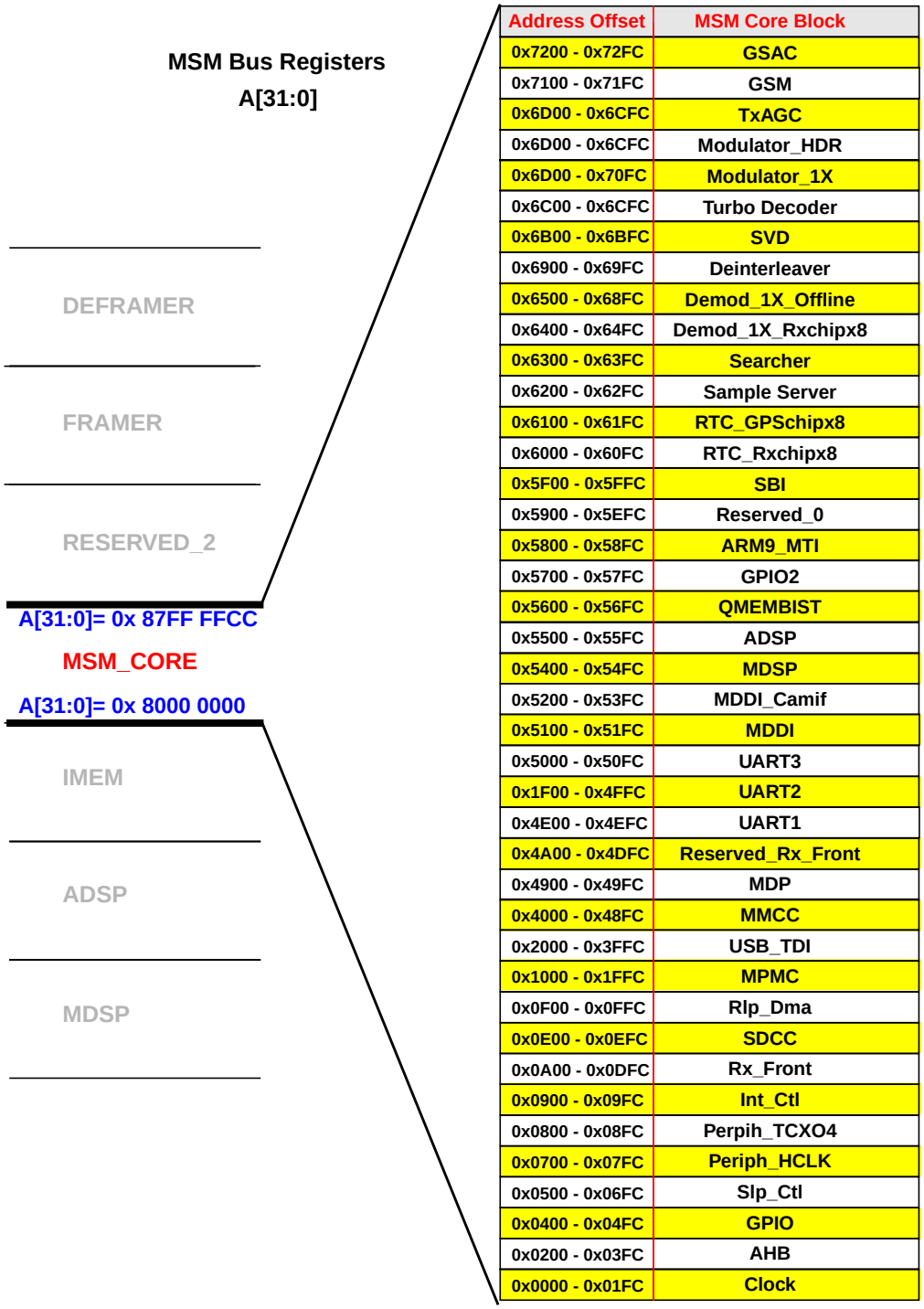


Figure 1-1 MSM6550 memory map

1.4.1 MSM_CORE region

Figure 1-2 shows how the MSM_CORE region of Figure 1-1 is divided into offset addresses for accessing MSM registers in MSM6550/6150 cores, using the MSM_bus micro interface.



Address Offset	MSM Core Block
0x7200 - 0x72FC	GSAC
0x7100 - 0x71FC	GSM
0x6D00 - 0x6CFC	TxAGC
0x6D00 - 0x6CFC	Modulator_HDR
0x6D00 - 0x70FC	Modulator_1X
0x6C00 - 0x6CFC	Turbo Decoder
0x6B00 - 0x6BFC	SVD
0x6900 - 0x69FC	Deinterleaver
0x6500 - 0x68FC	Demod_1X_Offline
0x6400 - 0x64FC	Demod_1X_Rxchipx8
0x6300 - 0x63FC	Searcher
0x6200 - 0x62FC	Sample Server
0x6100 - 0x61FC	RTC_GPSchipx8
0x6000 - 0x60FC	RTC_Rxchipx8
0x5F00 - 0x5FFC	SBI
0x5900 - 0x5EFC	Reserved_0
0x5800 - 0x58FC	ARM9_MTI
0x5700 - 0x57FC	GPIO2
0x5600 - 0x56FC	QMEMBIST
0x5500 - 0x55FC	ADSP
0x5400 - 0x54FC	MDSP
0x5200 - 0x53FC	MDDI_Camif
0x5100 - 0x51FC	MDDI
0x5000 - 0x50FC	UART3
0x1F00 - 0x4FFC	UART2
0x4E00 - 0x4EFC	UART1
0x4A00 - 0x4DFC	Reserved_Rx_Front
0x4900 - 0x49FC	MDP
0x4000 - 0x48FC	MMCC
0x2000 - 0x3FFC	USB_TDI
0x1000 - 0x1FFC	MPMC
0x0F00 - 0x0FFC	Rlp_Dma
0x0E00 - 0x0EFC	SDCC
0x0A00 - 0x0DFC	Rx_Front
0x0900 - 0x09FC	Int_Ctl
0x0800 - 0x08FC	Perpih_TCXO4
0x0700 - 0x07FC	Periph_HCLK
0x0500 - 0x06FC	Slp_Ctl
0x0400 - 0x04FC	GPIO
0x0200 - 0x03FC	AHB
0x0000 - 0x01FC	Clock

Figure 1-2 ARM-accessible MSM hardware register regions

1.5 Register definitions

In the register descriptions, the addresses shown are from the perspective of the ARM.

All non-command writable registers may have a reset condition as well as a time when the write takes effect. By default, writable registers do not have a reset condition, and a value written takes effect immediately. Registers not following the default behavior use the notation below:

Reset State: The value this register contains following a reset.

Each register also specifies the clock regime it is on:

Clock: <main regime>_<local regime>.

This information can be used by software to determine the main and/or local regime that a register is part of. Reads and writes to registers where the clock regime is disabled are not supported. The chip will not hang-up, but there is no guarantee the access will take place. Software can find where this may be an issue by enabling aborts for these types of accesses.

2 Clocks

2.1 ARM registers

Clock selection, programming, and gating are under microprocessor control. This section lists the microprocessor registers associated with the clock block in MSM6550/6150.

Addresses are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: CLK
- Number of word addresses: 128
- Byte Address: 0x0000 - 0x01FC
- MODULE OFFSET = CHIP_BASE+0x000
- MAX=CHIP_BASE+0x01FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x0000 MSM_CLK_HALTA

Type: Write/Read
Clock: HCLK
Reset State: 0xFFFFFFFF

Clock Regime Management Register.

All bits of this register are set (1) on RESOUT. However, all the clock regimes are on at the TCXO frequency during RESOUT due to the nature of CSC. The halt signals won't take effect until the RESOUT is de-asserted. Setting (1) this register has the effect of disabling all clock regimes.

Set (1) a bit to disable the corresponding clock regime. Clear (0) a bit to enable the corresponding clock regime.

MSM_CLK_HALTA

Bits	Name	Description
31	HDR_DEMOD_CLK_HALT	Asynchronous halt for the HDR demodulator clock regime.
30	SVD_CLK_HALT	Asynchronous halt for the svd clock regime.
29	TURBO_DEC_CLK_HALT	Asynchronous halt for the turbo decoder clock regime.
28	GSM_GSAC_CLK_HALT	Asynchronous halt for the GSM GSAC clock regime.
27	GSM_CLK_HALT	Asynchronous halt for the GSM clock regime.
26	PDM_CLK_HALT	Asynchronous halt for the PDM clock regime.
25	UART3_CLK_HALT	Asynchronous halt for the UART3 clock regime.
24	I2C_CLK_HALT	Asynchronous halt for the I2C clock regime.
23	ETM_CLK_HALT	Asynchronous halt for the ETM clock regime.
22	UART2_CLK_HALT	Asynchronous halt for the UART2 clock regime.
21	HDR_TDEC_CLK_HALT	Asynchronous halt for the HDR decoder clock regime.
20	CHAIN1_RXFRONT_CLK_HALT	Asynchronous halt for the chain 1 receive front end clock regime.
19	CHAIN0_RXFRONT_CLK_HALT	Asynchronous halt for the chain 0 receive front end clock regime.
18	MMCDIV_CLK_HALT	Asynchronous halt for the Multimedia Card clock regime.
17	SDAC_CLK_HALT	Asynchronous halt for the External Stereo DAC clock regime.
16	BT_CLK_HALT	Asynchronous halt for the Bluetooth clock regime.
15	BT_SLP_CLK_HALT	Asynchronous halt for the Bluetooth slpctl clock regime.
14	USB_MCLK_HALT	Asynchronous halt for the Universal Serial Bus clock regime.
13	USB_SCLK_HALT	Asynchronous halt for the Universal Serial Bus clock regime.
12	ICODEC_CLK_HALT	Asynchronous halt for the Internal Codec clock regime.
11	OFFLINE_CLK_HALT	Asynchronous halt for the offline engine clock regimes.
10	DEC_CLK_HALT	Asynchronous halt for the Deinterleaver clock regimes.
9	MDSP_CLK_HALT	Asynchronous halt for the Demodulator DSP clock regime.
8	SBI_CLK_HALT	Asynchronous halt for the Serial Bus Interface clock regime.
7	RESERVED_BIT7	
6	ADSP_CLK_HALT	Asynchronous halt for the Application DSP clock regime.

MSM_CLK_HALTA (Continued)

Bits	Name	Description
5	ECODECIF_CLK_HALT	Asynchronous halt for the External Codec Interface clock regime.
4	UNUSED_BIT4	
3	RXCHIPX8_CLK_HALT	Asynchronous halt for the CDMA Receive clock regime.
2	TXCHIPX8_CLK_HALT	Asynchronous halt for the CDMA Transmit clock regime.
1	UART1_CLK_HALT	Asynchronous halt for the UART1 clock regime.
0	GENERAL_CLK_HALT	Asynchronous halt for the General Tcxo/4 clock regime.

NOTE The clock source must be oscillating for this respective bit to take effect.

To disable a clock regime:

- Make sure that the external clock source is oscillating.
- Set the corresponding bit in MSM_CLK_HALTA.

To enable a disabled clock regime:

- Make sure that the external clock source is oscillating.
- Clear the corresponding bit in MSM_CLK_HALTA.

0x0004 MSM_CLK_HALTB

Type: Write/Read
Clock: HCLK
Reset State: 0x1FFFF

Clock Regime Management Register.

All bits of this register are set on RESOUT However, all the clock regimes are on at the TCXO frequency during RESOUT due to the nature of CSC. The halt signals won't take effect until the RESOUT is asserted. Set this register has the effect of disabling all clock regimes.

Set (1) a bit to disable the corresponding clock regime. Clear a bit to enable the corresponding clock regime.

MSM_CLK_HALTB

Bits	Name	Description
27	QMEMBIST_CLK_HALT	Asynchronous halt for the Qmembist IO regime.
26	MDDI_CLIENT_IO_CLK_HALT	Asynchronous halt for the MDDI client IO regime.
25	MDDI_CLIENT_DIV4_CLK_HALT	Asynchronous halt for the MDDI client div4 regime.
24	MDDI_CLIENT_VFE_CLK_HALT	Asynchronous halt for the MDDI client VFE regime.
23	MDDI_HOST_HCLK_HALT	Asynchronous halt for the MDDI host HCLK regime.
22	MDDI_HOST_CLK_HALT	Asynchronous halt for the MDDI host clock regime.
21	MDP_HCLK_HALT	Asynchronous halt for the MDP HCLK regime.
20	SDCC_MCLK_HALT	Asynchronous halt for the SDCC MCLK regime.
19	SDCC_HCLK_HALT	Asynchronous halt for the SDCC HCLK regime.
18	MPMC_FB_CLK_HALT	Asynchronous halt for the MPMC feedback clock regime.
17	GRAPHICS_HCLK_HALT	Asynchronous halt for the graphics HCLK regime.
16	USB_OTG_HCLK_HALT	Asynchronous halt for the USB OTG HCLK regime.
15	RESERVED_BIT15	
14	HDR_PDM_CLK_HALT	Asynchronous halt for the HDR modulator pdm clk regime.
13	SRCHCHIPX8_CLK_HALT	Asynchronous halt for the 8x-searcher clock regime.
12	RTC_GSPCHIPX8_CLK_HALT	Asynchronous halt for the GPS RTC clock regime.
11	RTC_HDRCHIPX8_CLK_HALT	Asynchronous halt for the HDR RTC clock regime.
10	VIDEO_HCLK_HALT	Asynchronous halt for the video HCLK regime.
9	VIDEO_VFE_CLK_HALT	Asynchronous halt for the video VFE clock regime.
8	MPMC_CLK_HALT	Asynchronous halt for the MPMC clock regime.

MSM_CLK_HALTB (Continued)

Bits	Name	Description
7	CAMCLK_PO_HALT	Asynchronous halt for the CAMCLK regime.
6	SLPFAST_FEE_CLK_HALT	Asynchronous halt for the sleep controller FEE clock regime.
5	SLPFAST_HDR_CLK_HALT	Asynchronous halt for the HDR sleep controller clock regime.
4	SLPFAST_GSM_CLK_HALT	Asynchronous halt for the GSM sleep controller clock regime.
3	SLPFAST_1X_CLK_HALT	Asynchronous halt for the 1X sleep controller clock regime.
2	SLPCTL_HDR_CLK_HALT	Asynchronous halt for the HDR sleep controller clock regime.
1	SLPCTL_GSM_CLK_HALT	Asynchronous halt for the GSM sleep controller clock regime.
0	SLPCTL_1X_CLK_HALT	Asynchronous halt for the 1X sleep controller clock regime.

0x0008 Reserved.

0x000C Reserved.

0x0010 DSP_DMA_CLK_OVR_REG

Type: Write/Read
Clock: HCLK
Reset State: 0xF

The clock gating override bits of DSP_DMA_CLK_OVR_REG are added to reduce the risk due to the addition of clock gating and the respective auto-sleep/wake-up circuits.

DSP_DMA_CLK_OVR_REG

Bits	Name	Description
3	ADSPCLK_1X_MODE	When set (1) the ADSP clock is running at 1x. When clear (0) the ADSP clock is running at div2. This value may only be changed when the ADSP clock is disabled and the ADSP is in reset.
2	MDSPCLK_1X_MODE	When set (1) the MDSP clock is running at 1x. When clear (0) the MDSP clock is running at div2. This value may only be changed when the MDSP clock is disabled and the MDSP is in reset.

DSP_DMA_CLK_OVR_REG (Continued)

Bits	Name	Description
1	ADSP_DMA_CLK_OVR_N	When set (1) the clock is controlled by the QDSP's internal control When clear (0) the dma clock is forced on when sclk is enabled
0	MDSP_DMA_CLK_OVR_N	When set (1) the clock is controlled by the QDSP's internal control When clear (0) the dma clock is forced on when sclk is enabled

0x0014 MISC_CLK_CTL

Type: Write/Read
Clock: HCLK
Reset State: 0x000000ff

This register contains miscellaneous clock control signals.

Bit[26:8] are cleared by default, while Bit[7:0] are set by default.

MISC_CLK_CTL

Bits	Name	Description
30	PLL1_EXT_DIV	When set (1) the external divider is enabled for pll1. When clear (0) the internal divider is enabled for pll1.
29	PLL0_EXT_DIV	When set (1) the external divider is enabled for pll0. When clear (0) the internal divider is enabled for pll0.
28	HCLK_SLEEP_EN	Set(1) to enable bus clocks to be off when micro_clk is off
27	XMEM_HCLK_SLEEP	Set(1) this bit to disable the xmem_hclk.
26	CHAIN0_RX_SAMPLE_SLEEP	Set(1) this bit to disable the chain0 rx_sample_clk. It is used for test purpose when doing power measurement.
25	CHAIN1_RX_SAMPLE_SLEEP	Set(1) this bit to disable the chain1 rx_sample_clk. It is used for test purpose when doing power measurement.
24	CHAIN0_SAMPCTRL_SLEEP	Set(1) this bit to disable the chain0 sample_ctl_clk. It is used for test purpose when doing power measurement.
23	CHAIN1_SAMPCTRL_SLEEP	Set(1) this bit to disable the chain1 sample_ctl_clk. It is used for test purpose when doing power measurement.
22	CHIPX16_SEL	Clear(0) this bit to select the chipx16 from digital MN/CNTR. Set (1) to select chipx16 generated from hv pll in analog_clk_src. This bit is cleared by default.

MISC_CLK_CTL (Continued)

Bits	Name	Description
21:20	PLL_CHIPX16_SEL	These bits are used in analog_clk_src block to select cx16 source from the pll divided down signal. These bits are cleared by default PLL CHIPX16 SEL — Clock Source 00 — pll_div5 01 — pll_div10 10 — pll_div15
19	CROSSBAR_RST	Set (1) this bit to assert reset to the chipxN crossbar (it is applied to the non-GPS clock dividers). Clear(0) this bit to bring the crossbar out of reset. The crossbar supplies clocks to the receive front-end. It is designed to minimize clock skew between the two receive chains.
18	PLLOUT_DIV5_EN	Set (1) to enable the divide by 5 logic following the Pll. Clear (0) to disable the divide logic and save power. This bit is cleared by default.
17	PLLOUT_DIV2_EN	Set (1) to enable the divide by 2 logic following the Pll/5. Clear (0) to disable the divide logic and save power. This bit is cleared by default.
16	PLLOUT_DIV3_EN	Set (1) to enable the divide by 3 logic following the Pll/5. Clear (0) to disable the divide logic and save power. This bit is cleared by default.
15	RESERVED_BIT15	
14	TURBO_DEC_SLEEP_EN	Set (1) to enable synchronous sleeping of the turbo_dec_clk based on turbo_dec_sleep from the decoder. Clear (0) to disable sleep.
13	SVD_SLEEP_EN	Set (1) to enable synchronous sleeping of the svd_clk based on svd_sleep from the decoder. Clear (0) to disable sleep.
12:10	GPS_CLK_DELAY	These three bits select the appropriate delay for the GPS mncntr reset. This delay should be chosen so as to match the delay of the clock source path. DEC_CLK_SRC_SEL — Clock Source 0x0 — Minimum Delay (Default) 0x1 0x2 0x3 0x4 0x5 0x6 0x7 — Maximum Delay

MISC_CLK_CTL (Continued)

Bits	Name	Description
9	TCXO_CNT_EN	Set (1) to enable the TCXO counter used to estimate the ring oscillator frequency. Clear (0) to disable the counter. Enabling the counter will send a 1 cycle reset to clear old values. Cleared on RESOUT
8	RING_OSC_CNT_EN	Set (1) to enable the ring oscillator counter used to estimate the ring oscillator frequency. Clear (0) to disable the counter. Enabling the counter will send a 1 cycle reset to clear old values. Cleared on RESOUT
7	RESERVED_BIT7	
6	AHB_WR_CLK_EN	When set (1) the ahb_wr_clk will be a free running hclk. This is backward compatible with other MSMs. When cleared (0) the ahb_wr_clk will only toggle when a write is done to the ahb_regs address space. This is done to save power. This bit is Set on RESOUT
5	RESERVED_BIT5	
4	RESERVED_BIT4	
3	RESERVED_BIT3	
2	RESERVED_BIT2	
1	AC_DC_N	AC_DC_n mode bit for TCXO Pad AC_DC_n Description 0 DC mode 1 AC mode (Default)
0	TXDAC_SLEEP	txdac_sleep bit output to TLMM. Set (1) on RESOUT

0x0018 MSM_CLK_RINGOSC

Type: Write/Read
Clock: HCLK
Reset State: 0x00

This register contains control signals for the two ring oscillator cores.

MSM_CLK_RINGOSC

Bits	Name	Description
7:6	RING_OSC_DIV_SEL	These bits define the division ratio for the external divider on the ring oscillator div source. Changes should only be made when the source is unused. RING_OSC_DIV_SEL div clock 00 Divide by 1 (Default) 01 Divide by 2 10 Divide by 3 11 Divide by 4
5:4	RING_OSC_EN	Enable bits to both internal ring oscillator cores. RING_OSC_EN Core1 Core2 Dividers 00 Disabled Disabled Disabled (Default) 01 Enabled Disabled Enabled 10 Disabled Enabled Enabled 11 Enabled Enabled Enabled
3:0	RING_OSC_SEL	Selects the divide ratio for the internal dividers of the ring oscillator, and the desired internal Core to use as the ring_osc_div_clk. RING_OSC_SEL Divider output 00 00 Core 1 (Default) 00 01 Core 1 Divide by 2 00 10 Core 1 Divide by 4 00 11 Core 1 Divide by 8 01 00 Core 1 Divide by 16 10 00 Core 2 10 01 Core 2 Divide by 2 10 10 Core 2 Divide by 4 10 11 Core 2 Divide by 8 11 00 Core 2 Divide by 16

0x001C TCXO_CNT

Type: Read
Clock: HCLK

This register holds the value stored in the tcxo counter. This value is used to compare the ring oscillator frequency to the tcxo frequency and thereby estimating its frequency. This register is reset automatically by the rising edge of the tcxo_cnt_en.

Bits	Name	Description
19:0	TCXO_CNT	This register holds the value stored in the tcxo counter. This value is used to compare the ring oscillator frequency to the tcxo frequency and thereby estimating its frequency. This register is reset automatically by the rising edge of the tcxo_cnt_en.

0x0020 RINGOSC_CNT

Type: Read
Clock: HCLK

This register holds the value stored in the ring oscillator counter. This value is used to compare the ring oscillator frequency to the tcxo frequency and thereby estimating its frequency. This register is reset automatically by the rising edge of ringosc_cnt_en.

Bits	Name	Description
19:0	RINGOSC_CNT	This register holds the value stored in the ring oscillator counter. This value is used to compare the ring oscillator frequency to the tcxo frequency and thereby estimating its frequency. This register is reset automatically by the rising edge of ringosc_cnt_en.

0x0024 MSM_CLK_SLEEPOSC

Type: Write/Read
Clock: HCLK
Reset State: 0x0

MSM_CLK_SLEEPOSC defines the sleep oscillator controls. This register selects the clock source for the CDMA sleep controller. Only TCXO=19.2MHz is supported in MSM6550/MSM6150; therefore, tcxo_sel=”10” is hard-coded and has no register bit to program it.

Bits of this register are cleared (0) on resin_n since the watchdog component of resout should not reset the clock divide select, nor the sleep clock source select logic. This register is required to be initialized shortly coming out of system reset (resin_n). Once written, this register is disabled and cannot be configured unless a system reset occurs. This would prevent software from corrupting the hardware setup.

MSM_CLK_SLEEPOSC

Bits	Name	Description
11:10	SLEEP_OSC_GAIN	SLEEP_OSC_GAIN controls the gain inverters to drive the oscillator feedback. This control bit for the sleep crystal oscillator provides four steps of gain. **SLEEP_OSC_GAIN** — **Amount of Gain** 00 — Minimum (Default) 01 10 11 — Maximum
9	SLEEP_OSC_RF_BYPASS	This bit disables the internal resistance on the schmitt trigger path and is powered up with a value of LOW.
8	SLEEP_OSC_RD_BYPASS	This bit disables the output resistance on the SLEEP_XTAL_OUT path and is powered up with a value of Low.
7	SLEEP_XTAL_EN	This bit enable the switch from tcxo to slp_xtal rate of wdog timer clock (slpctl_fee_clk) and BT slpctl clock (bt_slp_clk).
6:0	RESERVED_BITS7_0	

0x0028 MSM_CLK_USBOSC

Type: Write/Read
Clock: HCLK
Reset State: 0x5000

The MSM_CLK_USBOSC Register controls the USB crystal oscillator. All bits are Clear at default except bit 14 and bit 12.

MSM_CLK_USBOSC

Bits	Name	Description
14	USB_OSC_EN_N	This bit disables the 48MHz Xtal when high. Default to high as normal operation. USB_OSC_EN_N — Function 0 — Enable 48MHz crystal 1 — Disable 48MHz crystal (Default)
13	USB_OSC_GAIN	USB_OSC_GAIN controls the various gain inverters to drive the oscillator feedback. USB_OSC_GAIN — Amount of Gain 0 — Minimum (Default) 1 — Maximum
12	USB_OCS_RF_BYPASS	This bit disables the internal resistance on the schmitt trigger path and is powered up with a value of HIGH.

MSM_CLK_USBOSC (Continued)

Bits	Name	Description
11	USB_OSC_RD_BYPASS	This bit disables the output resistance on the USB_XTAL_OUT path and is powered up with a value of low.
10	UNUSED_BIT10	
9	ADSP_RPCM_OFF_EN	This bit is used to grant permission to the ADSP for stopping the RPCM clock by asserting RPCM_SLEEP. This bit is Clear on resout
8:0	UNUSED_BITS8_0	

0x0030 PLL0_MODE

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register configures and controls PLL0. All register bits will default to 0 on RESOUT. The power-up default for all bit fields is low.

PLL0_MODE

Bits	Name	Description
7:4	PLL_PRE_DIV	These bits define the internal PLL pre-divide value as follows: **PLL0_PRE_DIV** — **Counter Output** 0000 — input 0001 — divide by 2 0010 — divide by 3 0011 — divide by 4 0100 — divide by 5 0101 — divide by 6 0110 — divide by 7 0111 — divide by 8 1000 — divide by 9 1001 — divide by 10 1010 — divide by 11 1011 — divide by 12 1100 — divide by 13 1101 — divide by 14 1110 — divide by 15 1111 — divide by 16
3	PLL_PLLTEST	Set(1) this bit to enter PLL test mode. Clear (0) in normal mode
2	PLL_RESET_N	Low asserted reset for the digital logic in the PLL. This includes the MND counters and integer divider.

PLL0_MODE (Continued)

1	PLL_BYPASSNL	Clear (0) to bypass the pll. PLLOUT is therefore identical to input reference. Set (1) to use the pll. Default is '0' at RESOUT. (different from previous MSM)
0	PLL_OUTCTRL	Set (1) to activate the pll's output. The Analog circuitry is active but the output is not. Clear (0) to disable the pll's output and save power. Defaults is '0' at RESOUT.

0x0034 PLL0_L_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Integer value of fractional division ratio in the fractional-N PLL. The equation used to calculate the PLL output frequency is: VCO=TCXO / P * [(L-1)+ M/N)], given L , M, and N, the value in the PLL0_L_VAL, PLL0_M_VAL, and PLL0_N_VAL registers, respectively.

Bits	Name	Description
7:0	PLL_L	This register contains the 8-bit L value used by the PLL0's fractional division ratio.

0x0038 PLL0_M_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Value of numerator of fractional value in the fractional-N PLL.

Bits	Name	Description
10:0	PLL_M	This register contains the 11-bit M value used by the PLL0's numerator value in the fractional division ratio.

0x003C PLL0_N_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Value of denominator of fractional value in the fractional-N PLL.

Bits	Name	Description
11:0	PLL_N	This register contains the 12-bit N value used by the PLL0's denominator value of the fractional division ratio.

0x004C PLL1_MODE

Type: Write/Read
Clock: HCLK

This register configures and controls PLL1. All register bits will default to 0 on RESOUT. The power-up default for all bit fields is low.

PLL1_MODE

Bits	Name	Description
7:4	PLL_PRE_DIV	These bits define the internal PLL pre-divide value as follows: PLL1_PRE_DIV Counter Output 0000 input 0001 divide by 2 0010 divide by 3 0011 divide by 4 0100 divide by 5 0101 divide by 6 0110 divide by 7 0111 divide by 8 1000 divide by 9 1001 divide by 10 1010 divide by 11 1011 divide by 12 1100 divide by 13 1101 divide by 14 1110 divide by 15 1111 divide by 16
3	PLL_PLLTEST	Set(1) this bit to enter PLL test mode. Clear (0) in normal mode
2	PLL_RESET_N	Low asserted reset for the digital logic in the PLL. This includes the MND counters and integer divider.
1	PLL_BYPASSNL	Clear (0) to bypass the pll. PLLOUT is therefore identical to input reference. Set (1) to use the pll. Default is ’0’ at RESOUT. (different from previous MSM)
0	PLL_OUTCTRL	Set (1) to activate the pll’s output. The Analog circuitry is active but the output is not. Clear (0) to disable the pll’s output and save power. Defaults is ’0’ at RESOUT.

0x0050 PLL1_L_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

The equation used to calculate the PLL output frequency is: VCO=TCXO / P * [(L-1)+ M/N)], given L , M, and N, the value in the PLL1_L_VAL, PLL1_M_VAL, and PLL1_N_VAL registers, respectively.

Bits	Name	Description
7:0	PLL_L	This register contains the 8-bit L value used by the PLL1's fractional division ratio.

0x0054 PLL1_M_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Value of numerator of fractional value in the fractional-N PLL.

Bits	Name	Description
10:0	PLL_M	This register contains the 11-bit M value used by the PLL1's numerator value in the fraciotnal division ratio.

0x0058 PLL1_N_VAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Value of denominator of fractional value in the fractional-N PLL.

Bits	Name	Description
11:0	PLL_N	This register contains the 12-bit N value used by the PLL1's denominator value of the fractional division ratio.

0x0068 DCPLL0_MODE

Type: Write/Read
Clock: HCLK

This register configures and controls DCPLL0 (pll2). All register bits will default to 0 on RESOUT.

Bits	Name	Description
3	DCPLL_TEST	Set(1) this bit to enter DCPLL test mode. Clear (0) in normal mode
2	DCPLL_RESET_N	Low asserted reset for the digital logic in the DCPLL. This includes the MND counters and integer divider.
1	DCPLL_BYPASSNL	Clear (0) to bypass the DCPLL. PLLOUT is therefore identical to input reference. Set (1) to use the PLL.
0	DCPLL_OUTCTRL	Set (1) to activate the PLL's output. The Analog circuitry is active but the output is not. Clear (0) to disable the PLL's output and save power.

0x006C DCPLL0_LVAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

The equation used to calculate the DCPLL output frequency: VCO=TCXO / P * (L).

Bits	Name	Description
5:0	DCPLL_L	This register contains the 6-bit L value used by the DCPLL0.

0x007C DCPLL1_MODE

Type: Write/Read
Clock: HCLK

This register configures and controls DCPLL1 (pll3). All register bits will default to 0 on RESOUT.

DCPLL1_MODE

Bits	Name	Description
3	DCPLL_TEST	Set(1) this bit to enter DCPLL test mode. Clear (0) in normal mode
2	DCPLL_RESET_N	Low asserted reset for the digital logic in the DCPLL. This includes the MND counters and integer divider.

DCPLL1_MODE (Continued)

1	DCPLL_BYPASSNL	Clear (0) this bit to bypass the dcpll. PLLOUT is therefore identical to input reference. Set (1) to use the pll.
0	DCPLL_OUTCTRL	Set(1) this bit to activate the pll's output. The Analog circuitry is active but the output is not. Clear (0) to disable the pll's output and save power.

0x0080 DCPLL1_LVAL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

The equation used to calculate the DCPLL output frequency: VCO=TCXO / P * (L-1).

Bits	Name	Description
5:0	DCPLL_L	This register contains the 6-bit L value used by the DCPLL1.

0x0090 CHIPXN_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

The output frequency *Fout* of the clock regimes with the M/N:D counter are determined by the following equation:

```
Fout = Fsrc / (Pre-div) * ( M / N); M< = N/2, 1< = Duty cycle D < = (N-1)
```

Fsrc is the selected raw clk source frequency, *P* is the pre-divided down value, while *M* and *N* are the numerator and denominator values for the M/N counter, and *D* the value to determine duty cycle.

Note for a given *M*, *N* and *D* value used in the above equation, the *N-M* and *2*D* values are to be programmed in 1's complement form into the registers: *_CLK_MD[31:16]=M, *_CLK_MD[15:0]=NOT (2*D), and *_CLK_NS[31:16]=NOT (N-M)

CHIPXN_CLK_MD

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x0094 CHIPXN_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

CHIPXN_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x0098 GPSCHIPXN_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x009C GPSCHIPXN_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

GPSCHIPXN_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	

GPSCHIPXN_CLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00A0 ADSP_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00A4 ADSP_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

ADSP_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00A8 MDSP_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00AC MDSP_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

MDSP_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	

MDSP_CLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00B8 OFFLINE_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00BC OFFLINE_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

OFFLINE_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BIT	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00C8 ICODEC_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00CC ICODEC_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

ICODEC_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BIT	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	

ICODEC_CLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00D0 ECODEC_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00D4 ECODEC_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

ECODEC_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : **bt_clk** (special case) 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00D8 SDAC_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0093F82F

The SDAC M/N counter is used to produced a clock that is used for MP3 application and used to drive the stereo DAC interface inside the QDSP4 and off-chip stereo DAC. The most commonly used clocks include 1.4112 MHz, 1.536 MHz, 1.024 MHz (generated from 2.048 MHz rather than from this M/N counter), etc.

The default value of this register is set as TCXO=19.2MHZ, M=147, N=2000, and D=1000. That is, M=0x93, not (N-M)=0xF8C2, and not (2*D)=0xF82F.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00DC SDAC_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0xF8C20000

SDAC_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BIT	
11	RESERVED_BIT11	
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	FAST_CLK_SEL	This field is used for the fast clock source selection 00 : clock source mux 01 : sdac_1024_clk_source 10 : sdac_ext_mclk_src

SDAC_CLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00E0 GSM_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00E4 GSM_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

GSM_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BIT	
11	RESERVED_BIT11	
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00E8 GP_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00EC GP_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

GP_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BIT	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	

GP_CLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	clk_src_sel	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00F0 CAMCLK_PO_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00F4 CAMCLK_PO_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

CAMCLK_PO_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BITS13_12	

CAMCLK_PO_CLK_NS (Continued)

Bits	Name	Description
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	RESERVED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x00F8 SDCC_MCLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x00FC SDCC_MCLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

SDCC_MCLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	RESERVED_BITS7_6	

SDCC_MCLK_NS (Continued)

Bits	Name	Description
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x0100 MDDI_HOST_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x0104 MDDI_HOST_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

MDDI_HOST_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	RESERVED_BITS13_12	

MDDI_HOST_CLK_NS (Continued)

Bits	Name	Description
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	RESERVED_BITS7_6	
5:4	PRE_DIV_SEL	This field selects to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x0108 MICRO_CLK_MD

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Refer to the description for CHIPXN_CLK_MD.

Bits	Name	Description
31:16	M_VAL	This is the M value for the clock regime's M/N:D counter.
15:0	D_VAL	This is the NOT(2*D) value for the clock regime's M/N:D counter.

0x010C MICRO_CLK_NS

Type: Write/Read
Clock: HCLK
Reset State: 0x0

MICRO_CLK_NS

Bits	Name	Description
31:16	N_VAL	This is the NOT(N-M) value of the clock regime's M/N:D counter when it is used as either modulo-N counter or as full M/N:D counter.
13:12	UNUSED_BITS13_12	
11	RESERVED_BIT11	
10		Reserved.
9	RESERVED_BIT9	
8	MNCNTR_SEL	This field selects to activate or bypass the M/N counter. 0 : bypass 1 : active
7:6	UNUSED_BITS7_6	
5:4	PRE_DIV_SEL	Use this bit to field to activate or bypass the modulo divider before the M/N counter. 00 : bypass 01 : div-2 10 : undefined_encoding 11 : div-4
3:0	CLK_SRC_SEL	This field selects the output of the clock source mux. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3

0x0110 MICRO_ACTIVE_MD

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	MICRO_ACTIVE_MD_VAL	This is the M/D register for the active micro clock regime's M/N:D counter.

0x0114 MICRO_ACTIVE_NS

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	MICRO_ACTIVE_NS_VAL	This is the NS register for the active micro clock regime's M/N:D counter.

0x0118 CHAIN0_CLK_SEL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register contains the clock source selection bits for the chain0 rx front clock regimes.

CHAIN0_CLK_SEL

Bits	Name	Description
6:5	RX_SAMPLE_CSC_DIV	These bits select the divider value of chain0 rx_sample_clk 00 : div1 01 : div 2
4	SAMPCTRL_CLK_SRC_SEL	These bits select the clock source of chain0_sampctrl_clk. 0 : chipx8 (CDMA 1X and HDR mode) 1 : gps_chipx8 (GPS mode)

CHAIN0_CLK_SEL (Continued)

Bits	Name	Description
3:2	RX_FRONT_CLK_SRC_SEL	These bits select the clock source of chain0 rx_front_clk 00 : tcxo (test mode) 01 : chipx16 (CDMA 1X and HDR mode) 10 : gps_chipx16 (GPS mode) 11 : unused_encoding
1:0	RX_SAMPLE_CLK_SRC_SEL	These bits select the clock source of chain0: rx_sample_clk 00 : tcxo 01 : chipx16 10 : gps_cx32 msm_clk_inv2(9) has to be programmed to '1' to invert the clock in gps mode.

0x011C CHAIN1_CLK_SEL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register contains the clock source selection bits for the chain0 rx front clock regimes.

Bits	Name	Description
6:5	RX_SAMPLE_CSC_DIV	These bits select the divider value of chain1 rx_sample_clk 00 : div1 01 : div 2
4	SAMPCTRL_CLK_SRC_SEL	These bits select the clock source of chain0 sampctrl_clk: 0 : chipx8 (CDMA 1X and HDR mode) 1 : gps_chipx8 (GPS mode)
3:2	RX_FRONT_CLK_SRC_SEL	These bits select the clock source of chain1 rx_front_clk 00 : tcxo (test mode) 01 : chipx16 (CDMA 1X and HDR mode) 10 : gps_chipx16 (GPS mode)
1:0	RX_SAMPLE_CLK_SRC_SEL	These bits select the clock source of chain1 rx_sample_clk: 00 : tcxo 01 : chipx16 10 : gps_cx32 msm_clk_inv2(9) has to be programmed to '1' to invert the clock in gps mode.

0x0120 MISC_CLK_SEL1

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register contains the clock selection bits for the type2 clock regimes.

MISC_CLK_SEL1

Bits	Name	Description
20:19	SBI_CLK_SRC_SEL	These bits select the clock source of sbi clock 00 : tcxo 01 : tcxo4 10 : slp_xtal 11 : ring_osc
18	DEC_CLK_SRC_SEL	These bits select the clock source of deinterleaver clock 0 : tcxo 1 : chipx8
17	TURBO_DEC_CLK_SRC_SEL	These bits select the clock source of turbo decoder clock 0 : dem_clk (1x mode) 1 : offline_clk (HDR mode)
16	SLPFAST_FEE_CLK_SRC_SEL	These bits select the clock source of fee slp_ctl clock 0 : chipx8 (1x and HDR mode) 1 : gsm_clk (GSM mode)
15	RESERVED_BIT15	
14:13	GSM_GSAC_CLK_SRC_SEL	These bits select the clock source of GSM hardware accelerator clk source 00 : mdsp_clk 01 : adsp_clk 10 : chipx16
12	MMCDIV_CLK_SRC_SEL	This bit selects the clock source of MMC 0 : mmcdiv_clk_in from MMCC 1 : mmcdiv_clk_ext
11	TCXO4_CLK_SRC_SEL	This bit selects the clock source of general clock 0 : tcxo/4 1 : sleep_xtal
10:9	HKADC_CLK_SRC_SEL	These bits select the clock source of HKADC 00 : tcxo 01 : tcxo/4 10 : sleep_xtal 11 : uart_ext_clk

MISC_CLK_SEL1 (Continued)

Bits	Name	Description
8:7	UART3_CLK_SRC_SEL	These bits select the clock source of UART3 00 : tcxo 01 : tcxo/4 10 : gsm_div2 11 : uart_ext_clk
6:5	UART2_CLK_SRC_SEL	These bits select the clock source of UART2 00 : tcxo 01 : tcxo/4 10 : gsm_div2 11 : uart_ext_clk
4:3	UART1_CLK_SRC_SEL	These bits select the clock source of UART1 00 : tcxo 01 : tcxo/4 10 : sleep_xtal 11 : uart_ext_clk
2	SLPCTL_HDR_CLK_SRC_SEL	This bit selects the clock source of hdr sleep controller. 0 : tcxo/512 1 : sleep_xtal
1	SLPCTL_GSM_CLK_SRC_SEL	This bit selects the clock source of gsm sleep controller. 0 : tcxo/512 1 : sleep_xtal
0	SLPCTL_1X_CLK_SRC_SEL	This bit selects the clock source of 1x sleep controller. 0 : tcxo/512 1 : sleep_xtal

0x0124 MISC_CLK_SEL2

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register contains the clock selection bits for the type2 clock regimes.

MISC_CLK_SEL2

Bits	Name	Description
5:0	I2C_CLK_DIV_SEL	These bits select the divide value of module-N counter of I2Cclock The source of this counter is micro_clk_source. The counter should be programmed as (N-1)
7:6	MMCC_CLK_SRC_SEL	These bits select the clock source of mmcc clock: 00 : tcxo 01 : pll0 10 : micro_clk 11 : gp_clk
13:8	MMCC_CLK_DIV_SEL	These bits select the divide value of module-N counter of mmcc clock The counter should be programmed as (N-1)
14	RESERVED_BIT14	
16:15	USB_CLK_DIV_SEL	These bits are used to select the 48 MHz USB raw clock source. **USB_CLK_DIV_SEL Function** 00 : pll3_out 01 : pll3_out/2 10 : pll3_out/3 11 : pll3_out/4
18:17	MDDI_CLIENT_VFE_CLK_SRC_SEL	Clock source selection for MDDI client clock regime. 00 : hclk 01 : camif_pclk 10 : camclk_po 11 : undefined_encoding
20:19	RESERVED_BITS20_19	
22:21	USB_MCLK_CSC_DIV	These bits select the divide down value of usb_mclk 00 : same as clock source 01 : div-2
23	I2C_CNTR_RST	Set(1) this bit to reset i2c modn6 counter Clear(0) this bit to de-assert reset of i2c_modn6
24	MMCC_CNTR_RST	Set(1) this bit to reset mmcc modn6 counter Clear(0) this bit to de-assert reset of mmcc_modn6

MISC_CLK_SEL2 (Continued)

Bits	Name	Description
26:25	USB_CLK_SRC_SEL	These bits are used to select the 48 MHz USB raw clock source. **USB_CLK_SRC_SEL Function** 00 : tcxo 01 : usb_xtal 10 : pll3_out/N
29:27	MDDI_CLIENT_CLK_SRC_SEL	Clock source selection for MDDI client clock regime. 000 : tcxo 001 : mddi_pad_clk 010 : ext_clk1 011 : ext_clk2 100 : mddi_host_clk

0x0128 ANALOG_CLK0_SEL

Type: Write/Read
Clock: HCLK
Reset State: 0x1

ANALOG_CLK0_SEL

Bits	Name	Description
9		Reserved.
8	CLK_INV	This bit selects the phase of analog clk0. 0 : non-invert 1 : invert
7:5	CLK_SRC_SEL	This bit selects the clock source of BBRX clock. 000 : tcxo 001 : tcxo/2 010 : tcxo/4 011 : gps_chipx32 100 : pll_chipx16 101 : gsm_clk 110 : chipx16(digital)
4	CLK_EN	Set (1) to enable the analog clock . Clear (0) to disable it.
3:2	CLK_DRIVE	These bits are used to select the drive strength of the analog clock source driver.

ANALOG_CLK0_SEL (Continued)

Bits	Name	Description
1	CLK_SE_EN	Set (1) to enable two single-ended analog clock source drivers. Clear (0) to enable one single_ended analog clock source driver.
0	CLK_DIFF_SEL	Set (1) to make the analog clock a differential output. Clear (0) to make it single ended. This bit is SET by default

0x012C ANALOG_CLK1_SEL

Type: Write/Read
Clock: HCLK
Reset State: 0x1

Bits	Name	Description
9		Reserved.
8	CLK_INV	This bit selects the phase of analog clk1. 0 : non-invert 1 : invert
7:5	CLK_SRC_SEL	This bit selects the clock source of BBRX clock. 000 : tcxo 001 : N/A 010 : N/A 011 : gps_chipx32 100 : pll_chipx16 101 : gsm_clk 110 : chipx16 (digital)
4	CLK_EN	Set (1) to enable the analog clock . Clear (0) to disable it.
3:2	CLK_DRIVE	These bits are used to select the drive strength of the analog clock source driver.
1	CLK_SE_EN	Set (1) to enable two single-ended analog clock source drivers. Clear (0) to enable one single_ended analog clock source driver.
0	CLK_DIFF_SEL	Set (1) to make the analog clock a differential output. Clear (0) to make it single ended. This bit is SET by default

0x0130 MSM_CLK_INV1

Type: Write/Read
Clock: HCLK
Reset State: 0x1080

MSM_CLK_INV1

Bits	Name	Description
31	HDR_PDM_CSC_INV	Clock inversion bit for the modulator hdr pdm clock regime.
30	SLPFAST_FEE_CSC_INV	Clock inversion bit for the fee slp controller fast clock regime.
29	SLPCTL_FEE_CSC_INV	Clock inversion bit for the fee slp controller clock regime.
28	HDR_TDEC_CSC_INV	Clock inversion bit for the HDR decoder clock regime.
27	HDR_DEMOD_CSC_INV	Clock inversion bit for the HDR DEMOD clock regime.
26	I2C_CSC_INV	Clock inversion bit for the I2C clock regime.
25	VIDEO_VFE_CSC_INV	Clock inversion bit for the video VFE clock regime.
24	GSM_GSAC_CSC_INV	Clock inversion bit for the gsm gsac clock regime.
23	MMCDIV_CSC_INV	Clock inversion bit for the mmc clock regime.
22	TCXO4_CSC_INV	Clock inversion bit for the general clock regime.
21	UART3_CSC_INV	Clock inversion bit for the uart3 clock regime.
20	UART2_CSC_INV	Clock inversion bit for the uart2 clock regime.
19	UART1_CSC_INV	Clock inversion bit for the uart1 clock regime.
18	SLPFAST_HDR_CSC_INV	Clock inversion bit for the hdr slp controller fast clock regime.
17	SLPFAST_GSM_CSC_INV	Clock inversion bit for the gsm slp controller fast clock regime.
16	SLPFAST_1X_CSC_INV	Clock inversion bit for the 1x slp controller fast clock regime.
15	SLPCTL_HDR_CSC_INV	Clock inversion bit for the hdr slp controller clock regime.
14	SLPCTL_GSM_CSC_INV	Clock inversion bit for the gsm slp controller clock regime.
13	SLPCTL_1X_CSC_INV	Clock inversion bit for the 1x slp controller clock regime.
12	BT_PCM_CSC_INV	Clock inversion bit for the bt_pcm clock regime.
11	VIDEO_HCLK_CSC_INV	Clock inversion bit for the video AHB interface clock regime.
10	ADIE_CODEC_CLK_INV	Clock inversion bit for the codec_clk
9	BT_SLP_CSC_INV	Clock inversion bit for the bt slpctl clock regime.
8	ETM_CSC_INV	etm_clk and micro_clk should always be in phase. Inversion is not supported
7	RPCM_CSC_INV	Clock inversion bit for the rpcm clock regime.

MSM_CLK_INV1 (Continued)

Bits	Name	Description
6	ECODECIF_CSC_INV	Clock inversion bit for the external codec interface clock regime.
5	BT_CSC_INV	Clock inversion bit for the bluetooth clock regime.
4	USB_CSC_INV	Clock inversion bit for the USB clock regime.
3	TURBO_DEC_CSC_INV	Clock inversion bit for the Turbo Decoder clock regime.
2	SRCHCHIPX8_CSC_INV	Clock inversion bit for the 8xsearcher clock regime.
1	RESERVED_BIT1	
0	RESERVED_BIT0	

0x0134 MSM_CLK_INV2

Type: Write/Read
Clock: HCLK
Reset State: 0x7f0000

MSM_CLK_INV2

Bits	Name	Description
31	MDDI_CLIENT_IO_CSC_INV	Clock inversion bit for the MDDI HCLK regime.
30	MDDI_CLIENT_DIV4_CSC_INV	Clock inversion bit for the MDDI HCLK regime.
29	MDDI_CLIENT_VFE_CSC_INV	Clock inversion bit for the MDDI HCLK regime.
28	MDDI_HOST_HCLK_CSC_INV	Clock inversion bit for the MDDI HCLK regime.
27	MDDI_HOST_CSC_INV	Clock inversion bit for the MDDI clock regime.
26	MDP_HCLK_CSC_INV	Clock inversion bit for the MDP HCLK regime.
25	SDCC_MCLK_CSC_INV	Clock inversion bit for the SDCC MCLK regime.
24	SDCC_HCLK_CSC_INV	Clock inversion bit for the SDCC HCLK regime.
23	USB_OTG_HCLK_CSC_INV	Clock inversion bit for the USB OTG HCLK interface clock regime.
22	CHAIN1_SAMPCTRL_CSC_INV	Clock inversion bit for the chain1 sample server clock regime.
21	CHAIN0_SAMPCTRL_CSC_INV	Clock inversion bit for the chain0 sample server clock regime.
20	TXCHIPX8_CSC_INV	Clock inversion bit for the txchipx8 clock regime.
19	TXCHIPX16_CSC_INV	Clock inversion bit for the txchipx16 clock regime.
18	PDM_CSC_INV	Clock inversion bit for the 1x demod pdm clock regime.
17	CHAIN1_RX_FRONT_CSC_INV	Clock inversion bit for the chain1 rx_front clock regime.

MSM_CLK_INV2 (Continued)

Bits	Name	Description
16	CHAIN0_RX_FRONT_CSC_INV	Clock inversion bit for the chain0 rx_front clock regime.
15	GRAPHICS_HCLK_CSC_INV	Clock inversion bit for the graphics AHB interface clock regime.
14	RESERVED_BIT14	
13	RXCHIPX8_CSC_INV	Clock inversion bit for the 1x RTC/combiner clock regime.
12	RTC_GPSCHIPX8_CSC_INV	Clock inversion bit for the GPS RTC clock regime.
11	RTC_HDRCHIPX8_CSC_INV	Clock inversion bit for the HDR RTC clock regime.
10	CHAIN1_RX_SAMPLE_CSC_INV	Clock inversion bit for the chain1 rx_sample clock regime.
9	CHAIN0_RX_SAMPLE_CSC_INV	Clock inversion bit for the chain0 rx_sample clock regime.
8	GSM_CSC_INV	Clock inversion bit for the GSM clock regime.
7	MMCC_CSC_INV	Clock inversion bit for the MMCC clock regime.
6	SDAC_CSC_INV	Clock inversion bit for the SDAC clock regime.
5	ICODEC_CSC_INV	Clock inversion bit for the internal codec clock regime.
4	SBI_CSC_INV	Clock inversion bit for the SBI clock regime.
3	OFFLINE_CSC_INV	Clock inversion bit for the offline engine clock regime.
2	DEC_CSC_INV	Clock inversion bit for the deinterleaver, SVD clock regime.
1	MDSP_CSC_INV	Inversion is not supported
0	ADSP_CSC_INV	Inversion is not supported

0x0138 GLOBAL_RESET

Type: Write/Read
Clock: HCLK
Reset State: 0

Bits	Name	Description
0	GLOBAL_SW_RESET	Setting (1) this bit has the same function as setting (1) both the MSM_CLK_RESETA and MSM_CLK_RESETB: all clock regimes are reset.

0x013C MSM_CLK_FS_ON

Type: Write/Read
Clock: HCLK
Reset State: 0x0

MSM_CLK_FS_ON

Bits	Name	Description
21	MPMC_FB_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
20	MDP_HCLK_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
19	SDCC_HCLK_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
18	MDDI_HCLK_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
17	GRAPHICS_HCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
16	USB_HCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
15	VIDEO_HCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
14	HDR_DEMOD_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
13	MICRO_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
12	HCLK_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
11	RESERVED_BIT11	
10	MMCM_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
9	ETM_CLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
8	UNUSED_BIT8	
7	MPMC_HCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
6	XMEM_HCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
5	UNUSED_BIT5	

MSM_CLK_FS_ON (Continued)

Bits	Name	Description
4	UNUSED_BIT4	
3	MDSP_SCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.
2	UNUSED_BIT2	
1	UNUSED_BIT1	
0	ADSP_SCLK_UP_FS_ON	Set(1) this bit to force the footswitch logic on. Clear(0) to enable the foot switch controller in FSCSC.

0x0140 MSM_CLK_FS_CNT

Type: Write/Read
Clock: HCLK
Reset State: 0xffffffff

These bits are used for the sleep and wake-up values of the footswitch controller CSCs.

MSM_CLK_FS_CNT

Bits	Name	Description
29:27	HDR_DEMOD_CLK_FS_S	These bits are used for the sleep values of the micro footswitch controller CSC.
26:24	HDR_DEMOD_CLK_FS_W	These bits are used for the wake-up values of the micro footswitch controller CSC.
23:21	MICRO_CLK_FS_S	These bits are used for the sleep values of the micro footswitch controller CSC.
20:18	MICRO_CLK_FS_W	These bits are used for the wake-up values of the micro footswitch controller CSC.
17:15	AHB_CLK_FS_S	These bits are used for the sleep values of the ahb footswitch controller CSC.
14:12	AHB_CLK_FS_W	These bits are used for the wake-up values of the ahb footswitch controller CSC.
11:9	MDSP_CLK_FS_S	These bits are used for the sleep values of the mdsp footswitch controller CSC.
8:6	MDSP_CLK_FS_W	These bits are used for the wake-up values of the mdsp footswitch controller CSC.
5:3	ADSP_CLK_FS_S	These bits are used for the sleep values of the adsp footswitch controller CSC.
2:0	ADSP_CLK_FS_W	These bits are used for the wake-up values of the adsp footswitch controller CSC.

0x0144 GPS_SETUP

Type: Write/Read
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
2	CDMA_HDR_MODE	Clear(0) this bit to indicate CDMA_1X mode and enable gps_measure_1x to generate gps_reset. Set(1) this bit to indicate HDR mode and enable gps_measure_hdr to generate gps_reset
1	GPS_MEASURE_HDR	When SYSTEM_MODE = GPS_MODE, setting (1) this bit will switch the system into GPS mode at the upcoming HDR_PNROLL in HDR mode.
0	GPS_MEASURE_1X	When SYSTEM_MODE = GPS_MODE, setting (1) this bit will switch the system into GPS mode at the upcoming SYNC_80M in CDMA 1X mode, or system_reset in test mode.

0x0148 SWITCH_REF_CLK_SEL

Type: Write/Read
Clock: HCLK
Reset State: 0x0

The ref_clk is used as a stable clock source for micro_clk source switch. It is not glitch-free. Glitches will occur when switching the ref_clk source and/or pre-div ratio. SW has to program this register earlier to ensure it is stable prior to switch micro_clk_source. It is also recommended that the ref_clk frequency to be close to target micro_clk frequency rate.

SWITCH_REF_CLK_SEL

Bits	Name	Description
6:5	REF_PRE_DIV_SEL	This field selects the pre-div value of the reference clock source for the micro-clock switch. 00 : div1 01 : div2 10 : undefined_encoding 11 : div4

SWITCH_REF_CLK_SEL (Continued)

Bits	Name	Description
4	REF_EXT_CLK_SEL	This field selects the reference clock source for micro clock switch. 0 : ext_clk1 1 : ext_clk2
3:0	REF_CLK_SEL	This field selects the reference clock source for micro clock switch. 0000 : TCXO 0001 : Sleep Xtal 0010 : USB XTAL 0011 : Ring Oscillator 0100 : PLL0 0101 : PLL1 0110 : PLL2 0111 : PLL3 1 --- : Ext_clk (chosen by bit4)

0x014C ARM_MEMORY_EDGE_EN

Type: Write/Read
Clock: HCLK
Reset State: 0x0

This register bit controls the “edge_en” port on the ARM memories. When the bit is clear (0), the RAM acts normally. When the bit is set (1) the interface latch of the memory does not allow toggling inputs (such as address) to pass through to the internal memory logic for a period of time. This is to allow the toggling inputs to settle before the latch becomes transparent again.

Bits	Name	Description
0	EDGE_EN	Set (1) this bit when the ARM is running at less than 100MHz to save power. Clear (0) this bit when the ARM is running greater than 100MHz to meet timing.

3 AHB

3.1 ARM AHB_REGS registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: AHB_REGS (AHB)
- Number of word addresses: 128
- Byte Address: 0x0200 - 0x03FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

3.1.1 ARM clock controller registers

0x0000 **SWITCH_CLK**

Type: Write (Command)
Clock: HCLK

Bits	Name	Description
0	SWITCH	Writing to this register will trigger a state machine in the clock block, that will switch the clock to the programmed value. Reset state: 0.

0x0004 MICRO_CLK_OFF

Type: Read/Write
Clock: AHB_WR_CLK

The MICRO_CLK_OFF register powers down the microprocessor.

Bits	Name	Description
0	UPCLK_OFF	Writing the power down sequence, 1-0, to this register, will mask the ARM clock after a falling edge. Writing '1' to this register will activate the power down circuit only. Writing a '0' after activation starts the power down logic. Any interrupt detected on IRQ or FIQ triggers a power-up circuit that restores the ARM clock. If an interrupt is detected after activating the power down circuit, this event is saved and is used to power-up the ARM clock AFTER the power down sequence is completed.

0x000C HCLK_DIV

Type: Read/Write
Clock: AHB_WR_CLK

Bits	Name	Description
1:0	HCLK_DIVX	This register determines the active clock rate of the HCLK with respect to the ARM clock. 00 : divide by 1 01 : divide by 2 10 : divide by 3 11 : divide by 4 (Power on value = 0)

3.1.2 AHB reset/pause controller registers

0x0020 PAUSE_TIMER

Type: Write
Clock: HCLK

The PAUSE_TIMER register specifies the duration of time to stall the microprocessor. The actual stall time is 6 HCLK cycles plus the programmed value. The assertion of RESOUT_N disables the pause mechanism. The maximum delay is 6 + 65535 HCLK cycles.

Bits	Name	Description
15:0	PAUSE_TIMER	Stall time = 6 + PAUSE_TIMER

0x0024 RESET_STATUS

Type: Read
Clock: AHB_WR_CLK

This register returns the reset/pause status of the ARM.

Bits	Name	Description
0	RESET_STATUS	The reset status register returns the reset status of the system. This signal is cleared (0) when the source of the last system reset was RESIN_N asserting. When set (1), the source of the last system reset was due to the expiration of the internal watchdog timer.

3.1.3 Interface configuration register

0x0040 MDSP_INTF_CFG

Type: Write
Clock: AHB_WR_CLK

Bits	Name	Description
3:2	MDSP_WR_ACK_SEL	Selects the acknowledge signal to be used on the QDSP4 DMA micro0 write access. 00 : invalid 01 : micro0_ack_d 10 : micro0_ack_d2 11 : micro0_ack_d3 Power up value is 0x1
1:0	MDSP_RD_ACK_SEL	Selects the acknowledge signal to be used on the QDSP4 DMA micro0 read access: 00 : invalid 01 : micro0_ack_d 10 : micro0_ack_d2 11 : micro0_ack_d3 Power up value is 0x3

0x0044 ADSP_INTF_CFG

Type: Write
Clock: AHB_WR_CLK

Bits	Name	Description
3:2	ADSP_WR_ACK_SEL	Selects the acknowledge signal to be used on the QDSP4 DMA micro0 write access. 00 : invalid 01 : micro0_ack_d 10 : micro0_ack_d2 11 : micro0_ack_d3 Power up value is 0x1
1:0	ADSP_RD_ACK_SEL	Selects the acknowledge signal to be used on the QDSP4 DMA micro0 read access. 00 : invalid. 01 : micro0_ack_d 10 : micro0_ack_d2 11 : micro0_ack_d3 Power up value is 0x3

0x0048 IMEM_CFG

Type: Read/Write
Clock: AHB_WR_CLK

IMEM Configuration Register.

IMEM_CFG

Bits	Name	Description
7	RESERVED_BIT7	
6	RESERVED_BIT6	
5	RESERVED_BIT5	
4	RESERVED_BIT4	
3	SRAM_CLK_DIV_CFG	The IMEM SRAM clock divide configuration register bit determines the frequency at which the IMEM SRAM operates at. if this register is configured to a (0), then the SRAM operates at the HCLK frequency, else if configured to a (1), then the SRAM operates at HCLK / 2 frequency. 0 : one to one clock divide (default) to IMEM SRAM 1 : one to two clock divide (use if hclk is > 38MHz) to IMEM SRAM

IMEM_CFG (Continued)

Bits	Name	Description
2	SRAM_PORT_PRIORITY	The IMEM SRAM priority bit only effects the access time of either port when both ports are trying to access the same SRAM bank simultaneously (in the same hclk cycle). In this case, the port with the lower priority will have its memory request delayed until the higher priority port's memory request(s) have completed or do not access the same SRAM bank 0 : priority to port0 (default) 1 : priority to port1
1	SRAM_PORT1_SELECT	The SRAM port1 select register bit determines which bus interface (DMA AHB or Graphics Core dedicated bus) has access to the IMEM SRAM. The bus that is NOT selected cannot gain access (have a request accepted) to the IMEM SRAM until it is selected through this register bit. 0 : access to DMA AHB (default) 1 : access to graphics
0	SRAM_PORT0_SELECT	The SRAM port0 select register bit determines which bus interface (System AHB or Application AHB) has access to the IMEM SRAM. The bus that is NOT selected cannot gain access (have a request accepted) to the IMEM SRAM until it is selected through this register bit. 0 : access to System AHB (default) 1 : access to Application AHB

0x004C MSM_BRIDGE_CFG

Type: Read/Write
Clock: AHB_WR_CLK

MSM Bridge Configuration Register.

Bits	Name	Description
9	DELAY_REQ_EN	Forces msm_address to be stable for one cycle before the msm_req for any transaction (read or write). 1 : Enable 0 : Disable
8	TIMEOUT_EN	Controls time-out feature. 1 : Enable 0 : Disable
7:0	TIMEOUT_VALUE	This value indicates the maximum amount of cycles allowed for a transfer on the msm bus. It is enforced when TIMEOUT_EN is set.

3.1.4 EBI1 registers

0x0060 EBI1_CFG

Type: Read/Write
Clock: AHB_WR_CLK

The EBI1_CFG register configures the following:

EBI1_CFG

Bits	Name	Description
25	ROM1_CLK_DLY_BYPASS	Setting this bit to 1 will bypass the programmable delay cell and the XOR gate which is responsible for clock inversion. This functionality will improve the hold times of all the addr/ctl/data pins with respect to the ROM1_CLK pin. Set (1) this bit when using the XMEMC controller and CMD_DELAY_ENA = 0. Set (1) this bit when using the MPMC controller and EBI1_CLK_POL = 0 This bit must be cleared (0) if the programmable delay cell needs to be used to add additional delay on the ROM1_CLK_PIN.
24	CMD_DELAY_ENA	Setting this bit to 1 implies that all the address/control/data signals are generated from the falling edge of HCLK. This mode is useful only if increased hold time requirements are needed on the EBI1 bus interface. This bit applies only to the XMEM controller in EBI1. This bit has no effect on the MPMC controller On power-up this bit is initialized to 0.
22	ROM1_CLK_STATE	This bit only applies to the Burst Memory controller and has no effect on the MPMC controller accesses to external memory. Setting (1) this bit will ensure that the gating of the ROM1_CLK happens on the falling edge of HCLK. This will result in the ROM1_CLK being at a “logic 0” state when it is gated off. The default state of this bit is 0 which implies ROM1_CLK will be at a “logic 1” state. This bit is useful if the memory requires its CLK pin to be in a “logic 0” state. On power-up this bit is initialized to 0.
21	EBI1_APP_LOCK_ENA	This bit takes effect on both the MPMC controller and the Burst Memory Controller. When this bit is set (1), the HMASTLOCK signal can be used by the respective controllers. Usually, if the LOCK signal is asserted then the transactions on the APP bus will get completed without any interruption as long as the lock is asserted. When this bit is cleared (0), the HMASTLOCK signal from the APP bus is gated OFF and the memory controllers can try to break up a long burst to optimize memory accesses. On power-up this bit is initialized to 0.

EBI1_CFG (Continued)

Bits	Name	Description
20	EBI1_FBCLK_POL	This bit takes effect on both the MPMC Controller and the Burst Memory Controller (XMEMC) When this bit is set (1), the polarity of EBI1_FBCLK is inverted. When this bit is cleared (0), the polarity of EBI1_FBCLK is unchanged. **CAUTION** For the burst memory controller (XMEMC) this bit must be set to 0 *always,* since the inverted clock mode is no longer supported. For the SDRAM (MPMC) controller this bit must be set to 1 when operating in inverted clock mode(i.e. EBI1_CLK_POL = 1 and MPMC_DY_READ_CONFIG = 0x00). On power-up this bit is initialized to 0.
19:14	FBCLK_PGM_DLY_CTL	This register must be programmed by software to support synchronous burst memories (Burst Flash, Burst PSRAM) and SDRAM. This value will delay the fed back clock from the CLK block to match the delay of the ROM1_CLK (including the on-chip routing and external board delay). The optimum value for this register must be determined based on characterization data.from the ATE. On power-up this is initialized to 0.
13	IGN_HROM1_WAIT_N	To support 32-bit PSRAM memories two separate x16 PSRAM memories will be required. There will also be a separate “HROM1_WAIT_N” pin associated with the upper x16 PSRAM. The XMEM controller will monitor both the ROM1_WAIT_N and the HROM1_WAIT_N pins to determine if there is a refresh collision (CellularRAM parts implement this) and will restart the access if this occurs. Many other memories (COSMORAM specification) do not implement the refresh collision feature. In this case the MSM can be configured to ignore the HROM1_WAIT_N pin by setting this bit to 1. This way the value on the pin HROM1_WAIT_n will be ignored. On power-up this bit is initialized to 0.
12		Reserved. On power-up this bit is initialized to 0.

EBI1_CFG (Continued)

Bits	Name	Description
11	FBCLK_SLEEP_OVERRIDE	The FBCLK clock tree is toggling continuously during normal operation. When using the XMEMC the FBCLK will be gated OFF and turned ON only during read accesses from synchronous burst memories. Setting this bit to 1 will override the generation of ebi1_fbclk_sleep. This bit does not have any effect if the MPMC is the active controller This bit is intended to be a power-saving feature. Setting this bit is dependent on the initial access latency of the burst memory. If the latency is > 4 clocks then this bit can be set to 0. If the latency is 4 or less then this bit must be set to 1. On power-up this bit is initialized to 0.
10	PWRSAVE_HANDSHAKE_CTL	By default the ROM1_CLK is toggling after power-up. When this bit is set (1) it allows the PWRSAVE_CLK_DIS bit to turn the ROM1_CLK ON/OFF. To turn off the ROM1_CLK set PWRSAVE_HANDSHAKE_CTL = 1 and PWRSAVE_CLK_DIS = 1. This bit should be used only if all the memories on EBI1 are asynchronous memories which do not need the ROM1_CLK. **NOTE** Use this bit only if the ROM1_CLK toggling needs to be disabled after power-up. After a chip-select is configured into burst mode PWRSAVE_CLK_DIS will control the clock gating. On power-up this bit is initialized to 0.

EBI1_CFG (Continued)

Bits	Name	Description
9	EBI1_CLK_POL	This bit configures the actual polarity of the external memory clock. When clear (0), polarity of external memory source clock (MPMC or XMEMC) is not inverted. When set (1), polarity of external memory source clock (MPMC or XMEMC) is inverted **CAUTION** The EBI1_CLK_POL bit must be set to 0 for the XMEM controller. The inverted clock mode is no longer supported. It is recommended that the CMD_DLY_ENA bit be used for the XMEMC block if increased hold times are required for all the address/ctl/write data signals The MPMC controller still supports the inverted clock mode. This mode will incur 1 less pipeline delay for read bursts. This mode is useful when increased hold times are required (at the expense of smaller setup time). **NOTE** If the EBI1_CLK_POL = 1 mode is used then EBI1_FBCLK_POL must also be set to 1. The behavior is different from MSM6500 where EBI1_FBCLK_POL was required to be set to 0 On powerup this bit is initialized to 0.
8	SDRAM_PRESENT	Set this bit to high if SDRAM is being used AND code execution is out of SDRAM. This bit will be used for implementing Power Management functions pertaining to the SDRAM. On powerup this bit is initialized to 0.
7	PWRSAVE_CLK_DIS	This bit only applies to the Burst memory controller (XMEMC) When this bit is cleared (0), the burst clock coming out on the pin will always be toggling once any chip select is configured to be a burst memory. When this bit is set (1), the burst clock will be allowed to toggle only for accesses to burst memory. On power-up this bit is initialized to 0.

EBI1_CFG (Continued)

Bits	Name	Description
6	IGN_WAIT_FOR_WRITE	This bit applies only to the XMEMC controller and will be used for supporting burst PSRAM/FCRAM/CellularRAM memories. The XMEMC controller always monitors the ROM1_WAIT_N (and HROM1_WAIT_N pin for x32 PSRAM interface) once the chip-select is configured for burst mode operation. Set this bit to 1 If the PSRAM memory device is expected to operate in a fixed latency mode for burst write accesses and does not drive the WAIT/ pin. On power-up this bit is initialized to 0.
5:0	PGM_DLY_CTL	This register must be programmed by software for supporting burst memories (Burst Flash, Burst PSRAM) and SDRAM. The value programmed in these bits will add a delay to the ROM1_CLK output. This value must be programmed to ensure that the control/address/data signals will meet the setup/hold time requirements of the memory. Bit[5] is the control bit to enable the DELAY. If bit[5] is set to 0 the lower bits of this register will NOT have any effect. On power this value is initialized to 0.

0x0064 EBI1_MPMC_STDY_SEL

Type: Write
Clock: AHB_WR_CLK

The EBI1_CFG register configures the following:

Bits	Name	Description
1	MPMC_STDY_RAMCS3_SEL	Clear (0) this bit to configure chip select to static memory. Set (1) this bit to configure chip select to dynamic memory.
0	MPMC_STDY_RAMCS2_SEL	Clear (0) this bit to configure chip select to static memory. Set (1) this bit to configure chip select to dynamic memory.

0x0068 EBI1_PSRAM_CRE

Type: Write
Clock: AHB_WR_CLK

The EBI1_PSRAM_CRE register controls the SDRAM1_CLK_EN(this pin is available as a General Purpose Output when the MPMC controller is not active) pin for controlling synchronous PSRAM operations such as Deep Power Down, Programming the access type (wrap, incr, etc.).

Bits	Name	Description
0	PSRAM_CRE	Clearing (0) this bit outputs a 0 on the EBI1 SDRAM1_CLK_EN pin. Setting (1) this bit outputs a 1 on the EBI1 SDRAM1_CLK_EN pin.

0x006C+0x8n EBI1_CSn_CFG0, n=[0,3]

Type: Read/Write
Clock: AHB_WR_CLK

The EBI1_CSn_CFG registers configures the parameter and controller functionality of the QCT XMEM interface. Only CS0 and CS2 have power up default values.

(The bit table for this register is on the next page.)

EBI1_CSn_CFG0

Bits	Name	Description
30:28	PAGE_SIZE	This bit field indicates the page size of the page memory. Note that the PAGE_xx_ENA/BURST_xx_ENA must be set high (1) for this value to take effect. For a page memory device this bitfield will be used to insert the DELTA cycles whenever an access crosses the boundary specified in this bitfield. For burst memories this bit field is important to allow insertion of precharge cycles when going across these page boundaries. For 16-bit memories the definition is as follows: Value Page Size (in hwords) 000 No page mode 001 4 word page 010 8 word page 011 16 word page 100 32 word page 101 64 word page 110 128 word page 111 256 word page For 32-bit memories using 2 x16 parts the definition is as follows: Value Page Size (in hwords) 000 No page mode 001 2 word page 010 4 word page 011 8 word page 100 16 word page 101 32 word page 110 64 word page 111 128 word page Power-up value is 0x0.
27:24	PRECHARGE_CYC	This determines the number of clock cycles inserted between consecutive read or write accesses to the same chip select or when a burst access crosses the page boundary defined above. This bitfield is needed for some Toshiba PSRAMs, which require CS_N to be toggling whenever the access crosses a page boundary for reads and between consecutive write accesses. This bitfield applies to both page and burst mode memories. For non page crossing accesses the total number of cycles will be PRECHARGE_CYC + 1 for accesses that cross the page boundary the total will be PRECHARGE_CYC. Power up value is 0x00.

EBI1_CSn_CFG0 (Continued)

Bits	Name	Description
23:20	RECOVERY	This bit field determines how many recovery cycles are inserted from the time the current chip select is deasserted after a read access to the assertion of the next chip select, in order to allow more data recovery time and thus avoid bus contention. Recovery cycles will be inserted in the beginning of the data phase of the next transfer. The chip select and read/write enable signals are delayed after the recovery cycles. The recovery cycles are inserted under the following conditions: CASE 1: chip select turn-over 1) The following chip select signal is different from the current one, and 2) the current access to the external memory is a read. CASE 2: read followed by a write on the same chip select 1) The following chip select signal is the same as the current one, and 2) the current access is a read, and the following access is a write. In both cases there will always be 2 recovery cycles and any setting below 0x02 will result in 2 all settings > 0x01 will result in a total of RECOVERY + 1 cycles. The maximum number of programmable values to this bit field is 15 for 16 cycles of recovery time. Program 0 or 1 if the device speed is fast enough (or bus clock speed is slow enough) to eliminate the need for additional recovery cycles. The configuration of the bit field is per chip select basis so one can program device-dependent information to the bit field of the configuration register for each chip select. Powerup value is 0x03.
19:16	HOLD_WAIT	The bit field determines how many hold cycles are inserted after WE_N goes HIGH. The hold time of a write pulse (HIGH) is N bus cycles, where N is the value programmed to this field. The address, data and chip select signals change after the hold cycles. The value upon assertion of RESOUT_N is 3 for cs_n(0) **A minimum value of 1 must be programmed on this bitfield.** Powerup value is 0x03.
15:12	DELTA_WR	These bits determine the number of extra (delta) cycles inserted in the first write access to a page or burst memory. On power-up the delta cycles are inserted in every access to the memory. If CS0 is enabled for Burst accesses then the delta cycles are inserted only in the first access to memory. **A minimum value of 1 must be programmed on this bitfield.** Power up value is 0x0F.
11:8	DELTA_RD	These bits determine the number of extra (delta) cycles inserted in the first read access to a page or burst memory. On power-up the delta cycles are inserted in every access to the memory. If CS0 is enabled for Page or Burst accesses then the delta cycles are inserted only in the first access to memory. Power up value is 0x0C.

EBI1_CSn_CFG0 (Continued)

Bits	Name	Description
7:4	WAIT_WR	These bits determine the number of waitstates incurred in every write access. If BURST_WR_ENA is set to 1 then: * The first access to a page mode memory will incur WR_DELTA + (WR_WAIT + 1) cycles. * Subsequent accesses to a page mode memory will incur WAIT_WR + 1 cycles. Power up value is 0x00.
3:0	WAIT_RD	These bits determine the number of waitstates incurred in every read access. If PAGE_RD_ENA or BURST_RD_ENA is set to 1 then: * The first access to a page will incur RD_DELTA + (RD_WAIT + 1) cycles. * Subsequent accesses will incur WAIT_RD + 1 cycles. Power up value is 0x00

0x0070+0x8n EBI1_CSn_CFG1, n=[0,3]

Type: Read/Write
Clock: AHB_WR_CLK

The EBI1_CSn_CFG1 register configures parameters and functionality of the QCT XMEM:

EBI1_CSn_CFG1

Bits	Name	Description
19:18	AVD_OE_RECOVERY	This bitfield indicates the # of cycles elapsed before OE_N assertion. The reference is w.r.t the cycle in which AVD_N is asserted. For example a value of “10” implies 2 cycles between ADV_N deassertion and OE_N assertion Power up value is 0x00
17	ADDR_HOLD_ENA	This bit applies only for a 32-bit burst interface. In this case the address will be held for an extra cycle to meet hold time requirements with ADV_N. This bit powers up to 0 after reset.
16	ENA_x32_INTERFACE	Setting this bit to 1 implies the chip select is hooked up to a 32-bit burst memory configuration with address pins A1[17:2] being multiplexed as the databus
15	RESERVED_BIT15	
14	INCR_WR_MODE	This bit overrides the WRAP8_MODE bit for write accesses. When this bit is set all burst writes will be continuous.

EBI1_CSn_CFG1 (Continued)

Bits	Name	Description
13	WE_TIMING	This bit applies only if CS0_BURST_WR_ENA is set to 1. Setting this bit to 0 will imply that the WE_N signal will be a pulse for all burst memory accesses. The pulse WE_N strobe will be used by the Micron/Infineon/Cypress memory vendors. Setting to 1 implies WE_N stays lows for the duration of the access and has a similar timing as OE_N. This mode is useful for some of the COSMORAM memories
12	AVD_PULSE_ENA	Setting this bit to 1 will imply that the ADV_N signal will be a pulse for all asynchronous/page memory accesses. This bit MUST BE set to 1 when a 32-bit burst memory is connected to the chip-select. This is required during the configuration of the burst device. Setting to 0 implies ADV_N stays lows for the duration of the access.
11:10	WR_PRECHARGE	The Write-Read Precharge cycles will be inserted in between a Burst Write and a Read access to CS0. The number of cycles inserted will be 1 greater than the actual value. For example, a value of “10” will imply 3 cycles between a Burst Write and a Read access to CS0.
9:8	AVD_RECOVERY	The AVD_N pulse has to stay high for a minimum duration prior to its going low during a synchronous access. This can occur for the following conditions: 1) async/page write access to CS0 followed by any read access to CS 0/1/2/3. 2) async/page write access to CS0 followed by any write access to CS1/2/3. The value programmed will ensure that the minimum “high” pulse width requirement on AVD_N will be met. This condition occurs normally for an asynchronous/page Read/Write access followed by a burst Read/Write access. However, this bitfield will also impact the following types of accesses: 1) Burst write access to CS0 followed by Burst Read access to CS 0/1/2/3. 2) Burst write access to CS0 followed by Burst Write access to CS 1/2/3 (Note that Write to CS0 followed by Write to CS0 not impacted) For example: Programming “10” will imply the insertion of 2+1=3 clock cycles for the following different conditions: 1) Async/Page/Burst Write - Any Read 2) Async/Page/Burst Write - Any Write to different chip-select Power up value is 0x0
7	RESERVED_BIT7	Power up value is 0x0.

EBI1_CSn_CFG1 (Continued)

Bits	Name	Description
6	WRITE_PROTECT	If this bit is set to 1 then writes to CS0 will be allowed only in SUPERVISOR mode. USER mode writes will generate aborts. Power up value is 0x0.
5	WRAP8_MODE	Setting this bit (1) implies CS0 page size is configured to be 16 hwords and the accesses will be optimized for WRAP8 operation. Clearing (0) this bit Implies CS0 is configured in the INCR mode (default). Power up value is 0.
4	AVD_STATE	Setting this bit (1) implies AVD_N is high during access to CS0. Clearing (0) this bit implies ADV_N is low during access to CS0(default). This bit MUST be set only for asynchronous/page memories. Power up value is 0.
3	BURST_WR_ENA	Setting this bit (1) implies CS0 support synchronous burst writes. Clearing (0) this bit implies CS0 is connected to an asynchronous memory (default). Power up value is 0.
2	BURST_RD_ENA	Setting this bit (1) implies CS0 supports synchronous burst reads. Clearing (0) this bit implies CS0 is connected to an asynchronous memory (default). Power up value is 0.
1	PAGE_WR_ENA	Setting this bit to Setting (1) this bit will NOT insert delta cycles to subsequent write accesses within the same page for a page mode memory Clearing (0) this bit will insert delta cycles for every write access Power up value is 0. This bit should be set to 1 only if the Memory supports page writes.
0	PAGE_RD_ENA	Setting (1) this bit causes delta cycles NOT to be inserted to subsequent read accesses within the same page for a page mode memory. Clearing (0) this bit causes delta cycles to be inserted for every read access (default). Power up value is 0.

0x008C EBI1_MEM_CTLR_SEL_CMD

Type: Command
Clock: HCLK

Bits	Name	Description
0	DATA	Data written to this register is ignored. Writing to this register will cause a switch to the other controller.

0x0090 EBI1_MEM_CTLR_SEL_STATUS

Type: Read
Clock: AHB_WR_CLK

Bits	Name	Description
0	DATA	Reading this register will determine which controller is currently active. Power-on value is 0. Clearing (0) this bit implies XMEMC is the active controller. Setting (1) this bit implies MPMC is the active controller.

3.1.5 EBI2 registers

0x00B0 EBI2_CFG

Type: Write
Clock: AHB_WR_CLK

This register configures EBI2 interface functionality.

Bits	Name	Description
1	NAND_BUS_EN	This bit indicates whether the NAND controller has control over the GP0_CS_N pin. 1 : NAND_CS ON (the NAND controller has control of the GP0_CS_N, that is, GP0_CS_N pin outputs the NAND controller's nand_cs_n). 0 : GP0_CS ON (the NAND controller does NOT have control of the GP0_CS_N, that is, GP0_CS_N pin outputs the XMEM controller GP0_CS_N chip select). When access is attempted by the NAND controller, the data and control pins toggle under the control of the NAND controller, however, the NAND access does not occur since the NAND chip select is not routed to the GP0_CS_N output pin. Power on value is 1.
0	EBI2_PRIORITY	This bit controls the relative arbitration priority between the DSP AHB bus and the ARM AHB bus. 0 : AHB HIGHER (the ARM AHB bus has higher priority than the DSP AHB bus.) 1: DSP HIGHER (the DSP AHB bus has higher priority than the ARM AHB bus.) Power on value is 0.

0x00B4 LCD_CFG0

Type: Write
Clock: AHB_WR_CLK

This register configures the characteristics of the external memory access to a parallel LCD connected to the LCD chip select (lcd_cs_n). The register does not have a power-up default value and must be initialized before the chip select is used. See GPn_CFG0 descriptions for the descriptions of the bits.

Bits	Name	Description
31:28	LCD_CS_SETUP	Field applies to both Motorola (6800) and Intel (8080) types of parallel LCD interface select.
27:24	LCD_RECOVERY	Field applies to both Motorola (6800) and Intel (8080) types of parallel LCD interfaces.
23:22	RESERVED_BITS23_22	
21:20	LCD_HOLD_WR	Field applies ONLY to Intel (8080) type of parallel LCD interface. When using Intel (8080) type of parallel LCD interface, program this field to a minimum of 0x1. When using Motorola (6800) type of parallel LCD interface, program this field to 0x0.
19:18	RESERVED_BITS19_18	
17:16	LCD_HOLD_RD	Field applies ONLY to Intel (8080) type of parallel LCD interface. When using Intel (8080) type of parallel LCD interface, program this field to a minimum of 0x0. When using Motorola (6800) type of parallel LCD interface, program this field to 0x0.
15:8	LCD_WAIT_WR	Field applies to both Motorola (6800) and Intel (8080) types of parallel LCD interfaces. In both cases indicates for how long the lcd_cs_n signal is asserted during a write access. For Motorola (6800) type of interface, the minimum value that can be programmed is n = 2. For Intel (8080) type of interface, the minimum value that can be programmed is n = 1.
7:0	LCD_WAIT_RD	Field applies to both Motorola (6800) and Intel (8080) types of parallel LCD interfaces. In both cases indicates for how long the lcd_cs_n signal is asserted during a read access. For Motorola (6800) type of interface, the minimum value that can be programmed is n = 2. For Intel (8080) type of interface, the minimum value that can be programmed is n = 1.

0x00B8 LCD_CFG1

Type: Write
Clock: AHB_WR_CLK

This register configures the characteristics of the external memory access to a parallel LCD connected to the LCD chip select(lcd_cs_n). The register does not have a power-up default value and must be initialized before the chip select is used.

LCD_CFG1

Bits	Name	Description
25	LCD_18_BIT_MODE	This bit indicated whether the lcd is in normal datapath mode or in 18-bit data datapath mode. In this mode WORD accesses are required for 18-bit transfers. Bus sizing is disabled. 1 : ENABLE (lcd 18-bit mode). 0 : DISABLE (lcd 18-bit mode).
24	LCD_INTERFACE_TYPE	This bit indicates what type of parallel LCD interface the lcd_cs_n is connected to. When cleared (0), the lcd_cs_n connects to an Intel (8080) type of parallel LCD interface. When set (1), the lcd_cs_n connects to a Motorola(6800) type of parallel LCD interface.
23	LCD_WP	See GPn_CFG1.
22	LCD_BYTE_DEVICE_EN	See GPn_CFG1.
21:18	LCD_E_SETUP_WRITE	These bits control how many clock cycles are used from the start of LCD_CS_N assertion to LCD_E pin asserting high during a LCD write access. To get *n* clock cycles, write *n* to those bits. Minimum allowed value for n is 1. This field applies only to a Motorola(6800) type parallel LCD interface. When using an Intel(8080) type of parallel LCD interface, program this field to 0x0.
17:14	LCD_E_SETUP_READ	These bits control how many clock cycles are used from the start of LCD_CS_N assertion to LCD_E pin asserting high during a LCD read access. To get *n* clock cycles, write *n* to those bits. Minimum allowed value for *n* is 1. This field applies only to a Motorola(6800) type of parallel LCD interface. When using an Intel(8080) type of parallel LCD interface program this field to 0x0.

LCD_CFG1 (Continued)

Bits	Name	Description
13:7	LCD_E_HIGH_WRITE	These bits control how many clock cycles are used from LCD_E pin asserting high to LCD-E pin de-asserting low, i.e., the pulse width of the LCD_E pin, during an LCD write access. To get *n* clock cycles, program these bits to *n*. Minimum required value for *n* (in order for LCD_E to assert) is 1. This field applies only to a Motorola(6800) type of parallel LCD interface. When using an Intel(8080) type of parallel LCD interface, program this field to 0x0.
6:0	LCD_E_HIGH_READ	These bits control how many clock cycles are used from LCD_E pin asserting high to LCD-E pin de-asserting, i.e., the pulse width of the LCD_E pin, during an LCD read access. To get *n* clock cycles, program these bits to *n*. Minimum required value for *n* (in order for LCD_E to assert) is 1. This field applies only to a Motorola(6800) type of parallel LCD interface. When using an Intel(8080) type of parallel LCD interface, program this field to 0x0.

0x00BC+0x8nGPn_CFG0, n=[0,3]

Type: Read/Write
Clock: AHB_WR_CLK

Configures the characteristics of the external memory access to the memory connected to the General Purpose (GP) chip selects. The register does not have a power-up default value and must be initialized before the chip select is used.

Bits	Name	Description
31:28	CS_SETUP	This bit field indicates how many cycles after the address is present on the external memory bus the cs_n signal asserts. This feature is useful to support some types of pseudo-ram memories. In general, it can be used to provide more setup time for the address. For a bus-sized transfer, this bit field only affects the first access of the transfer. Recovery wait states and cs_setup wait states have the same effect on the access. Therefore, when both wait states are required, only the field with the highest number of wait states is applied, but not both. Default 0x0

(Continued)

Bits	Name	Description
27:24	RECOVERY	This bit field determines how many recovery cycles are inserted from the time the current chip select is deasserted after a read access to the assertion of the next chip select, in order to allow more data recovery time and thus avoid bus contention. A recovery cycle is an extra cycle inserted in the beginning of the data phase of the next transfer. The chip select and read/write enable signals are delayed after the recovery cycles. The recovery cycles are inserted under the following conditions: CASE 1: chip select turn-over 1) The following chip select signal is different from the current one, and 2) the current access to the external memory is a read. CASE 2: read followed by a write on the same chip select 1) The following chip select signal is the same as the current one, and 2) the current access is a read, and the following access is a write. The maximum number of programmable values to this bit field is 15 for 15 cycles of recovery time. Program 0 if the device speed is fast enough (or bus clock speed is slow enough) to eliminate the need for a recovery cycle. The configuration of the bit field is per chip select basis so one can program device-dependent information to the bit field of the configuration register for each chip select. For a bus-sized transfer, this bit field only affects the first access of the transfer. Recovery wait states and cs_setup wait states have the same effect on the access. Therefore, when both wait states are required, only the type with the highest number of wait states is applied, but not both.
23:22	RESERVED_BITS23_22	
21:20	HOLD_WR	This bit field determines how many hold cycles (“dead cycles”) are inserted after we_n deasserts (after we_n rising edge). A hold cycle is an extra cycle inserted after the rising edge of we_*n* for which *we_n* stays HIGH. The address, data, and chip select signals change after the hold cycles.The hold time is n bus cycles, where *n* is the value programmed to this bit field. The minimum value that should be written to this field is 1, for 1 cycle of hold time.
19:18	RESERVED_BITS19_18	
17:16	HOLD_RD	This bit field determines how many hold cycles (“dead cycles”) are inserted after oe_n deasserts (after oe_n rising edge). A hold cycle is an extra cycle inserted after the rising edge of oe_*n* for which *oe_n* stays HIGH. The address and chip select signals change after the hold cycles.The hold time is *n* bus cycles, where *n* is the value programmed to this bit field. The minimum value that should be written to this field is 0, for 0 cycles of hold time. If a value > 0 is written in this field, care should be taken to ensure keepers can hold the read data until the end of the access which is when the microprocessor consumes the data. **Typical devices require n = 0.**

(Continued)

Bits	Name	Description
15:8	WAIT_WR	These bits control the minimum number of clock cycles used during a 16-bit or 8-bit access to the external memory (i.e., no bus sizing.) The minimum total access time is *n*+1 bus cycles, where *n* is the value programmed to this bit field. Twice or four times as many (*n*+1) bus cycles are used if a 32-bit access is performed, because it requires two or four bus sizing phases. Note that extra access cycles in addition to the minimum programmed value in the bit field can be added at the beginning of an access as defined by Delta Wait State. **The minimum programmed value should be greater than or equal to 1**.
7:0	WAIT_RD	These bits control the minimum number of clock cycles used during a 16-bit or 8-bit access to the external memory (i.e., no bus sizing.) The minimum total access time is *n*+1 bus cycles, where *n* is the value programmed to this bit field. Twice or four times as many (*n* +1) bus cycles are used if a 32-bit access is performed, because it requires two or four bus sizing phases. Note that extra access cycles in addition to the minimum programmed value in the bit field can be added at the beginning of an access as defined by Delta Wait State. **The minimum programmed value should be greater than or equal to 1.**

0x00C0+0x8n GPn_CFG1, n=[0,3]

Type: Write
Clock: AHB_WR_CLK

This register configures the characteristics of the external memory access to the memory connected to the GP chip select. The register does not have a power-up default value and must be initialized before the chip select is used.

Bits	Name	Description
4	WP	This bit controls the chip select write protection feature against writes to the memory connected to the chip select when the access is done in ARM user mode. 1 : WP_ENABLED 0 : WP_DISABLED
3	BYTE_DEVICE_EN	This bit controls the bsizing function of the external memory interface. 0 : 16 BIT DEVICE (connected to the chip select). 1 : 8 BIT DEVICE (connected to the chip select).
2:0	RESERVED_BITS2_0	RESERVED

0x00DC ETM_GPIO2_CFG

Type: Write
Clock: AHB_WR_CLK

The ETM_GPIO2_CFG register is identical to GP0_CFG0 except the referenced chip select (ETM_GPIO2) and has no default value upon the assertion of RESOUT_N. Please refer to the content of the GP0_CFG0 register for configuration information. This register is not readable.

Bits	Name	Description
31:28	CS_SETUP	Refer to the description of the same field in the GPn_CFG0 register.
27:24	RECOVERY	Refer to the description of the same field in the GPn_CFG0 register.
23:22	RESERVED_ BITS23_22	Refer to the description of the same field in the GPn_CFG0 register.
21:20	HOLD_WR	Refer to the description of the same field in the GPn_CFG0 register.
19:18	RESERVED_ BITS19_18	Refer to the description of the same field in the GPn_CFG0 register.
17:16	HOLD_RD	Refer to the description of the same field in the GPn_CFG0 register.
15:8	WAIT_WR	Refer to the description of the same field in the GPn_CFG0 register.
7:0	WAIT_RD	Refer to the description of the same field in the GPn_CFG0 register.

3.1.6 Boot ECC status registers

0x100 BOOT_ECC_SELF_ERR

Type: Read
Clock: AHB_WR_CLK

Records ECC SELF ERROR condition found by the NAND flash controller while the boot sequencer loads the first 16 pages of data from the NAND flash during booting from NAND flash.

Bits	Name	Description
0	BOOT_ECC_SELF_ERR	When the MSM boots up from the NAND flash, the boot sequencer loads the first 16 pages of the NAND flash to an on-chip SRAM. For each page read, if there is a 1-bit error in the ECC parity region, the boot sequencer would continue with the boot-up process, but would record this error condition in this register. After all 16 pages have been loaded into the SRAM, if this register has a value of 1, then 1 or more of the 16 pages read has a 1-bit error in the ECC parity region. Power up value is 0x0.

0x0104 BOOT_ECC_CORRECTED

Type: Read
Clock: AHB_WR_CLK

Records whether ECC corrections are performed while the boot sequencer loads the first 16 pages of data from the NAND flash during booting from NAND flash.

Bits	Name	Description
15:0	BOOT_ECC_CORRECTED	When the MSM boots up from the NAND flash, the boot sequencer loads the first 16 pages of the NAND flash to an on-chip SRAM. For each page read, if the NAND controller performs ECC corrections to the data, the boot sequencer would continue with the boot-up process, but would record this condition in this register. Each bit is associated with a page, with bits 15:0 representing page 15:0, respectively. After all 16 pages have been loaded into the SRAM, each bit in this register, if containing a 1, indicates that the corresponding page has been corrected by the NAND flash controller's ECC correction logic. Power up value is 0x0.

3.1.7 BOOT control registers

0x0120 BOOT_MODE_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
1	BOOT_MODE2	Indicates data width of external NAND device. 1 : 16_BIT 0 : 8_BIT
0	BOOT_MODE	Indicates from which memory the msm boots. 1 : NAND BOOT MODE 0 : NOR BOOT MODE

3.1.8 Delay control register

0x0130 DELAY_CNTL

Type: Write
Clock: AHB_WR_CLK

In order to meet the TPA device's timing requirement for ETM's TRACECLK, this register can be programmed to delay the arrival of TRACECLK to the pin if necessary. All bits of this register have power-on value of 0.

Bits	Name	Description
5	DELAY_CNTL_EN	Clearing (0) this bit disables the delay chain, and bits 4:0 are ignored. Setting (1) this bit enables the delay chain; the amount of delay is determined by the value in bits 4:0.
4:0	DELAY_CNTL	The value in these bits is the amount of delay.

3.2 ARM Framer registers

0x0000 FRAMER_CMD

Type: Write
Clock: HCLK

The FRAMER_CMD register issues a software command to the framer logic. Writing to this register causes the framer to perform different functions, based on the value of the databus LSBs.

Bits	Command	Description
4	DRAIN	The software issues this command to force the residual bytes to be written into the framer buffer.
3	INSERT_CRC	This command forces the framer to insert the two CRC bytes into the framer.
2	INSERT_FLAG	This command forces the framer to insert a flag byte (0x7e) into the framer buffer.
1	WRITE_DONE	After all data is written, the software issues this command, so the hardware can reset the write address pointer for the next write.
0	RESET	This command is the framer software reset.

0x0004 FRAMER_ACCM_VALUE

Type: Read/Write
Clock: HCLK

Bits	Name	Description
31:0	DATA	The FRAMER_ACCM_VALUE register provides the ACCM value for the hardware. A mapped value between 0x00 and 0x1F is escaped during the framing process if the corresponding bit in this register is set (1). For example, a value of 0x000000FF in this register means 0x00-0x07 is escaped during the framing process. This power-up value for this register is 0xFFFFFFFF. But, the software usually changes the value to 0x00000000 after network negotiation.

0x0008 FRAMER_BYTE_CNT

Type: Read
Clock: HCLK

Bits	Name	Description
7:0	DATA	The FRAMER_BYTE_CNT register is prepared by the hardware to tell the software how many bytes have been written into the framer buffer. It indicates the framer buffer usage during a write operation and how many bytes to read out of the buffer after a certain number of bytes have been written into the framer.

0x0018 FRAMER_CTL

Type: Write
Clock: HCLK

Bits	Name	Description
1	IGNORE_MSB	Set this bit(1) to give option to ignore the MSB
0	FRAMER_ESC_DEL	Set this bit (1) to give option to escape the DEL characters(0x7F).

0x0080 FRAMER_DATA_IN

Type: Write
Clock: HCLK

Bits	Name	Description
31:0	DATA	The FRAMER_DATA_IN register is the base address of the framer buffer. Data written to this register is framed according to the HDLC framing rules and the specified async-control-character-map (ACCM) value.

0x00C0 FRAMER_DATA_OUT

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	DATA	The software reads the framer buffer contents through the FRAMER_DATA_OUT register. Reading this register causes the internal address pointer for the framer buffer to increase by 1 and to load the next word contents from the buffer into this register. This allows a pre-loading mechanism to be used for single-cycle read access to the deframer buffer. Any access to this register is treated as a word access. Accesses to this register are single-cycle accesses.

3.3 ARM Deframer registers

0x0000 DEFRAMER_CMD

Type: Write/Command
Clock: HCLK

Bits	Name	Description
2:0	DATA	The DEFRAMER_CMD register issues software commands to the deframer logic. Writing to this register causes the deframer to perform different functions, based on the value of the databus LSBs. 00 : deframer software reset (RESET) 01 : write address reset (After all data in an RLP buffer is written, software issues this command so that the deframer resets the write address for the next write (WRITE_DONE.)

0x0004 DEFRAMER_HEADER_CNT

Type: Read
Clock: HCLK

Bits	Name	Description
7:0	DATA	The DEFRAMER_HEADER_CNT register is prepared by the hardware when an RLP buffer is deframed. After the software finishes writing an entire RLP buffer, the software reads this register to determine how many PPP packets are inside the output buffer.

0x0008 DEFRAMER_BYTE_CNT

Type: Read
Clock: HCLK

Bits	Name	Description
7:0	DATA	The DEFRAMER_BYTE_CNT register is prepared by the hardware when an RLP buffer is deframed. The software reads this register at any time during the write process and determines how many bytes of the deframer buffer have been used up.

0x000C Reserved.

0x0080 DEFRAMER_DATA_IN

Type: Write
Clock: HCLK

Bits	Name	Description
31:0	DATA	The DEFRAMER_DATA_IN register is the base address of the deframer buffer. Data that is written to this register is deframed according to the HDLC framing rules. The deframed data is then written to the deframer buffer.

0x00C0 DEFRAMER_DATA_OUT

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	DATA	The software reads the deframer buffer contents through the DEFRAMER_DATA_OUT register. Reading this register causes the internal address pointer of the deframer buffer to increase by 1 and load the next word contents from the buffer into this register. This allows a pre-loading mechanism to be used for single-cycle read access to the deframer buffer. Any access to this register is treated as a word access and accesses to this register are single-cycle accesses.

4 MPMC

4.1 ARM registers

The registers described in this document were taken from the ARM PrimeCell MultiPort Memory Controller (PL172) Design Manual (PL172 DDES 0000 A). The names were modified to comply with QUALCOMM naming conventions. Refer to the ARM document for complete register descriptions.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.These MPMC registers are not actually used by the VHDL(PL172). They are meant mainly for simulation/testing purposes. All of these have to be declared as WORD. Byte/hword accesses to the MPMC register will generate an error on the HRESP signals.

- Block name: MPMC
- MPMC Memory Space: 0x80001000 - 0x80001FFC
- Offset =CHIP_BASE+0x1000
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x000 MPMC_CONTROL

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_CONTROL	This register is used to control the operation of the memory controller. 0x1: Reset value HRESETn 0x3: Reset value nPOR (ARM: MPMCControl)

0x004 MPMC_STATUS

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
2:0	MPMC_STATUS	This register provides the status of the memory controller. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCStatus)

0x008 MPMC_CONFIG

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:8	MPMC_CONFIG	This register is used to configure the operation of the memory controller. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCConfig)

0x020 MPMC_DY_CNTL

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
14:0	MPMC_DY_CNTL	This register is used to control the operation of the dynamic memory controller. N/A: Reset value HRESETn 0x002: Reset value nPOR (ARM: MPMCDyCntl) (Reserved bits - 12:9, 6:3)

0x024 MPMC_DY_REF

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
10:0	MPMC_DY_REF	This register is used to control the refresh period of the Dynamic memories. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCDyRef)

0x028 MPMC_DY_TRD_CFG

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
1:0	MPMC_DY_TRD_CFG	This register configures the dynamic memory read strategy. 0x0: clock out delayed strategy (MPMCCLKOUT used) 0x1: command delayed strategy (MPMCCLKDELAY used) 0x2: command delayed strategy plus one clock cycle (MPMCCLKDELAY used) 0x3: command delayed strategy plus two clock cycles (MPMCCLKDELAY used)

0x030 MPMC_DY_TRP

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TRP	This register allows the Dynamic memory precharge command period (t_{RP}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (ARM: MPMCDytRP)

0x034 MPMC_DY_TRAS

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TRAS	This register allows the Dynamic memory precharge period (t_{RAS}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (ARM: MPMCDytRAS)

0x038 MPMC_DY_TSREX

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TSREX	This register allows the Dynamic memory self refresh exit time (t_{SREX}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (ARM: MPMCDytSREX)

0x03C MPMC_DY_TAPR

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TAPR	This register allows the Dynamic memory last data-out to Active command in case of autoprecharge read (to same bank) time (t_{APR}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (Arm: MPMCDytAPR)

0x040 MPMC_DY_TDAL

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TDAL	This register allows the Dynamic memory Data-In to Active command time (t_{DAL}or t_{APW}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (ARM: MPMCDytDAL)

0x044 MPMC_DY_TWR

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_DY_TWR	This register allows the Dynamic memory write recovery time or the Data-In to precharge time (t_{WR} or t_{DPL} or t_{RWL} or t_{RDL}) to be programmed. N/A: Reset value HRESETn 0xF: Reset value nPOR (ARM: MPMCDytWR)

0x048 MPMC_DY_TRC

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_DY_TRC	This register allows the Dynamic memory AutoRefresh and Active Command period to be programmed (t_{RC}). N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCDytRC

0x04C MPMC_DY_TRFC

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_DY_TRFC	This register allows the Dynamic memory AutoRefresh to Active Command Period (t_{RFC}) to be programmed. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCDytRFC)

0x050 MPMC_DY_TXSR

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_DY_TXSR	This register allows the Dynamic memory Self refresh exit to Active command time (t_{XSR}) to be programmed. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCDytXSR)

0x054 MPMC_DY_TRRD

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_DY_TRRD	This register allows the Dynamic memory Active Bank A to Active Bank B time (t_{RRD}) to be programmed. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCDytRRD)

0x058 MPMC_DY_TMRD

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_DY_TMRD	This register allows the Dynamic memory Load Mode register to Active Command time (t_{MRD}) to be programmed. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCDytMRD)

0x080 MPMC_ST_EXD_WT

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:0	MPMC_ST_EXD_WT	This register allows the Static memory Extended Wait Access time to be programmed. N/A: Reset value HRESETn 0xx000: Reset value nPOR (Arm: MPMCStExdWt)

0x100 MPMC_DY_CONFIG0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
29:0	MPMC_DY_CONFIG0	This register allows the configuration information for the SDRAM connected to chip select 4 to be programmed. N/A: Reset value HRESETn 0x000: Reset value nPOR (Reserved bits: 27, 25, 23, 21, 18:15, 13, 6:5, 2:0) (ARM: MPMCDyConfig0)

0x104 MPMC_DY_RAS_CAS0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:0	MPMC_DY_RAS_CAS0	This register allows the RAS and CAS latencies for the SDRAM connected to chip select 4 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 7:2) (ARM: MPMCDyRasCas0)

0x120 MPMC_DY_CONFIG1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
29:0	MPMC_DY_CONFIG1	This register allows the configuration information for the SDRAM connected to chip select 5 to be programmed. N/A: Reset value HRESETn 0x000: Reset value nPOR (Reserved bits: 27, 25, 23, 21, 18:15, 13, 6:5, 2:0) (ARM: MPMCDyConfig1)

0x124 MPMC_DY_RAS_CAS1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:0	MPMC_DY_RAS_CAS1	This register allows the RAS and CAS latencies for the SDRAM connected to chip select 5 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 7:2) (ARM: MPMCDyRasCas1)

0x140 MPMC_DY_CONFIG2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
29:0	MPMC_DY_CONFIG2	This register allows the configuration information for the SDRAM connected to chip select 6 to be programmed. N/A: Reset value HRESETn 0x000: Reset value nPOR (Reserved bits: 27, 25, 23, 21, 18:15, 13, 6:5, 2:0) (ARM: MPMCDyConfig2)

0x144 MPMC_DY_RAS_CAS2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:0	MPMC_DY_RAS_CAS2	This register allows the RAS and CAS latencies for the SDRAM connected to chip select 6 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 7:2) (ARM: MPMCDyRasCas2)

0x160 MPMC_DY_CONFIG3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
29:0	MPMC_DY_CONFIG3	This register allows the configuration information for the SDRAM connected to chip select 7 to be programmed. N/A: Reset value HRESETn 0x000: Reset value nPOR (Reserved bits: 27, 25, 23, 21, 18:15, 13, 6:5, 2:0) (ARM: MPMCDyConfig3)

0x164 MPMC_DY_RAS_CAS3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
9:0	MPMC_DY_RAS_CAS3	This register allows the RAS and CAS latencies for the SDRAM connected to chip select 7 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 7:2) (ARM: MPMCDyRasCas3)

0x200 MPMC_ST_CONFIG0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
20:0	MPMC_ST_CONFIG0	This register allows the configuration information for the static memory connected to chip select 0 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 18:9, 5:4, 2) (ARM: MPMCStConfig0)

0x204 MPMC_ST_W_TWEN0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TWEN0	This register allows the delay from the chip select assertion to the write enable assertion to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCStWtWen0)

0x208 MPMC_ST_W_TOEN0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TOEN0	This register allows the delay from the chip select assertion to the output enable assertion to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM:MPMCStWtOen0)

0x20C MPMC_ST_W_TRD0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TRD0	This register allows the delay from the chip select assertion to the read access to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtRd0)

0x210 MPMC_ST_W_TPG0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TPG0	This register allows the delay from the chip select assertion to the asynchronous page mode read access to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtPg0)

0x214 MPMC_ST_W_TWR0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TWR0	This register allows the delay from the chip select assertion to the write access to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtWr0)

0x218 MPMC_ST_W_TTURN0

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TTURN0	This register allows the number of bus turn around cycles to be programmed for the static memory connected to chip select 0. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtTurn0)

0x220 MPMC_ST_CONFIG1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
20:0	MPMC_ST_CONFIG1	This register allows the configuration information for the static memory connected to chip select 1 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 18:9, 5:4, 2) (ARM: MPMCStConfig1)

0x224 MPMC_ST_W_TWEN1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TWEN1	This register allows the delay from the chip select assertion to the write enable assertion to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCStWtWen1)

0x228 MPMC_ST_W_TOEN1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TOEN1	This register allows the delay from the chip select assertion to the output enable assertion to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM:MPMCStWtOen1)

0x22C MPMC_ST_W_TRD1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TRD1	This register allows the delay from the chip select assertion to the read access to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtRd1)

0x230 MPMC_ST_W_TPG1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TPG1	This register allows the delay from the chip select assertion to the asynchronous page mode read access to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtPg1)

0x234 MPMC_ST_W_TWR1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TWR1	This register allows the delay from the chip select assertion to the write access to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtWr1)

0x238 MPMC_ST_W_TTURN1

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TTURN1	This register allows the number of bus turn around cycles to be programmed for the static memory connected to chip select 1. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtTurn1)

0x240 MPMC_ST_CONFIG2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
20:0	MPMC_ST_CONFIG2	This register allows the configuration information for the static memory connected to chip select 2 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 18:9, 5:4, 2) (ARM: MPMCStConfig2)

0x244 MPMC_ST_W_TWEN2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TWEN2	This register allows the delay from the chip select assertion to the write enable assertion to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCStWtWen2)

0x248 MPMC_ST_W_TOEN2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TOEN2	This register allows the delay from the chip select assertion to the output enable assertion to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM:MPMCStWtOen2)

0x24C MPMC_ST_W_TRD2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TRD2	This register allows the delay from the chip select assertion to the read access to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtRd2)

0x250 MPMC_ST_W_TPG2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TPG2	This register allows the delay from the chip select assertion to the asynchronous page mode read access to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtPg2)

0x254 MPMC_ST_W_TWR2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TWR2	This register allows the delay from the chip select assertion to the write access to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtWr2)

0x258 MPMC_ST_W_TTURN2

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TTURN2	This register allows the number of bus turn around cycles to be programmed for the static memory connected to chip select 2. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtTurn2)

0x260 MPMC_ST_CONFIG3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
20:0	MPMC_ST_CONFIG3	This register allows the configuration information for the static memory connected to chip select 3 to be programmed. N/A: Reset value HRESETn 0x00: Reset value nPOR (Reserved bits: 18:9, 5:4, 2) (ARM: MPMCStConfig3)

0x264 MPMC_ST_W_TWEN3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TWEN3	This register allows the delay from the chip select assertion to the write enable assertion to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCStWtWen3)

0x268 MPMC_ST_W_TOEN3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
3:0	MPMC_ST_W_TOEN3	This register allows the delay from the chip select assertion to the output enable assertion to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x0: Reset value nPOR (ARM:MPMCStWtOen2)

0x26C MPMC_ST_W_TRD3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TRD3	This register allows the delay from the chip select assertion to the read access to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtRd3)

0x270 MPMC_ST_W_TPG3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TPG3	This register allows the delay from the chip select assertion to the asynchronous page mode read access to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtPg3)

0x274 MPMC_ST_W_TWR3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TWR3	This register allows the delay from the chip select assertion to the write access to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtWr3)

0x278 MPMC_ST_W_TTURN3

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
4:0	MPMC_ST_W_TTURN3	This register allows the number of bus turn around cycles to be programmed for the static memory connected to chip select 3. N/A: Reset value HRESETn 0x1F: Reset value nPOR (ARM: MPMCStWtTurn3)

0xF00 MPMCITCR

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
0	MPMCITCR	This one-bit register is used to select the various test modes. 0x0: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCITCR)

0xF20 MPMCITIP

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMCITIP	Control and read the following input lines: ■ MPMCSTCS0PO ■ MPMCSTCS1POL ■ MPMCSTCS2POL ■ MPMCSTCS3POL ■ MPMCSTCS1MW ■ MPMCBIGENDIAN ■ MPMCSREFREQ 0x0: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCITIP)

0xF40 MPMCITOP

Type: Read/Write
Clock: same rate as HCLK

Bits	Name	Description
1:0	MPMCITOP	Control and read the following output lines: ■ MPMCRPVHHOUT ■ MPMCSREFACK 0x00: Reset value HRESETn 0x00: Reset value nPOR (ARM: MPMCITOP)

0xFD0 MPMC_PERIPH_ID4

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID4	Peripheral ID register bits 39:32 0x1: Reset value HRESETn 0x1: Reset value nPOR (ARM: MPMCPeriphId4)

0xFD4 MPMC_PERIPH_ID5

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID5	Reserved (Peripheral ID register). 0x0: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCPeriphId5)

0xFD8 MPMC_PERIPH_ID6

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID6	Reserved (Peripheral ID register). 0x0: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCPeriphId6)

0xFDC MPMC_PERIPH_ID7

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID7	Reserved (Peripheral ID register). 0x0: Reset value HRESETn 0x0: Reset value nPOR (ARM: MPMCPeriphId7)

0xFE0 MPMC_PERIPH_ID0

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID0	Peripheral ID register. Bits 7:0 0x72: Reset value HRESETn 0x72: Reset value nPOR (ARM: MPMCPeriphId0)

0xFE4 MPMC_PERIPH_ID1

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID1	Peripheral ID register. Bits 15:8. 0x11: Reset value HRESETn 0x11: Reset value nPOR (ARM: MPMCPeriphId1)

0xFE8 MPMC_PERIPH_ID2

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID2	Peripheral ID register. Bits 23:16. 0x04: Reset value HRESETn 0x04: Reset value nPOR (ARM: MPMCPeriphId2)

0xFEC MPMC_PERIPH_ID3

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PERIPH_ID3	Peripheral ID register. Bits 31:24. 0xC2: Reset value HRESETn 0xC2: Reset value nPOR (ARM: MPMCPeriphId3)

0xFF0 MPMC_PCELL_ID0

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PCELL_ID0	PrimeCell ID register. Bits 7:0. 0x0D: Reset value HRESETn 0x0D: Reset value nPOR (ARM:MPMCPCellId0)

0xFF4 MPMC_PCELL_ID1

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PCELL_ID1	PrimeCell ID register. Bits 15:8. 0xF0: Reset value HRESETn 0xF0: Reset value nPOR (ARM: MPMCPCellId1)

0xFF8 MPMC_PCELL_ID2

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PCELL_ID2	PrimeCell ID register. Bits 23:16. 0x05: Reset value HRESETn 0x05: Reset value nPOR (ARM: MPMCPCellId2)

0xFFC MPMC_PCELL_ID3

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
7:0	MPMC_PCELL_ID3	PrimeCell ID register. Bits 31:24. 0xB1: Reset value HRESETn 0xB1: Reset value nPOR (ARM: MPMCPCellId3)

5 NAND Flash Memory Interface

5.1 Overview

These registers comprise one part of the total five blocks of AHB registers (AHB, EBI1, EBI2, NANDFlash, GRP).

5.2 ARM registers

An internal buffer starts at address 0 of the nand memory space, and the registers in this section follow that, beginning at 0x0300. Thus the offsets in this chapter begin at 0x0300. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: NAND
- Number of word addresses: 32
- NAND memory space: 0x60000000 - 0x67FFFFFF
- OFFSET = NAND_BASE+0x0
- NAND_BASE = 0x60000000
- nandaddr range = 9:2

0x0300 NAND_FLASH_CMD

Type: Write
Clock: HCLK

Send the operation command by writing to this register.

Bits	Name	Description
2:0	OP_CMD	An interrupt will be generated upon each operation, except on the soft-reset, which is done immediately. 000 : Software reset (of NAND flash controller . Ongoing operation will abort, NAND controller will be forced into idle, and the configure register is reset to power up default value.) 001 : page_read (Transfer one page of data from the NAND flash to the SRAM buffer. The NAND flash page address is provided by the NAND_FLASH_ADDR register.) 010 : flag_read (This command reads one byte from the spare area of the NAND flash. The page address and byte address are all provided by the NAND_FLASH_ADDR register. Some NAND flash makers, e.g., SAMSUNG, use the spare area to mask bad blocks upon shipping. After a flag_read operation, a reset operation is suggested before issuing a page_write command.) 011: page_write (Transfer one page of data from the SRAM buffer to the NAND flash. The page address is provided by the NAND_FLASH_ADDR register.) 100 : block_erase (Erase one block of NAND flash memory. The block address is provided by the NAND_FLASH_ADDR register.) 101: ID_fetch (Read the NAND flash maker ID and device ID. (This ID information can be used to determine the actual size of the NAND device.) 110: status_check (Check the NAND flash device status.) 111: reset NAND flash memory (This reset-operation is suggested upon power-up. This reset-operation is also suggested after a soft-reset command abort an ongoing operation.)

0x0304 NAND_FLASH_ADDR

Type: Write
Clock: HCLK

This register holds the NAND flash page address for a page read, page write, or a block erase operation. It also holds the address for a flag read operation.

Bits	Name	Description
31:9	NAND_FLASH_PAGE_ADDRESS	Depending on the actual size of the NAND flash memory chip, i.e., for an 8-MByte NAND flash chip, only bits 22:9 are valid, bits 31:23 are all ignored. For a small page size NAND device, these same sets of bits are used for the page address to be consistent, even though there are only 256 bytes in each page.
8:4	RESERVED_BITS8_4	Ignored.
3:0	SPARE_AREA_BYTE_ADDRESS	Valid only for a flag read operation For a small page size NAND device, only bits [2:0] are valid and we recommend programming bit 3 to 0 .

0x0308 NAND_FLASH_STATUS

Type: Read
Clock: HCLK

NAND_FLASH_STATUS

Bits	Name	Description
31	READ_ERROR	This bit is set (1) when an error is detected in the page of data that was just read in but could not be corrected by the ECC. ECC_self_err (see bit 6) will also cause this bit to be set if bit-7 of NAND_FLASH_CFG register is 0. If bit-7 of NAND_FLASH_CFG register is1, ECC_self_err will not set this bit. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
30:15	NAND_FLASH_ID	The result of ID_fetch CMD. Bits 30:23 contain the Maker ID. Bits 22:15 contain the Device ID
14	WRITE_PROTECT	NAND flash device status, the result of status_check CMD. The meaning of these bits was defined by NAND manufacturers. 1: Not protected. 0 : Protected (the NAND flash memory is read only).
13	READY_BUSY_N_STATUS	NAND flash device status, the result of status_check CMD. The meaning of this bit was defined by NAND manufacturers. 1 : Ready 0: Busy

NAND_FLASH_STATUS (Continued)

Bits	Name	Description
12:8	RESERVED_BITS12_8	NAND flash device status, the result of status_check CMD. The meaning of these bits was defined by NAND manufacturers. Reserved by the NAND flash device maker.
7	OP_RESULT	NAND flash device status, the result of status_check CMD. The meaning of this bit was defined by NAND manufacturers. 0: Last page write passed (The last page_write or block_erase operation passed. 1: Last page write failed (The last page_write or block_erase operation failed, and block replacement is suggested.)
6	ECC_SELF_ERR	This bit will be set (1) when an error is detected in the spare area of the page that was just read in, which means an ECC self bit-error. Theoretically, this could be a multiple-bit error also; however, most likely it is a 1-bit error with ECC self. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
5	READY_BUSY_N	0: External NAND flash busy 1: External NAND flash ready This is the signal connect to the Ready_Busy_n pin.
4	CORRECTABLE_ERROR	This bit will be set (1) when an error is detected in the page of data that was just read in and is automatically corrected by the ECC. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.

NAND_FLASH_STATUS (Continued)

Bits	Name	Description
3	OP_ERR	Reading 1 from this bit indicates there is some error concerning the last operation. There are two possibilities: Case 1: An operation command (other than a soft reset) can be issued only when the NAND flash controller is at idle op_status (see bit 2:0); otherwise, the command will be ignored and the operation error bit will be set. Case 2: To support a NAND flash device that requires the CE_n (chip select) signal to stay low during read busy, the external memory interface ports were held to prevent another access from affecting the NAND reading. A watch-dog was used to count for MAX 40 ~ 200 uS (for AHB clock frequency range from 10 ~ 100 MHz). If the NAND is still in read-busy (the ready_busy_n signal is low) when the counter expires, the read operation will be aborted and the NAND controller will be reset to power-up the initial state. An operation done interrupt will be generated and sent to the processor, the page read error bit (bit 31) will be set (1) and this bit will be set (1) also. This bit will be cleared (0) with a properly issued op_cmd.
2:0	OPERATION_STATUS	000 : Idle and waiting (for next operation command; powerup default). 001 : page read op ongoing (op = operation) 010 : flag read op ongoing 011 : page write op ongoing 100 : page erase op ongoing 101 : ID fetch op ongoing 110 : status check op ongoing 111: NAND flash reset (NAND flash memory being reset)

0x030C NAND_FLASH_ECC_0

Type: Read
Clock: HCLK

The value in this register is valid only when the last completed operation is either a page/read or a page/write with ECC enabled.

Bits	Name	Description
31	ECC_SELF_ERR	This bit will be set (1) when there is 1-bit difference between the newly calculated ECC and the one that was read back from the NAND spare area Theoretically, this could also be a multiple-bit error; however, most likely it is a 1-bit error with ECC self This bit is only valid when the last completed operation is a page/read with ECC enabled.
30	CORRECTION	Set (1) when an error is detected in the quarter-page of data that was just read in and is automatically corrected by the ECC. This bit is valid only when the last completed operation is a page/read with ECC enabled.
29	UN_CORRECTABLE_ERROR	This bit will be set (1) when an error is detected in the just read-in quarter page of data that was not able to be corrected by the ECC. ECC_self_err will also cause this bit to be set if bit-7 of the NAND_FLASH_CFG register is 0. If bit-7 of NAND_FLASH_CFG register is1, ECC_self_err will not set this bit. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
28:20	RESERVED_BITS28_20	
19:0	ECC	Error correction code for bytes 0 to 127 in current page

0x0310 NAND_FLASH_ECC_1

Type: Read
Clock: HCLK

The value in this register is valid only when last completed operation is either a page/read or page/write with ECC enabled.

Bits	Name	Description
31	ECC_SELF_ERR	This bit will be set (1) when there is 1-bit difference between the newly calculated ECC and the one that was read back from the NAND spare area. Theoretically, this could also be multiple-bit error; however, most likely it is a 1-bit error with ECC self. This bit is only valid when the last completed operation is a page/read with ECC enabled.
30	CORRECTION	Set (1) when an error is detected in the quarter-page of data that was just read in and is automatically corrected by the ECC. This bit is valid only when the last completed operation is a page/read with ECC enabled.
29	UN_CORRECTABLE_ERROR	This bit will be set (1) when an error is detected in the just read-in quarter page of data that was not able to be corrected by the ECC. ECC_self_err will also cause this bit to be set if bit-7 of the NAND_FLASH_CFG register is 0; If bit-7 of NAND_FLASH_CFG register is1, ECC_self_err will not set this bit. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
28:20	RESERVED_BITS28_20	
19:0	ECC	Error correction code for bytes 128 to 255 in the current page.

0x0314 NAND_FLASH_ECC_2

Type: Read
Clock: HCLK

The value in this register is valid only when last completed operation is either a page/read or page/write with ECC enable.

Bits	Name	Description
31	ECC_SELF_ERR	This bit will be set (1) when there is 1-bit difference between the newly calculated ECC and the one that was read back from the NAND spare area. Theoretically, this could also be multiple-bit error; however, most likely it is a 1-bit error with ECC self. This bit is only valid when the last completed operation is a page/read with ECC enabled.
30	CORRECTION	Set (1) when an error is detected in the quarter-page of data that was just read in and is automatically corrected by the ECC. This bit is valid only when the last completed operation is a page/read with ECC enabled.
29	UN_CORRECTABLE_ERROR	This bit will be set (1) when an error is detected in the just read-in quarter page of data that was not able to be corrected by the ECC. ECC_self_err will also cause this bit to be set if bit-7 of the NAND_FLASH_CFG register is 0; If bit-7 of NAND_FLASH_CFG register is1, ECC_self_err will not set this bit. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
28:20	RESERVED_BITS28_20	
19:0	ECC	Error correction code for bytes 256 to 383 in current page.

0x0318 NAND_FLASH_ECC_3

Type: Read
Clock: HCLK

The value in this register is valid only when last completed operation is either a page/read or page/write with ECC enabled.

Bits	Name	Description
31	ECC_SELF_ERR	This bit will be set (1) when there is 1-bit difference between the newly calculated ECC and the one that was read back from the NAND spare area. Theoretically, this could also be multiple-bit error; however, most likely it is a 1-bit error with ECC self. This bit is only valid when the last completed operation is a page/read with ECC enabled.
30	CORRECTION	Set (1) when an error is detected in the quarter-page of data that was just read in and is automatically corrected by the ECC. This bit is valid only when the last completed operation is a page/read with ECC enabled.
29	UN_CORRECTABLE_ERROR	This bit will be set (1) when an error is detected in the just read-in quarter page of data that was not able to be corrected by the ECC. ECC_self_err will also cause this bit to be set if bit-7 of the NAND_FLASH_CFG register is 0; If bit-7 of NAND_FLASH_CFG register is1, ECC_self_err will not set this bit. This bit is valid only when the last successfully completed operation is a page read with ECC enabled.
28:20	RESERVED_BITS28_20	
19:0	ECC	Error correction code for bytes 384 to 512 in current page.

0x031C NAND_FLASH_CFG1

Type: Write
Clock: HCLK

NAND_FLASH_CFG1

Bits	Name	Description
10:8	NAND_RECOVER_CYCLES	When a read from a NAND device is finished and the CE-n is deasserted, the NAND flash will typically continue to drive the data bus for 20 ns before it enters a High-Z state. A recover cycle must be inserted before the next access of other memory start, to avoid the double-drive power consumption issue. 00: No recover cycle insert 01: 1 recover cycle insert 10: 2 recover cycle inserts 11: 3 recover cycle inserts
7	ECC_SELF_ERROR_DETECT	When clear (0), detected ECC self 1-bit error causes the corresponding un_correctable_err bit to be set. When set (1), the detected ECC self 1-bit error does NOT cause the corresponding un_correctable_err bit to be set.
6	BUFFER_MEM_WRITE_WAIT	When clear (0), internal buffer SRAM access (R/W) with1 AHB clock (default). When set (1), Internal buffer SRAM write (needs two AHB clocks).
5	WIDE_NAND	When clear (0), the controller assumes that the NAND flash I/O is 8-bit wide. Default. When set (1), the controller assumes that the NAND flash I/O is 16-bit wide.
4	RESERVED_BIT4	
3	CTLR_CLK_HALT_DIS	When clear (0), the HCLK going to most of the registers in the controller, except 160 ECC registers, is gated off by a CGC cell unless the controller is performing an access to the NAND flash. When set (1), this CGC never gates off the HCLK. 0: Ctlr clk halt enabled 1: Ctlr clk halt disabled
2	ECC_CLK_HALT_DIS	When clear (0), the HCLK going to 160 ECC registers, are gated off by a CGC cell except once every quarter page and once per spare data transferred during a page read or a page write command. When set (1), this CGC never gates off the HCLK. 0: ECC clk halt enabled 1: ECC clk halt disabled

NAND_FLASH_CFG1 (Continued)

Bits	Name	Description
1	BUSFREE_SUPPORT_SELECT	When clear (0), support only NAND devices that do not require CE_n active low during read/busy. When (set), support also NAND devices that require CE_n active low during read/bu (default).
0	ECC_DISABLE	0: ECC enabled (error correction coding). Default. 1: ECC disabled.

0x0320 NAND_FLASH_CFG2

Type: Write
Clock: HCLK

This register controls the NAND interface wait state by specifying the width of high pulse and low pulse of the WE_N and OE_N signals in number of HCLK cycles.

For each write transfer, the WE_N signal is divided into 3 segments, namely wr_setup (high pulse), wr_active (low pulse) and wr_hold (high pulse).

For each read transfer, the OE_N is divided into 2 segments, namely rd_setup (high pulse) and rd_active (low pulse).

NAND_FLASH_CFG2

Bits	Name	Description
29:25	WR_SETUP	Write setup. This field specifies the length of the first segment of a write transfer. For example, in a single write transfer, this corresponds to the period from the time that CS_N goes low to the time that WE_N goes low. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x0.
24:20	WR_ACTIVE	Write active. This field specifies the length of the second segment of a write transfer, i.e., from the time that WE_N goes low to the time that WE_N goes high. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x0.

NAND_FLASH_CFG2 (Continued)

Bits	Name	Description
19:15	WR_HOLD	Write hold. This field specifies the length of the third segment of a write transfer. For example, in a single write transfer, this corresponds to the period from the time that WE_N goes high to the time that CS_N goes high. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x0.
14:10	RD_ACTIVE	Read active. This field specifies the length of the second segment of a read transfer, i.e., from the time OE_N goes low to the time OE_N goes high. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x0.
9:5	RD_SETUP	Read setup. This field applies to ANY read transfers that are NOT caused by the ID_fetch command. It specifies the length of the first segment of the read transfer. For example, in a single read transfer, this corresponds to the period from the time CS_N goes low to the time OE_N goes low. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x0.
4:0	ID_RD_SETUP	ID read setup. This field only applies to read transfers caused by the ID_fetch command. It specifies the length of the first segment of the read transfer. For example, in a single read transfer, this corresponds to the period from the time CS_N goes low to the time OE_N goes low. This is a zero-based value. For example, 0 means 1 HCLK cycle. Power-on default is 0x3.

0x0324 NAND_SPARE_DATA

Type: Read
Clock: HCLK

The result of Flag_read operation.

Bits	Name	Description
15:0	SPARE_DATA	The Flag_read command will read only 1 column address from the NAND-spare area. This register holds the data read back from the device, which can be a byte or a 16-bit word, depending on the I/O width of the device. For 8-bit I/O devices, only bits [7:0] contain valid data. For 16-bit I/O devices, all 16 bits contain valid data.

6 Interrupt Controller

6.1 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: INTCTL
- Byte Address: 0x0900 - 0x09FC (inside the web address space: 0x700 - 0x9FC)
- OFFSET=CHIP_BASE+0x0900
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

6.1.1 Interrupt controller write registers

0x0000 INT_CLEAR_0

Type: Write
Clock: HCLK

This register is used to clear the register bits of the interrupt status register. If a bit in the interrupt clear register is set (1), the corresponding bit in the interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

INT_CLEAR_0

Bits	Name	Description
31	RTC_ROLLOVER_INT	Set (1) to clear status bit of RTC_ROLLOVER_INT
30	SLP_XTAL_TIMETICK_INT	Set (1) to clear status bit of SLP_XTAL_TIMETICK_INT
29	MDSP_INT2	Set (1) to clear status bit of MDSP_INT[2]
28	MDSP_INT1	Set (1) to clear status bit of MDSP_INT[1]
27	MDSP_INT0	Set (1) to clear status bit of MDSP_INT[0]

INT_CLEAR_0 (Continued)

Bits	Name	Description
26	OFFLINE_DONE_INT	Set (1) to clear status bit of OFFLINE_DONE_INT
25	RXCHIPX8_ENABLE_INT	Set (1) to clear status bit of RXCHIPX8_ENABLE_INT
24	AUX_CODEC_ONE_INT	Set (1) to clear status bit of AUX_CODEC_ONE_INT
23	DEM_SEARCH_DMA_REQ	Set (1) to clear status bit of DEM_SEARCH_DMA_REQ
22	QUICK_PAGING_INT	Set (1) to clear status bit of DEMOD_INT
21	DEM_SEARCHER_DONE_INT	Set (1) to clear status bit of DEM_SEARCHER_DONE_INT
20	DEC_DONE_STROBE	Set (1) to clear status bit of DEC_DONE_STROBE
19	MOD_TX_FR_INT	Set (1) to clear status bit of MOD_TX_FR_INT
18	MOD_TX_1_25MS_INT	Set (1) to clear status bit of MOD_TX_1_25MS_INT
17	SDCC_0_INT	Set (1) to clear status bit of SDCC_0_INT
16	SDCC_1_INT	Set (1) to clear status bit of SDCC_1_INT
15	RESERVED_BIT15	
14	GSM_MICRO_IRQ	Set (1) to clear status bit of GSM_MICRO_IRQ
13	GSM_GSAC0_DONE_IRQ	Set (1) to clear status bit of GSM_GSAC0_DONE_IRQ
12	GSM_GSAC1_DONE_IRQ	Set (1) to clear status bit of GSM_TIME_INT2
11	HDR_WAKEUP_INT	Set (1) to clear status bit of HDR_WAKEUP_INT
10	HDR_SLEEP_ENDING_INT	Set (1) to clear status bit of HDR_SLEEP_ENDING_INT
9	SLOT_INT	Set (1) to clear status bit of SLOT_INT
8	GSM_SLEEP_INT	Set (1) to clear status bit of GSM_SLEEP_INT
7	SLP_XTAL_TIMER_INT	Set (1) to clear status bit of SLP_XTAL_TIMER_INT
6	SLEEP_FEE_INT	Set (1) to clear status bit of SLEEP_FEE_INT
5	WAKEUP_1X_INT	Set (1) to clear status bit of WAKEUP_1X_INT
4	DEM_TIME_INT2	Set (1) to clear status bit of DEM_TIME_INT2
3	DEM_TIME_INT1	Set (1) to clear status bit of DEM_TIME_INT1
2	TIME_TICK_INT	Set (1) to clear status bit of TIME_TICK_INT
1	DEM_26MS_INT	Set (1) to clear status bit of DEM_26MS_INT. DEM_26MS_INT is PN_ROLLOVER_1X delayed by 128PN chips.
0	EVEN_SEC_INT	Set (1) to clear status bit of EVEN_SEC_INT

0x0004 INT_CLEAR_1

Type: Write
Clock: HCLK

This register is used to clear the register bits of the interrupt status register. If a bit in the interrupt clear register is set (1), the corresponding bit in the interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

INT_CLEAR_1

Bits	Name	Description
31	GPIO_GROUP_INT	Set (1) to clear status bit of GPIO_GROUP_INT
30	GPIO2_GROUP_INT	Set (1) to clear status bit of GPIO2_GROUP_INT
29	MDDI_HOST_INT	Set (1) to clear status bit of MDDI_HOST_INT
28	MDDI_CLIENT_INT	Set (1) to clear status bit of MDDI_CLIENT_INT
27	DEC_OB_IRQ	Set (1) to clear status bit of DEC_OB_IRQ
26	DDO_FULL_IRQ	Set (1) to clear status bit of DDO_FULL_IRQ
25	DDI_READY_IRQ	Set (1) to clear status bit of DDI_READY_IRQ
24	USB_OTG_INT	Set (1) to clear status bit of USB_OTF_INT
23	COMPARE2_INT	Set (1) to clear status bit of COMPARE2_INT
22	COMPARE1_INT	Set (1) to clear status bit of COMPARE1_INT
21	PER_SLOT_INT	Set (1) to clear status bit of PER_SLOT_INT
20	BT_WAKEUP	Set (1) to clear status bit of BT_WAKEUP
19	ADSP_INT2	Set (1) to clear status bit of ADSP_INT[2]
18	ADSP_INT1	Set (1) to clear status bit of ADSP_INT[1]
17	ADSP_INT0	Set (1) to clear status bit of ADSP_INT[0]
16	GRAPHICS_INT	Set (1) to clear status bit of GRAPHICS_INT
15	MDP_INT	Set (1) to clear status bit of MDP_INT
14	NAND_WR_ER_DONE_INT	Set (1) to clear status bit of NAND_WR_ER_DONE_INT
13	NAND_OP_DONE_INT	Set (1) to clear status bit of GPIO_GROUP_INT
12	MMCC_ERR_INT	Set (1) to clear status bit of MMCC_ERR_INT
11	MMCC_INT	Set (1) to clear status bit of MMCC_INT
10	KEYSENSE_GROUP_INT	Set (1) to clear status bit of KEYSENSE_GROUP_INT
9	PN_ROLLOVER_1X	Set (1) to clear status bit of PN_ROLLOVER_1X
8	SBI1_INT	Set (1) to clear status bit of SBI1_INT

INT_CLEAR_1 (Continued)

Bits	Name	Description
7	SBI0_INT	Set (1) to clear status bit of SBI0_INT
6	I2CC_INT	Set (1) to clear status bit of I2CC_INT
5	UART3_DP_RX_DATA_INT	Set (1) to clear status bit of UART3_DP_RX_DATA_INT
4	UART2_DP_RX_DATA_INT	Set (1) to clear status bit of UART2_DP_RX_DATA_INT
3	UART1_DP_RX_DATA_INT	Set (1) to clear status bit of UART1_DP_RX_DATA_INT
2	UART3_INT	Set (1) to clear status bit of UART3_INT
1	UART2_INT	Set (1) to clear status bit of UART2_INT
0	UART1_INT	Set (1) to clear status bit of UART1_INT

0x0008 GPIO_INT_CLEAR_0

Type: Write
Clock: HCLK

This register is used to clear the register bits of the interrupt status register. If a bit in the interrupt clear register is set (1), the corresponding bit in the interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

Bits	Name	Description
31:0	GPIO_INT[31:0]	Set (1) to clear GPIO[31:0] interrupt status bit.

0x000C GPIO_INT_CLEAR_1

Type: Write
Clock: HCLK

This register is used to clear the register bits of the interrupt status register. If a bit in the interrupt clear register is set (1), the corresponding bit in the interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

Bits	Name	Description
21:0	GPIO_INT[53:32]	Set (1) to clear GPIO[81:67,38:32] interrupt status bit

0x0010 GPIO_INT_CLEAR_4

Type: Write
Clock: HCLK

This register is used to clear the register bits of the interrupt status register. If a bit in the interrupt clear register is set (1), the corresponding bit in the interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

Bits	Name	Description
24:0	GPIO_INT[78:54]	Set (1) to clear GPIO[106:82] interrupt status bit

6.1.2 Interrupt controller read/write registers

0x0014 IRQ_EN_0

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable an interrupt source that is going to the IRQ interrupt of the ARM core. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in the register is set (1).

IRQ_EN_0

Bits	Name	Description
31	RTC_ROLLOVER_INT	Set(1) to enable RTC_ROLLOVER_INT
30	SLP_XTAL_TIMETICK_INT	Set(1) to enable SLP_XTAL_TIMETICK_INT
29	MDSP_INT2	Set(1) to enable MDSP_INT[2]
28	MDSP_INT1	Set(1) to enable MDSP_INT[1]
27	MDSP_INT0	Set(1) to enable MDSP_INT[0]
26	OFFLINE_DONE_INT	Set(1) to enable OFFLINE_DONE_INT
25	RXCHIPX8_ENABLE_INT	Set(1) to enable RXCHIPX8_ENABLE_INT
24	AUX_CODEC_ONE_INT	Set(1) to enable AUX_CODEC_ONE_INT
23	DEM_SEARCH_DMA_REQ	Set(1) to enable DEM_SEARCH_DMA_REQ
22	QUICK_PAGING_INT	Set(1) to enable DEMOD_INT
21	DEM_SEARCHER_DONE_INT	Set(1) to enable DEM_SEARCHER_DONE_INT
20	DEC_DONE_STROBE	Set(1) to enable DEC_DONE_STROBE

IRQ_EN_0 (Continued)

Bits	Name	Description
19	MOD_TX_FR_INT	Set(1) to enable MOD_TX_FR_INT
18	MOD_TX_1_25MS_INT	Set(1) to enable MOD_TX_1_25MS_INT
17	SDCC_0_INT	Set(1) to enable SDCC_0_INT
16	SDCC_1_INT	Set(1) to enable SDCC_1_INT
15	RESERVED_BIT15	
14	GSM_MICRO_IRQ	Set(1) to enable GSM_MICRO_IRQ
13	GSM_GSAC0_DONE_IRQ	Set(1) to enable GSM_GSAC0_DONE_IRQ
12	GSM_GSAC1_DONE_IRQ	Set(1) to enable GSM_TIME_INT2
11	HDR_WAKEUP_INT	Set(1) to enable HDR_WAKEUP_INT
10	HDR_SLEEP_ENDING_INT	Set(1) to enable HDR_SLEEP_ENDING_INT
9	SLOT_INT	Set(1) to enable SLOT_INT
8	GSM_SLEEP_INT	Set(1) to enable GSM_SLEEP_INT
7	SLP_XTAL_TIMER_INT	Set(1) to enable SLP_XTAL_TIMER_INT
6	SLEEP_FEE_INT	Set(1) to enable SLEEP_FEE_INT
5	WAKEUP_1X_INT	Set(1) to enable WAKEUP_1X_INT
4	DEM_TIME_INT2	Set(1) to enable DEM_TIME_INT2
3	DEM_TIME_INT1	Set(1) to enable DEM_TIME_INT1
2	TIME_TICK_INT	Set(1) to enable TIME_TICK_INT
1	DEM_26MS_INT	Set(1) to enable DEM_26MS_INT
0	EVEN_SEC_INT	Set(1) to enable EVEN_SEC_INT

0x0018 IRQ_EN_1

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable an interrupt source that is going to the IRQ interrupt of the ARM core. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in the register is set.

IRQ_EN_1

Bits	Name	Description
31	GPIO_GROUP_INT	Set(1) to enable GPIO_GROUP_INT
30	GPIO2_GROUP_INT	Set(1) to enable GPIO2_GROUP_INT
29	MDDI_HOST_INT	Set(1) to enable MDDI_HOST_INT
28	MDDI_CLIENT_INT	Set(1) to enable MDDI_CLIENT_INT
27	DEC_OB_IRQ	Set(1) to enable DEC_OB_IRQ
26	DDO_FULL_IRQ	Set(1) to enable DDO_FULL_IRQ
25	DDI_READY_IRQ	Set(1) to enable DDI_READY_IRQ
24	USB_OTG_INT	Set(1) to enable USB_OTF_INT
23	COMPARE2_INT	Set(1) to enable COMPARE2_INT
22	COMPARE1_INT	Set(1) to enable COMPARE1_INT
21	PER_SLOT_INT	Set(1) to enable PER_SLOT_INT
20	BT_WAKEUP	Set(1) to enable BT_WAKEUP
19	ADSP_INT2	Set(1) to enable ADSP_INT[2]
18	ADSP_INT1	Set(1) to enable ADSP_INT[1]
17	ADSP_INT0	Set(1) to enable ADSP_INT[0]
16	GRAPHICS_INT	Set(1) to enable GRAPHICS_INT
15	MDP_INT	Set(1) to enable MDP_INT
14	NAND_WR_ER_DONE_INT	Set(1) to enable NAND_WR_ER_DONE_INT
13	NAND_OP_DONE_INT	Set(1) to enable GPIO_GROUP_INT
12	MMCC_ERR_INT	Set(1) to enable MMCC_ERR_INT
11	MMCC_INT	Set(1) to enable MMCC_INT
10	KEYSENSE_GROUP_INT	Set(1) to enable KEYSENSE_GROUP_INT

IRQ_EN_1 (Continued)

Bits	Name	Description
9	PN_ROLLOVER_1X	Set(1) to enable PN_ROLLOVER_1X
8	SBI1_INT	Set(1) to enable SBI1_INT
7	SBI0_INT	Set(1) to enable SBI0_INT
6	I2CC_INT	Set(1) to enable I2CC_INT
5	UART3_DP_RX_DATA_INT	Set(1) to enable UART3_DP_RX_DATA_INT
4	UART2_DP_RX_DATA_INT	Set(1) to enable UART2_DP_RX_DATA_INT
3	UART1_DP_RX_DATA_INT	Set(1) to enable UART1_DP_RX_DATA_INT
2	UART3_INT	Set(1) to enable UART3_INT
1	UART2_INT	Set(1) to enable UART2_INT
0	UART1_INT	Set(1) to enable UART1_INT

0x001C FIQ_EN_0

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable an interrupt source that is going to the FIQ interrupt of the ARM core. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

FIQ_EN_0

Bits	Name	Description
31	RTC_ROLLOVER_INT	Set(1) to enable RTC_ROLLOVER_INT
30	SLP_XTAL_TIMETICK_INT	Set(1) to enable SLP_XTAL_TIMETICK_INT
29	MDSP_INT2	Set(1) to enable MDSP_INT[2]
28	MDSP_INT1	Set(1) to enable MDSP_INT[1]
27	MDSP_INT0	Set(1) to enable MDSP_INT[0]
26	OFFLINE_DONE_INT	Set(1) to enable OFFLINE_DONE_INT
25	RXCHIPX8_ENABLE_INT	Set(1) to enable RXCHIPX8_ENABLE_INT
24	AUX_CODEC_ONE_INT	Set(1) to enable AUX_CODEC_ONE_INT
23	DEM_SEARCH_DMA_REQ	Set(1) to enable DEM_SEARCH_DMA_REQ
22	QUICK_PAGING_INT	Set(1) to enable DEMOD_INT

FIQ_EN_0 (Continued)

Bits	Name	Description
21	DEM_SEARCHER_DONE_INT	Set(1) to enable DEM_SEARCHER_DONE_INT
20	DEC_DONE_STROBE	Set(1) to enable DEC_DONE_STROBE
19	MOD_TX_FR_INT	Set(1) to enable MOD_TX_FR_INT
18	MOD_TX_1_25MS_INT	Set(1) to enable MOD_TX_1_25MS_INT
17	SDCC_0_INT	Set(1) to enable SDCC_0_INT
16	SDCC_1_INT	Set(1) to enable SDCC_1_INT
15	RESERVED_BIT15	
14	GSM_MICRO_IRQ	Set(1) to enable GSM_MICRO_IRQ
13	GSM_GSAC0_DONE_IRQ	Set(1) to enable GSM_GSAC0_DONE_IRQ
12	GSM_GSAC1_DONE_IRQ	Set(1) to enable GSM_TIME_INT2
11	HDR_WAKEUP_INT	Set(1) to enable HDR_WAKEUP_INT
10	HDR_SLEEP_ENDING_INT	Set(1) to enable HDR_SLEEP_ENDING_INT
9	SLOT_INT	Set(1) to enable SLOT_INT
8	GSM_SLEEP_INT	Set(1) to enable GSM_SLEEP_INT
7	SLP_XTAL_TIMER_INT	Set(1) to enable SLP_XTAL_TIMER_INT
6	SLEEP_FEE_INT	Set(1) to enable SLEEP_FEE_INT
5	WAKEUP_1X_INT	Set(1) to enable WAKEUP_1X_INT
4	DEM_TIME_INT2	Set(1) to enable DEM_TIME_INT2
3	DEM_TIME_INT1	Set(1) to enable DEM_TIME_INT1
2	TIME_TICK_INT	Set(1) to enable TIME_TICK_INT
1	DEM_26MS_INT	Set(1) to enable DEM_26MS_INT
0	EVEN_SEC_INT	Set(1) to enable EVEN_SEC_INT

0x0020 FIQ_EN_1

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable an interrupt source that is going to the FIQ interrupt of the ARM core. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

FIQ_EN_1

Bits	Name	Description
31	GPIO_GROUP_INT	Set(1) to enable GPIO_GROUP_INT
30	GPIO2_GROUP_INT	Set(1) to enable GPIO2_GROUP_INT
29	MDDI_HOST_INT	Set(1) to enable MDDI_HOST_INT
28	MDDI_CLIENT_INT	Set(1) to enable MDDI_CLIENT_INT
27	DEC_OB_IRQ	Set(1) to enable DEC_OB_IRQ
26	DDO_FULL_IRQ	Set(1) to enable DDO_FULL_IRQ
25	DDI_READY_IRQ	Set(1) to enable DDI_READY_IRQ
24	USB_OTG_INT	Set(1) to enable USB_OTF_INT
23	COMPARE2_INT	Set(1) to enable COMPARE2_INT
22	COMPARE1_INT	Set(1) to enable COMPARE1_INT
21	PER_SLOT_INT	Set(1) to enable PER_SLOT_INT
20	BT_WAKEUP	Set(1) to enable BT_WAKEUP
19	ADSP_INT2	Set(1) to enable ADSP_INT[2]
18	ADSP_INT1	Set(1) to enable ADSP_INT[1]
17	ADSP_INT0	Set(1) to enable ADSP_INT[0]
16	GRAPHICS_INT	Set(1) to enable GRAPHICS_INT
15	MDP_INT	Set(1) to enable MDP_INT
14	NAND_WR_ER_DONE_INT	Set(1) to enable NAND_WR_ER_DONE_INT
13	NAND_OP_DONE_INT	Set(1) to enable GPIO_GROUP_INT
12	MMCC_ERR_INT	Set(1) to enable MMCC_ERR_INT
11	MMCC_INT	Set(1) to enable MMCC_INT

FIQ_EN_1 (Continued)

Bits	Name	Description
10	KEYSENSE_GROUP_INT	Set(1) to enable KEYSENSE_GROUP_INT
9	PN_ROLLOVER_1X	Set(1) to enable PN_ROLLOVER_1X
8	SBI1_INT	Set(1) to enable SBI1_INT
7	SBI0_INT	Set(1) to enable SBI0_INT
6	I2CC_INT	Set(1) to enable I2CC_INT
5	UART3_DP_RX_DATA_INT	Set(1) to enable UART3_DP_RX_DATA_INT
4	UART2_DP_RX_DATA_INT	Set(1) to enable UART2_DP_RX_DATA_INT
3	UART1_DP_RX_DATA_INT	Set(1) to enable UART1_DP_RX_DATA_INT
2	UART3_INT	Set(1) to enable UART3_INT
1	UART2_INT	Set(1) to enable UART2_INT
0	UART1_INT	Set(1) to enable UART1_INT

0x0024 GPIO_INT_EN_0

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable GPIO interrupt sources. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

Bits	Name	Description
31:0	GPIO_INT[31:0]	Set (1) to enable GPIO[31:0] interrupt status bit.

0x0028 GPIO_INT_EN_1

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable GPIO interrupt sources. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

Bits	Name	Description
21:0	GPIO_INT[53:32]	Set (1) to enable GPIO[81:67,38:32] interrupt status bit.

0x002C GPIO_INT_EN_4

Type: Read/Write
Clock: HCLK
Reset State: 0x0

This register is used to enable GPIO interrupt sources. When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

Bits	Name	Description
24:0	GPIO_INT[78:54]	Set (1) to enable GPIO[106:82] interrupt status bit.

0x0030 INT_POLARITY_0

Type: Read/Write
Clock: HCLK
Reset State: 0x0

The INT_POLARITY_0 register controls the polarity for GPIO[31:0] interrupts. These bits must be initialized before its associated interrupts can be enabled.

Bits	Name	Description
31:0	GPIO_INT[31:0]	Polarity value for GPIO[31:0] interrupt.

0x0034 INT_POLARITY_1

Type: Read/Write
Clock: HCLK

The INT_POLARITY_1 register controls the polarity for GPIO[81:61] interrupts and GPIO[38:32] interrupts. These bits must be initialized before its associated interrupts can be enabled.

Bits	Name	Description
21:0	GPIO_INT[53:32]	Polarity value for GPIO[81:67,38:32] interrupt.

0x0038 INT_POLARITY_2

Type: Read/Write
Clock: HCLK
Reset State: 0x0

The INT_POLARITY_2 register controls the polarity for UART_DP_RX_DATA interrupts. These bits must be initialized before its associated interrupts can be enabled.

Bits	Name	Description
2	UART3_DP_RX_DATA	Polarity value for UART3 dp_rx_data interrupt.
1	UART2_DP_RX_DATA	Polarity value for UART2 dp_rx_data interrupt.
0	UART1_DP_RX_DATA	Polarity value for UART1 dp_rx_data interrupt.

0x003C INT_POLARITY_5

Type: Read/Write
Clock: HCLK
Reset State: 0x0

The INT_POLARITY_5 register controls the polarity for GPIO[106:82] interrupts. These bits must be initialized before its associated interrupts can be enabled.

Bits	Name	Description
24:0	GPIO_INT[78:54]	Polarity value for GPIO[106:82] interrupt.

0x0048 GPIO_INT_DETECT_CTL_0

Type: Read/Write
Clock: HCLK
Reset State: 0x0

Each bit controls the interrupt triggering mechanism: level detection, or edge detection. Setting the specific bit to ‘1’ will select edge detection for interrupt triggering, setting the specific bit to ‘0’ will select level match (with POLARITY register values) for interrupts.

Bits	Name	Description
31:0	GPIO_INT[31:0]	Set (1) to edge detect GPIO[31:0] interrupt.

0x004C GPIO_INT_DETECT_CTL_1

Type: Read/Write
Clock: HCLK

Each bit controls the interrupt triggering mechanism: level detection, or edge detection. Setting the specific bit to ‘1’ will select edge detection for interrupt triggering, setting the specific bit to ‘0’ will select level match (with POLARITY register values) for interrupts.

Bits	Name	Description
21:0	GPIO_INT[53:32]	Set (1) to edge detect GPIO[81:67,38:32] interrupt.

0x0050 GPIO_INT_DETECT_CTL_4

Type: Read/Write
Clock: HCLK
Reset State: 0x0

Each bit controls the interrupt triggering mechanism: level detection, or edge detection. Setting the specific bit to ‘1’ will select edge detection for interrupt triggering, setting the specific bit to ‘0’ will select level match (with POLARITY register values) for interrupts.

Bits	Name	Description
24:0	GPIO_INT[78:54]	Set (1) to edge detect GPIO[106:82] interrupt.

6.1.3 Interrupt controller read registers

0x0054 INT_STATUS_0

Type: Read
Clock: HCLK

Each bit of the interrupt status register corresponds to a particular interrupt source. If any of these bits are set, an interrupt is asserted by the associated interrupt source. All of the bits in this register are sticky. This means that once a bit is set in this register, the source of the interrupt request must be reset, and the corresponding bit in the corresponding interrupt clear register must be set in order to clear these bits (in the said order).

INT_STATUS_0

Bits	Name	Description
31	RTC_ROLL_INT	The status bit value for interrupt source RTC_ROLL_INT
30	SLP_XTAL_TIMETICK_INT	The status bit value for interrupt source SLP_XTAL_TIMETICK_INT
29	MDSP_INT2	The status bit value for interrupt source MDSP_INT[2]
28	MDSP_INT1	The status bit value for interrupt source MDSP_INT[1]
27	MDSP_INT0	The status bit value for interrupt source MDSP_INT[0]
26	OFFLINE_DONE_INT	The status bit value for interrupt source OFFLINE_DONE_INT
25	RXCHIPX8_ENABLE_INT	The status bit value for interrupt source RXCHIPX8_ENABLE_INT
24	AUX_CODEC_ONE_INT	The status bit value for interrupt source AUX_CODEC_ONE_INT
23	DEM_SEARCH_DMA_REQ	The status bit value for interrupt source DEM_SEARCH_DMA_REQ
22	QUICK_PAGING_INT	The status bit value for interrupt source DEMOD_INT
21	DEM_SEARCHER_DONE_INT	The status bit value for interrupt source DEM_SEARCHER_DONE_INT
20	DEC_DONE_STROBE	The status bit value for interrupt source DEC_DONE_STROBE
19	MOD_TX_FR_INT	The status bit value for interrupt source MOD_TX_FR_INT
18	MOD_TX_1_25MS_INT	The status bit value for interrupt source MOD_TX_1_25MS_INT
17	SDCC_0_INT	The status bit value for interrupt source SDCC_0_INT
16	SDCC_1_INT	The status bit value for interrupt source SDCC_1_INT
15	RESERVED_BIT15	
14	GSM_MICRO_IRQ	The status bit value for interrupt source GSM_MICRO_IRQ
13	GSM_GSAC0_DONE_IRQ	The status bit value for interrupt source GSM_GSAC0_DONE_IRQ

INT_STATUS_0 (Continued)

Bits	Name	Description
12	GSM_GSAC1_DONE_IRQ	The status bit value for interrupt source GSM_TIME_INT2
11	HDR_WAKEUP_INT	The status bit value for interrupt source HDR_WAKEUP_INT
10	HDR_SLEEP_ENDING_INT	The status bit value for interrupt source HDR_SLEEP_ENDING_INT
9	SLOT_INT	The status bit value for interrupt source SLOT_INT
8	GSM_SLEEP_INT	The status bit value for interrupt source GSM_SLEEP_INT
7	SLP_XTAL_TIMER_INT	The status bit value for interrupt source SLP_XTAL_TIMER_INT
6	SLEEP_FEE_INT	The status bit value for interrupt source SLEEP_FEE_INT
5	WAKEUP_1X_INT	The status bit value for interrupt source WAKEUP_1X_INT
4	DEM_TIME_INT2	The status bit value for interrupt source DEM_TIME_INT2
3	DEM_TIME_INT1	The status bit value for interrupt source DEM_TIME_INT1
2	TIME_TICK_INT	The status bit value for interrupt source TIME_TICK_INT
1	DEM_26MS_INT	The status bit value for interrupt source DEM_26MS_INT.
0	EVEN_SEC_INT	The status bit value for interrupt source EVEN_SEC_INT

0x0058 INT_STATUS_1

Type: Read
Clock: HCLK

Each bit of the interrupt status register corresponds to a particular interrupt source. If any of these bits are set, an interrupt is asserted by the associated interrupt source. All of the bits in this register are sticky. This means that once a bit is set in this register, the source of the interrupt request must be reset, and the corresponding bit in the corresponding interrupt clear register must be set in order to clear these bits (in the said order).

INT_STATUS_1

Bits	Name	Description
31	GPIO_GROUP_INT	The status bit value for interrupt source GPIO_GROUP_INT
30	GPIO2_GROUP_INT	The status bit value for interrupt source GPIO2_GROUP_INT
29	MDDI_HOST_INT	The status bit value for interrupt source MDDI_HOST_INT
28	MDDI_CLIENT_INT	The status bit value for interrupt source MDDI_CLIENT_INT
27	DEC_OB_IRQ	The status bit value for interrupt source DEC_OB_IRQ
26	DDO_FULL_IRQ	The status bit value for interrupt source DDO_FULL_IRQ
25	DDI_READY_IRQ	The status bit value for interrupt source DDI_READY_IRQ

INT_STATUS_1 (Continued)

Bits	Name	Description
24	USB_OTG_INT	The status bit value for interrupt source USB_OTF_INT
23	COMPARE2_INT	The status bit value for interrupt source COMPARE2_INT
22	COMPARE1_INT	The status bit value for interrupt source COMPARE1_INT
21	PER_SLOT_INT	The status bit value for interrupt source PER_SLOT_INT
20	BT_WAKEUP	The status bit value for interrupt source BT_WAKEUP
19	ADSP_INT2	The status bit value for interrupt source ADSP_INT[2]
18	ADSP_INT1	The status bit value for interrupt source ADSP_INT[1]
17	ADSP_INT0	The status bit value for interrupt source ADSP_INT[0]
16	GRAPHICS_INT	The status bit value for interrupt source GRAPHICS_INT
15	MDP_INT	The status bit value for interrupt source MDP_INT
14	NAND_WR_ER_DONE_INT	The status bit value for interrupt source NAND_WR_ER_DONE_INT
13	NAND_OP_DONE_INT	The status bit value for interrupt source GPIO_GROUP_INT
12	MMCC_ERR_INT	The status bit value for interrupt source MMCC_ERR_INT
11	MMCC_INT	The status bit value for interrupt source MMCC_INT
10	KEYSENSE_GROUP_INT	The status bit value for interrupt source KEYSENSE_GROUP_INT
9	PN_ROLLOVER_1X	The status bit value for interrupt source PN_ROLLOVER_1X
8	SBI1_INT	The status bit value for interrupt source SBI1_INT
7	SBI0_INT	The status bit value for interrupt source SBI0_INT
6	I2CC_INT	The status bit value for interrupt source I2CC_INT
5	UART3_DP_RX_DATA_INT	The status bit value for interrupt source UART3_DP_RX_DATA_INT
4	UART2_DP_RX_DATA_INT	The status bit value for interrupt source UART2_DP_RX_DATA_INT
3	UART1_DP_RX_DATA_INT	The status bit value for interrupt source UART1_DP_RX_DATA_INT
2	UART3_INT	The status bit value for interrupt source UART3_INT
1	UART2_INT	The status bit value for interrupt source UART2_INT
0	UART1_INT	The status bit value for interrupt source UART1_INT

0x005C GPIO_INT_STATUS_0

Type: Read
Clock: HCLK

Each bit of the GPIO interrupt status register corresponds to a particular GPIO interrupt source. If any of these bits are set, an interrupt is asserted by the associated GPIO interrupt source. All of the bits in this register are sticky. This means that once a bit is set in this register, the source of the GPIO interrupt request must be reset, and the corresponding bit in the corresponding GPIO interrupt clear register must be set in order to clear these bits (in the said order).

Bits	Name	Description
31:0	GPIO_INT[31:0]	The status bit value for interrupt source GPIO[31:0].

0x0060 GPIO_INT_STATUS_1

Type: Read
Clock: HCLK

Each bit of the GPIO interrupt status register corresponds to a particular GPIO interrupt source. If any of these bits are set, an interrupt is asserted by the associated GPIO interrupt source. All of the bits in this register are sticky. This means that once a bit is set in this register, the source of the GPIO interrupt request must be reset, and the corresponding bit in the corresponding GPIO interrupt clear register must be set in order to clear these bits (in the said order).

Bits	Name	Description
21:0	GPIO_INT[53:32]	The status bit value for interrupt source GPIO[81:67,38:32].

0x0064 GPIO_INT_STATUS_4

Type: Read
Clock: HCLK

Each bit of the GPIO interrupt status register corresponds to a particular GPIO interrupt source. If any of these bits are set, an interrupt is asserted by the associated GPIO interrupt source. All of the bits in this register are sticky. This means that once a bit is set in this register, the source of the GPIO interrupt request must be reset, and the corresponding bit in the corresponding GPIO interrupt clear register must be set in order to clear these bits (in the said order).

Bits	Name	Description
24:0	GPIO_INT[78:54]	The status bit value for interrupt source GPIO[106:82].

7 General Purpose I/O (GPIO)

7.1 Overview

This chapter describes the general purpose I/O (GPIO) registers accessible by the ARM. There are 107 GPIOs. GPIOs 0-38 and 67-106 are controlled by these registers, while GPIOs 39-66 are controlled by the GPIO2 registers (see Chapter 8 for GPIO2 register descriptions). A paging scheme is used to address individual GPIOs and alternate functions.

The registers are arranged in three groups: input, output, and output enable, with two additional registers for programming output functions.

The reset state for all GPIOs is “input”, so to get a GPIO value as input, simply read the appropriate GPIO_IN register.

To have a GPIO output its main function as a value in the GPIO_OUT_*X* register, set the GPIO_OE_*X* register to 1 to enable the output path, and program the appropriate GPIO_OUT register. For example, to get the GPIO_OUT_0 output:

```
m_write(GPIO_OUT_0, 0x5555555555);
m_write(GPIO_OE_0, 0x11111111);
```

To use any alternate GPIO functions, you must:

1. Write the number of the GPIO to the GPIO_PAGE register (e.g., to program GPIO(*X*), write the value of *X* to GPIO_PAGE).
2. Set GPIO_CFG to select which function to output. Refer to Table 7-1, “GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0,” on page 6.

7.2 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: GPIO
- Number of word addresses: 64
- Byte Address: 0x2200 - 0x22FC
- OFFSET=CHIP_BASE+0x2200
- CHIP_BASE = 0x80000000
- chipaddr range = 13:2

7.2.1 GPIO output value registers

The GPIO_OUT_*x* registers in this section contain the output values for the enabled GPIO pins.

0x0000 GPIO_OUT_0

Type: Write
Clock: ASYNC

Bits	Name	Description
31:0	GPIO_OUT[31:0]	GPIO[31:0] output values for the enabled GPIO[31:0] pins. Reset state:0x00000000.

0x0004 GPIO_OUT_1

Type: Write
Clock: ASYNC
Reset State: 0x00

Bits	Name	Description
6:0	GPIO_OUT[38:32]	GPIO[38:32] output values for the enabled GPIO[38:32] pins.

0x0008 GPIO_OUT_4

Type: Write
Clock: ASYNC
Reset State: 0x0000

Bits	Name	Description
31:0	GPIO_OUT[98:67]	GPIO[98:67] output values for the enabled GPIO[98:67] pins.

0x000C GPIO_OUT_5

Type: Write
Clock: ASYNC
Reset State: 0x00

Bits	Name	Description
7:0	GPIO_OUT[106:99]	GPIO[106:99] output values for the enabled GPIO[106:99] pins.

7.2.2 GPIO output enabling registers

The GPIO_OE_*x* registers in this section are for output enabling.

0x0010 GPIO_OE_0

Type: Write
Clock: ASYNC
Reset State: 0x00000000

Bits	Name	Description
31:0	GPIO_OE[31:0]	The register is the output enable for GPIO[31:0]. Writing a 1 drives the output to the value specified in GPIO_OUT_0. Power up value: 0x00000000

0x0014 GPIO_OE_1

Type: Write
Clock: ASYNC
Reset State: 0x00

Bits	Name	Description
6:0	GPIO_OE[38:32]	The register is the output enable for GPIO[38:32]. Writing a 1 drives the output to the value specified in GPIO_OUT_1 Power up value : 0x00

0x0018 GPIO_OE_4

Type: Write
Clock: ASYNC
Reset State: 0x0000

Bits	Name	Description
31:0	GPIO_OE[98:67]	The register is the output enable for GPIO[98:67]. Writing a 1 drives the output to the value specified in GPIO_OUT_4.

0x001C GPIO_OE_5

Type: Write
Clock: ASYNC
Reset State: 0x00

Bits	Name	Description
7:0	GPIO_OE[106:99]	The register is the output enable for GPIO[106:99]. Writing a 1 drives the output to the value specified in GPIO_OUT_5.

7.2.3 GPIO output alternate-function registers

The two registers in this section are used to select an alternate function for a GPIO output.

0x0020 GPIO_PAGE

Type: Write
Clock: ASYNC
Reset State: 0x00

Bits	Name	Description
6:0	DATA[6:0]	Set GPIO_PAGE to the number of the GPIO selected (e.g., to select GPIO[*X*], set GPIO_PAGE to the value, *X*. Valid values are 0-38, 67-106). For GPIOs 39-66, refer to the GPIO2 registers.

0x0024 GPIO_CFG

Type: Write
Clock: ASYNC

Depending on the value of GPIO_PAGE (the GPIO selected), the values in GPIO_CFG will determine the function asserted.

Bits	Name	Description
5:2	FUNC_SEL	Set this to the appropriate value for the function desired, according to Table 7-1.
1	GPIO_PUPD_EN	'1' = pull is enabled (PD or PU, depending on programming of bit 0) '0' = pull is disabled (no pull = NP)
0	GPIO_PUPD_N	'1' = pull up (PU) '0' = pull down (PD)

Depending on the GPIO selected, the FUNC_SEL value will determine the function output. Table 7-1 summarizes the functions and the value to set to select them.

NOTE The GPIO_FUNC_SEL bits listed here are bits 2 and above of GPIO_CFG. Bits 1:0 control pull up/down behavior (GPIO_PUPD_EN, GPIO_PUPD_N), as shown in Table 7-1.

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 106				(default 10 = PD)
	gpio_out_reg(106)	gpio_oe_reg(106)	xxx0 (default)	
	mddi_low_freq_tstpt	’1’	xxx1	
gpio 105				(default 10 = PD)
	gpio_out_reg(105)	gpio_oe_reg(105)	xxx0 (default)	
	mdp_vsync_primary	gnd_tie	xxx1	
gpio 104				(default 10 = PD)
	gpio_out_reg(104)	gpio_oe_reg(104)	xxx0 (default)	
	mdp_vsync_secondary	gnd_tie	xxx1	
gpio 103				(default 10 = PD)
	gpio_out_reg(103)	gpio_oe_reg(103)	xx00 (default)	
	sdac_dout	’1’	xx01	
	aux_pcm_dout	aux_pcm_dout_en	xx10	
gpio 102				(default 10 = PD)
	gpio_out_reg(102)	gpio_oe_reg(102)	xx00 (default)	
	sdac_l_r	’1’	xx01	
	aux_pcm_sync	aux_pcm_sync_en	xx10	
gpio 101				(default 11 = PU)
	gpio_out_reg(101)	gpio_oe_reg(101)	xxx0 (default)	
	sdcc_datout(3)	sdcc_dat_en	xxx1	
gpio 100				(default 11 = PU)
	gpio_out_reg(100)	gpio_oe_reg(100)	xxx0 (default)	
	sdcc_datout(2)	sdcc_dat_en	xxx1	
gpio 99				(default 11 = PU)
	gpio_out_reg(99)	gpio_oe_reg(99)	xx00 (default)	
	sdcc_datout(1)	sdcc_dat_en	xx01	
	qmembist_ext_clk	gnd_tie	xx10	
gpio 98				(default 11 = PU)
	gpio_out_reg(98)	gpio_oe_reg(98)	xxx0 (default)	
	uart1_rfr_n	’1’	xxx1	
gpio 97				(default 10 = PD)
	gpio_out_reg(97)	gpio_oe_reg(97)	xxx0 (default)	
	uart1_cts_n	gnd_tie	xxx1	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 96				(default 10 = PD)
	gpio_out_reg(96)	gpio_oe_reg(96)	xxx0 (default)	
	uart1_dp_rx_data	gnd_tie	xxx1	
gpio 95				(default 11 = PU)
	gpio_out_reg(95)	gpio_oe_reg(95)	xxx0 (default)	
	uart1_dp_tx_data	’1’	xxx1	
gpio 94				(default 11 = PU)
	gpio_out_reg(94)	gpio_oe_reg(94)	xxx0 (default)	
	rx1_test_in(7)	gnd_tie	xxx1	
gpio 93				(default 10 = PD)
	gpio_out_reg(93)	gpio_oe_reg(93)	xxx0 (default)	
	sbst1	’1’	xxx1	
gpio 92				(default 10 = PD)
	gpio_out_reg(92)	gpio_oe_reg(92)	xx00 (default)	
	gp_pdm0	gp_pdm0_en	xx01	
	rx1_test_in(6)	gnd_tie	xx10	
gpio 91				(default 10 = PD)
	gpio_out_reg(91)	gpio_oe_reg(91)	xx00	
	uart2_rfr_n	’1’	xx01	
	uim_clk	’1’	xx10	
	rx1_test_in(5)	gnd_tie	xx11	
gpio 90				(default 10 = PD)
	gpio_out_reg(90)	gpio_oe_reg(90)	xx00	
	uart2_cts_n	gnd_tie	xx01	
	rx1_test_in(4)	gnd_tie	xx10	
gpio 89				(default 10 = PD)
	gpio_out_reg(89)	gpio_oe_reg(89)	xx00	
	uart2_dp_rx_data	gnd_tie	xx01	
	rx1_test_in(3)	gnd_tie	xx10	
gpio 88				(default 10 = PD)
	gpio_out_reg(88)	gpio_oe_reg(88)	xx00	
	uart2_dp_tx_data	’1’	xx01	
	uim_data_out	uim_data_en	xx10	
	rx1_test_in(2)	gnd_tie	xx11	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 87				(default 10 = PD)
	gpio_out_reg(87)	gpio_oe_reg(87)	xx00	
	uart3_rfr_n	'1'	xx01	
	uim2_clk	'1'	xx10	
	rx1_test_in(1)	gnd_tie	xx11	
gpio 86				(default 10 = PD)
	gpio_out_reg(86)	gpio_oe_reg(86)	xx00	
	uart3_cts_n	gnd_tie	xx01	
	rx1_test_in(0)	gnd_tie	xx10	
gpio 85				(default 10 = PD)
	gpio_out_reg(85)	gpio_oe_reg(85)	xx00	
	uart3_dp_rx_data	gnd_tie	xx01	
	rx0_test_in(7)	gnd_tie	xx10	
gpio 84				(default 10 = PD)
	gpio_out_reg(84)	gpio_oe_reg(84)	xx00	
	uart3_dp_tx_data	'1'	xx01	
	uim2_data_out	uim2_data_en	xx10	
	rx0_test_in(6)	gnd_tie	xx11	
gpio 83				(default 10 = PD)
	gpio_out_reg(83)	gpio_oe_reg(83)	xx00	
	rx0_test_in(5)	gnd_tie	xx01	
	grfc(14)	'1'	xx10	
	camif_data(0)	'1'	xx11	
gpio 82				(default 10 = PD)
	gpio_out_reg(82)	gpio_oe_reg(82)	xx00	
	camif_pclk	'1'	xx01	
	rx0_test_in(4)	gnd_tie	xx10	
	grfc(13)	'1'	xx11	
gpio 81				(default 10 = PD)
	gpio_out_reg(81)	gpio_oe_reg(81)	x000	
	gnd_tie	gnd_tie		
	rx0_test_in(3)	gnd_tie	x010	
	grfc(12)	'1'	x011	
	camif_data(1)	'1'	x100	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 80				(default 10 = PD)
	gpio_out_reg(80)	gpio_oe_reg(80)	xx00	
	aux_pcm_clk_out	aux_pcm_clk_en	xx01	
	rx0_test_in(2)	gnd_tie	xx10	
	grfc(11)	'1'	xx11	
gpio 79				(default 10 = PD)
	gpio_out_reg(79)	gpio_oe_reg(79)	xxx0	
	a1(24)	'1'	xxx1	
gpio 78				(default 10 = PD)
	gpio_out_reg(78)	gpio_oe_reg(78)	xxx0	
	a1(23)	'1'	xxx1	
gpio 77				(default 11 = PU)
	gpio_out_reg(77)	gpio_oe_reg(77)	xxx0	
	xmem1_cs_n(3)	'1'	xxx1	
gpio 76				(default 11 = PU)
	gpio_out_reg(76)	gpio_oe_reg(76)	xxx0	
	xmem1_cs_n(1)	'1'	xxx1	
gpio 75				(default 10 = PD)
	gpio_out_reg(75)	gpio_oe_reg(75)	xx00	
	sdram1_dqm_3	'1'	xx01	
	a(25)	'1'	xx10	
gpio 74				pull not programmable (keeper)
	gpio_out_reg(74)	gpio_oe_reg(74)	xx00	
	sdram1_d_out(23)	sdram1_d_en(23)	xx01	
	ebi1_hub_n	'1'	xx10	
gpio 73				pull not programmable (keeper)
	gpio_out_reg(73)	gpio_oe_reg(73)	xx00	
	sdram1_d_out(22)	sdram1_d_en(22)	xx01	
	ebi1_hlb_n	'1'	xx10	
gpio 72				pull not programmable (keeper)
	gpio_out_reg(72)	gpio_oe_reg(72)	xxx0	
	sdram1_d_out(21)	sdram1_d_en(21)	xxx1	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 71				pull not programmable (keeper)
	gpio_out_reg(71)	gpio_oe_reg(71)	xxx0	
	sdram1_d_out(20)	sdram1_d_en(20)	xxx1	
gpio 70				pull not programmable (keeper)
	gpio_out_reg(70)	gpio_oe_reg(70)	xxx0	
	sdram1_d_out(19)	sdram1_d_en(19)	xxx1	
gpio 69				pull not programmable (keeper)
	gpio_out_reg(69)	gpio_oe_reg(69)	xxx0	
	sdram1_d_out(18)	sdram1_d_en(18)	xxx1	
gpio 68				pull not programmable (keeper)
	gpio_out_reg(68)	gpio_oe_reg(68)	xxx0	
	sdram1_d_out(17)	sdram1_d_en(17)	xxx1	
gpio 67				pull not programmable (keeper)
	gpio_out_reg(67)	gpio_oe_reg(67)	xxx0	
	sdram1_d_out(16)	sdram1_d_en(16)	xxx1	
Refer to the GPIO2 chapter for GPIOs66-39.				
gpio 38				(default 11 = PU)
	gpio_out_reg(38)	gpio_oe_reg(38)	xxx0	
	lcd_cs_n	’1’	xxx1	
gpio 37				(default 10 = PD)
	gpio_out_reg(37)	gpio_oe_reg(37)	xx00	
	lcd_en	’1’	xx01	
	test_gps_resync	gnd_tie	xx10	
gpio 36				(default 11 = PU)
	gpio_out_reg(36)	gpio_oe_reg(36)	xxx0	
	xmem2_cs_n(3)	’1’	xxx1	
gpio 35				(default 11 = PU)
	gpio_out_reg(35)	gpio_oe_reg(35)	xxx0	
	xmem2_cs_n(2)	’1’	xxx1	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 34				(default 10 = PD)
	gpio_out_reg(34)	gpio_oe_reg(34)	xxx0	
	a2(20)	’1’	xxx1	
gpio 33				(default 11 = PD)
	gpio_out_reg(33)	gpio_oe_reg(33)	xx00	
	adsp_dbg_stb	’1’	xx01	
	nand2_flash_ready	gnd_tie	xx10	
gpio 32				H(default 11 = PU)
	gpio_out_reg(32)	gpio_oe_reg(32)	x000	
	mmc_data_out	mmc_data_en	x001	
	test_cdma_resync	gnd_tie	x010	
	mdsp_dbg_stb	’1’	x011	
	sdcc_dat0	’1’	x100	
gpio 31				(default 11 = PU)
	gpio_out_reg(31)	gpio_oe_reg(31)	x000	
	mmc_clk	’1’	x001	
	rx0_y2q_out	’1’	x010	
	dbg_bus(31)	’1’	x011	
	sdcc_clk	’1’	x100	
gpio 30				H(default 11 = PU)
	gpio_out_reg(30)	gpio_oe_reg(30)	x000	
	mmc_cmd_out	mmc_cmd_en	x001	
	rx0_y2i_out	’1’	x010	
	dbg_bus(30)	’1’	x011	
	sdcc_cmd	’1’	x100	
gpio 29				(default 10 = PD)
	gpio_out_reg(29)	gpio_oe_reg(29)	xx00	
	usb_rx_data	gnd_tie	xx01	
	rx0_y1q_out	’1’	xx10	
	dbg_bus(29)	’1’	xx11	
gpio 28				(default 10 = PD)
	gpio_out_reg(28)	gpio_oe_reg(28)	xx00	
	rx0_y1i_out	’1’	xx01	
	dbg_bus(28)	’1’	xx10	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 27				(default 11 = PU)
	gpio_out_reg(27)	gpio_oe_reg(27)	xx00	
	i2c_scl_out	i2c_scl_en	xx01	
	rx0_q_out(3)	’1’	xx10	
	dbg_bus(27)	’1’	xx11	
gpio 26				(default 11 = PU)
	gpio_out_reg(26)	gpio_oe_reg(26)	xx00	
	i2c_sda_out	i2c_sda_en	xx01	
	rx0_q_out(2)	’1’	xx10	
	dbg_bus(26)	’1’	xx11	
gpio 25				(default 11 = PU)
	gpio_out_reg(25)	gpio_oe_reg(25)	x000	
	bt_clk	gnd_tie	x001	
	rx0_q_out(1)	’1’	x010	
	dai_clk	’1’	x011	
	dbg_bus(25)	’1’	x100	
gpio 24				(default 11 = PU)
	gpio_out_reg(24)	gpio_oe_reg(24)	x000	
	bt_sbst	’1’	x001	
	rx0_q_out(0)	’1’	x010	
	dai_reset	gnd_tie	x011	
	dbg_bus(24)	’1’	x100	
gpio 23				(default 11 = PU)
	gpio_out_reg(23)	gpio_oe_reg(23)	x000	
	bt_sbck	’1’	x001	
	rx0_i_out(3)	’1’	x010	
	ad_data_clk_ch0	’1’	x011	
	dai_test_ctl1	gnd_tie	x100	
	dbg_bus(23)	’1’	x101	
gpio 22				(default 11 = PU)
	gpio_out_reg(22)	gpio_oe_reg(22)	x000	
	bt_sbdt_out	bt_sbdt_en	x001	
	rx0_i_out(2)	’1’	x010	
	dai_test_ctl2	gnd_tie	x011	
	dbg_bus(22)	’1’	x100	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 21				(default 10 = PD)
	gpio_out_reg(21)	gpio_oe_reg(21)	x000	
	bt_tx_rx_n	’1’	x001	
	rx0_i_out(1)	’1’	x010	
	dai_data_in	gnd_tie	x011	
	dbg_bus(21)	’1’	x100	
gpio 20				(default 10 = PD)
	gpio_out_reg(20)	gpio_oe_reg(20)	x000	
	bt_data_out	bt_data_en	x001	
	rx0_i_out(0)	’1’	x010	
	dai_data_out	’1’	x011	
	txdac_clk	’1’	x100	
	dbg_bus(20)	’1’	x101	
gpio 19				(default 10 = PD)
	gpio_out_reg(19)	gpio_oe_reg(19)	x000	
	synth_lock	gnd_tie	x001	
	rx1_q_out(3)	’1’	x010	
	gp_clk	’1’	x011	
	dbg_bus(19)	’1’	x100	
gpio 18				(default 10 = PD)
	gpio_out_reg(18)	gpio_oe_reg(18)	xx00	
	ringer	’1’	xx01	
	rx1_q_out(2)	’1’	xx10	
	dbg_bus(18)	’1’	xx11	
gpio 17				(default 11 = PU)
	gpio_out_reg(17)	gpio_oe_reg(17)	xx00	
	rx1_q_out(1)	’1’	xx01	
	dbg_bus(17)	’1’	xx10	
gpio 16				(default 10 = PD)
	gpio_out_reg(16)	gpio_oe_reg(16)	xx00	
	camif_vsync_out	camif_vsync_en	xx01	
	rx1_q_out(0)	’1’	xx10	
	dbg_bus(16)	’1’	xx11	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 15				(default 10 = PD)
	gpio_out_reg(15)	gpio_oe_reg(15)	x000	
	camif_hsync_in	gnd_tie	x001	
	rx1_i_out(3)	’1’	x010	
	sleep_n	’1’	x011	
	dbg_bus(15)	’1’	x100	
gpio 14				(default 10 = PD)
	gpio_out_reg(14)	gpio_oe_reg(14)	xx00	
	aux_pcm_din_out	aux_pcm_din_en	xx01	
	rx1_i_out(2)	gnd_tie	xx10	
	dbg_bus(14)	’1’	xx11	
gpio 13				(default 10 = PD)
	gpio_out_reg(13)	gpio_oe_reg(13)	x000	
	gp_mn	’1’	x001	
	rx1_i_out(1)	’1’	x010	
	txdac_iin(9)	’1’	x011	
	txdac_qin(9)	’1’	x100	
	dbg_bus(13)	’1’	x101	
	camclk_po	’1’	x110	
gpio 12				(default 10 = PD)
	gpio_out_reg(12)	gpio_oe_reg(12)	x000	
	grfc(9)	’1’	x001	
	rx1_i_out(0)	’1’	x010	
	txdac_iin(8)	’1’	x011	
	txdac_qin(8)	’1’	x100	
	dbg_bus(12)	’1’	x101	
gpio 11				(default 10 = PD)
	gpio_out_reg(11)	gpio_oe_reg(11)	x000	
	grfc(8)	’1’	x001	
	rx1_y2q_out	’1’	x010	
	txdac_iin(7)	’1’	x011	
	txdac_qin(7)	’1’	x100	
	tx_iqdata(7)	’1’	x101	
	dbg_bus(11)	’1’	x110	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 10				(default 10 = PD)
	gpio_out_reg(10)	gpio_oe_reg(10)	x000	
	grfc(7)	’1’	x001	
	rx1_y2i_out	’1’	x010	
	txdac_iin(6)	’1’	x011	
	txdac_qin(6)	’1’	x100	
	tx_iqdata(6)	’1’	x101	
	uart_ext_clk_src	gnd_tie	x110	
	dbg_bus(10)	’1’	x111	
gpio 9				(default 10 = PD)
	gpio_out_reg(9)	gpio_oe_reg(9)	x000	
	grfc(6)	’1’	x001	
	rx1_y1q_out	’1’	x010	
	txdac_iin(5)	’1’	x011	
	txdac_qin(5)	’1’	x100	
	tx_iqdata(5)	’1’	x101	
	codec_ovs_d_in	gnd_tie	x110	
	mod_mask_data	’1’	x111	
	dec_voc_fr_ref	’1’	1000	
	sleep_schmitt_out	’1’	1001	
	dbg_bus(9)	’1’	1010	
gpio 8				(default 11 = PU)
	gpio_out_reg(8)	gpio_oe_reg(8)	x000	
	grfc(5)	’1’	x001	
	rx1_y1i_out	’1’	x010	
	txdac_iin(4)	’1’	x011	
	txdac_qin(4)	’1’	x100	
	tx_iqdata(4)	’1’	x101	
	dbg_bus(8)	’1’	x110	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 7				(default 11 = PU)
	gpio_out_reg(7)	gpio_oe_reg(7)	x000	
	grfc(4)	’1’	x001	
	txdac_iin(3)	’1’	x010	
	txdac_qin(3)	’1’	x011	
	tx_iqdata(3)	’1’	x100	
	nsout_left	’1’	x101	
	dbg_bus(7)	’1’	x110	
	codec_rd_req	’1’	x111	
gpio 6				(default 10 = PD)
	gpio_out_reg(6)	gpio_oe_reg(6)	x000	
	grfc(3)	’1’	x001	
	txdac_iin(2)	’1’	x010	
	txdac_qin(2)	’1’	x011	
	tx_iqdata(2)	’1’	x100	
	nsout_right	’1’	x101	
	dbg_bus(6)	’1’	x110	
	codec_wr_req	’1’	x111	
gpio 5				(default 10 = PD)
	gpio_out_reg(5)	gpio_oe_reg(5)	x000	
	grfc(2)	’1’	x001	
	gnd_tie	gnd_tie	x010	
	txdac_iin(1)	’1’	x011	
	txdac_qin(1)	’1’	x100	
	tx_iqdata(1)	’1’	x101	
	intctl_fiq_n	’1’	x110	
	dbg_bus(5)	’1’	x111	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 4				(default 10 = PD)
	gpio_out_reg(4)	gpio_oe_reg(4)	x000	
	grfc(1)	’1’	x001	
	txdac_iin(0)	’1’	x010	
	txdac_qin(0)	’1’	x011	
	tx_iqdata(0)	’1’	x100	
	intctl_irq_n	’1’	x101	
	dbg_bus(4)	’1’	x110	
gpio 3				(default 10 = PD)
	gpio_out_reg(3)	gpio_oe_reg(3)	x000	
	grfc(0)	’1’	x001	
	mod_tx_clk	’1’	x010	
	gnd_tie	’1’	x011	
	dbg_bus(3)	’1’	x100	
	intctl_fiq_n	’1’	x101	
gpio 2				(default 10 = PD)
	gpio_out_reg(2)	gpio_oe_reg(2)	x000	
	pa_on1	’1’	x001	
	adsp_ext_irq	gnd_tie	x010	
	dbg_clk_out	’1’	x011	
	dbg_bus(2)	’1’	x100	
	intctl_irq_n	’1’	x101	
	wsym_clk	’1’	x110	
	ring_osc_clk	’1’	x111	
	clk_diff	’1’	1000	
	ebi1_clk_diff	’1’	1001	
	abc_sel(2)	gnd_tie	1010	

Table 7-1 GPIO functions: GPIO106-GPIO67, GPIO38-GPIO0 (Continued)

GPIO_PAGE selects	Function	Output enable	GPIO_CFG[5:2] value	GPIO_CFG[1:0] value
gpio 1				This GPIO uses a pukp pad, not an nppdpu pad, which most GPIOs use. (default x0 = PU) (x1 = KP)
	gpio_out_reg(1)	gpio_oe_reg(1)	x000	
	sbdt1_out	sbdt1_en	x001	
	mdsp_ext_irq	gnd_tie	x010	
	ad_data_clk_ch1	’1’	x011	
	dbg_bus(1)	’1’	x100	
	abc_sel(1)	gnd_tie	x101	
gpio 0				(default 01 = PD)
	gpio_out_reg(0)	gpio_oe_reg(0)	x000	
	sbck1	’1’	x001	
	wdog_stb	’1’	x010	
	adsp_wait_cnd	gnd_tie	x011	
	dbg_bus(0)	’1’	x100	
	bist_fail_en	gnd_tie	x101	
	abc_sel(0)	gnd_tie	x110	

7.2.4 GPIO input value registers

The registers in this section contain GPIO input values.

0x0034 GPIO_IN_0

Type: Read
Clock: ASYNC

Bits	Name	Description
31:0	GPIO_IN[31:0]	The register is the value of GPIO[31:0]. These values can be read at any time.

0x0038 GPIO_IN_1

Type: Read
Clock: ASYNC

Bits	Name	Description
6:0	GPIO_IN [38:32]	The register is the value of GPIO[42:32]. These values can be read at any time.

0x003C GPIO_IN_4

Type: Read
Clock: ASYNC

Bits	Name	Description
31:0	GPIO_IN[98:67]	The register is the value of GPIO[98:67]. These values can be read at any time.

0x0040 GPIO_IN_5

Type: Read
Clock: ASYNC

Bits	Name	Description
7:0	GPIO_IN[106:99]	The register is the value of GPIO[106:99]. These values can be read at any time.

8 GPIO2

8.1 ARM registers

This chapter describes the GPIO2 registers that control GPIOs 39-66 of the 99 GPIOs in MSM6550/6150. For descriptions of the other GPIO registers, please refer to the GPIO chapter of the Software Manual. These registers work in an identical manner as those in the other GPIO blocks; however, there are no alternate functions available.

- GPIO2 MEMORY SPACE = 0x80000000 - 0x8FFFFFFF (128 MBytes)
- OFFSET = GPIO2_BASE+0x0
- GPIO2_BASE = 0x80000000
- gpio2addr range 7:2

0x0000 GPIO_OE_2

Type: Write
Clock: Async

Bits	Name	Description
15:0	GPIO_OE[54:39]	The register is the output enable for GPIO[54:39]. Writing a 1 drives the output to the value specified in GPIO_OUT_2. Power up value is 0x0000. Reset state : 0x0000

0x0004 GPIO_OE_3

Type: Write
Clock: Async

Bits	Name	Description
11:0	GPIO_OE[66:55]	The register is the output enable for GPIO[[66:55]. Writing a 1 drives the output to the value specified in GPIO_OUT_3. Power up value is 0x000. Reset state : 0x0000

0x0008 GPIO_OUT_2

Type: Write
Clock: Async

Bits	Name	Description
15:0	GPIO_OUT[54:39]	GPIO[54:39] output values for the enabled GPIO[54:39] pins. Reset state : 0x0000

0x000C GPIO_OUT_3

Type: Write
Clock: Async

Bits	Name	Description
11:0	GPIO_OUT[66:55]	GPIO[66:55] output values for the enabled GPIO[66:55] pins. Reset state : 0x0000

0x0010 GPIO_INT_DETECT_CTL_2

Type: Read/Write
Clock: Async

Each bit controls the interrupt triggering mechanism: level detection, or edge detection. Setting the specific bit to ‘1’ will select edge detection for interrupt triggering, setting the specific bit to ‘0’ will select level match (with POLARITY register values) for interrupts.

Bits	Name	Description
15:0	GPIO_INT_CTL[54:39]	Sets the corresponding bit to enable the edge detection for the GPIO interrupt sources [54:39]

0x0014 GPIO_INT_DETECT_CTL_3

Type: Read/Write
Clock: Async

Each bit controls the interrupt triggering mechanism: level detection, or edge detection. Setting the specific bit to '1' will select edge detection for interrupt triggering, setting the specific bit to '0' will select level match (with POLARITY register values) for interrupts.

Bits	Name	Description
11:0	GPIO_INT_CTL[66:55]	Sets the corresponding bit to enable the edge detection for the GPIO interrupt sources [66:55]

0x0018 INT_POLARITY_3

Type: Read/Write
Clock: Async

Bits	Name	Description
15:0	GPIO_POLARITY[54:39]	If GPIO[n] is matching with GPIO_POLARITY[n], then it can cause GPIO group interrupt (depending on GPIO_INT_DETECT_CTL_n and GPIO_INT_MASK_n setup).

0x001C INT_POLARITY_4

Type: Read/Write
Clock: Async

Bits	Name	Description
11:0	GPIO_POLARITY[66:55]	If GPIO[n] is matching with GPIO_POLARITY[n], then it can cause GPIO group interrupt (depending on GPIO_INT_DETECT_CTL_n and GPIO_INT_MASK_n setup).

0x0020 GPIO_INT_EN_2

Type: Read/Write
Clock: Async

This register is used to enable GPIO level 2 interrupt sources.

When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

Bits	Name	Description
15:0	GPIO_INT_EN[54:39]	The enable bit value for GPIO interrupt source [54:39].

0x0024 GPIO_INT_EN_3

Type: Read/Write
Clock: Async

This register is used to enable GPIO level 2 interrupt sources.

When a bit in this register is cleared (0), the corresponding interrupt source is masked. When a bit in this register is set (1), an interrupt request from the corresponding interrupt source is enabled. A masked interrupt source remains masked until the corresponding bit in this register is set (1).

Bits	Name	Description
11:0	GPIO_INT_EN[66:55]	The enable bit value for GPIO interrupt source [66:55].

0x0028 GPIO_INT_CLEAR_2

Type: Write
Clock: Async

This register is used to clear the register bits of the GPIO interrupt status register. If a bit in the GPIO interrupt clear register is set (1), the corresponding bit in the GPIO interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

Bits	Name	Description
15:0	GPIO[54:39]	The clear bit value for GPIO level 2 interrupt sources [54:39].

0x002C GPIO_INT_CLEAR_3

Type: Write
Clock: Async

This register is used to clear the register bits of the GPIO interrupt status register. If a bit in the GPIO interrupt clear register is set (1), the corresponding bit in the GPIO interrupt status register is cleared (0). The interrupt request is also cleared if there are no pending interrupt requests.

Bits	Name	Description
11:0	GPIO[66:55]	The clear bit value for GPIO level 2 interrupt sources [66:55].

0x0030 GPIO2_PAGE

Type: Write
Clock: Async

Bits	Name	Description
6:0	DATA	Set GPIO2_PAGE to the number of the GPIO selected (e.g., to select GPIO[*X*], set GPIO2_PAGE to the value, *X*. Valid values are 39-66). For GPIOs 0-38, 67-98, refer to the GPIO registers.

0x0034 GPIO2_CFG

Type: Write
Clock: Async

Bits	Name	Description
5:2	FUNC_SEL	Set this to the appropriate value for the function desired, according to Table 8-1.
1	GPIO_PUPD_EN	'1' = pull is enabled (PD or PU, depending on programming of bit 0) '0' = pull is disabled (no pull = NP)
0	GPIO_PUPD_N	'1' = pull up (PU) '0' = pull down (PD)

Depending on the GPIO selected, the FUNC_SEL value will determine the function output.Table 8-1 summarizes the functions and the value to set to select them.

NOTE The GPIO_FUNC_SEL bits listed here are bits 2 and above of GPIO2_CFG. Bits 1:0 control pull up/down behavior (GPIO_PUPD_EN, GPIO_PUPD_N), as shown in Table 8-1.

Table 8-1 GPIO2 functions: GPIO66-GPIO39

GPIO2_PAGE selects	Function	Output enable	GPIO2_CFG[5:2] value	GPIO2_CFG[1:0] value
gpio 66				(default 11 = PU)
	gpio_out_reg(66)	gpio_oe_reg(66)	xxx0	
	rx0_test_in(1)	gnd_tie	xxx1	
gpio 65				(default 11 = PU)
	gpio_out_reg(65)	gpio_oe_reg(65)	xxx0	
	rx0_test_in(0)	gnd_tie	xxx1	

Table 8-1 GPIO2 functions: GPIO66-GPIO39 (Continued)

GPIO2_PAGE selects	Function	Output enable	GPIO2_CFG[5:2] value	GPIO2_CFG[1:0] value
gpio 64				(default 11 = PU)
	gpio_out_reg(64)	gpio_oe_reg(64)	xxx0	
	rx1_test_in(17)	gnd_tie	xxx1	
gpio 63				(default 11 = PU)
	gpio_out_reg(63)	gpio_oe_reg(63)	xxx0	
	rx1_test_in(16)	gnd_tie	xxx1	
gpio 62				(default 11 = PU)
	gpio_out_reg(62)	gpio_oe_reg(62)	xxx0	
	rx1_test_in(15)	gnd_tie	xxx1	
gpio 61				pull not programmable (keeper)
	gpio_out_reg(61)	gpio_oe_reg(61)	xx00	
	camif_data9	gnd_tie	xx01	
	rx1_test_in(14)	gnd_tie	xx10	
	ext_clk2	gnd_tie	xx11	
gpio 60				pull not programmable (keeper)
	gpio_out_reg(60)	gpio_oe_reg(60)	xx00	
	camif_data8	gnd_tie	xx01	
	rx1_test_in(13)	gnd_tie	xx10	
	ext_clk1	gnd_tie	xx11	
gpio 59				pull not programmable (keeper)
	gpio_out_reg(59)	gpio_oe_reg(59)	x000	
	camif_data7	gnd_tie	x001	
	rx1_test_in(12)	gnd_tie	x010	
	en_ant_gps	’1’	x011	
	en_ant_1x	’1’	x100	
	en_ant_hdr	’1’	x101	

Table 8-1 GPIO2 functions: GPIO66-GPIO39 (Continued)

GPIO2_PAGE selects	Function	Output enable	GPIO2_CFG[5:2] value	GPIO2_CFG[1:0] value
gpio 58				pull not programmable (keeper)
	gpio_out_reg(58)	gpio_oe_reg(58)	x000	
	camif_data6	gnd_tie	x001	
	rx1_test_in(11)	gnd_tie	x010	
	en_ant_gps	’1’	x011	
	en_ant_1x	’1’	x100	
	en_ant_hdr	’1’	x101	
gpio 57				pull not programmable (keeper)
	gpio_out_reg(57)	gpio_oe_reg(57)	xx00	
	camif_data5	gnd_tie	xx01	
	rx1_test_in(10)	gnd_tie	xx10	
gpio 56				pull not programmable (keeper)
	gpio_out_reg(56)	gpio_oe_reg(56)	xx00	
	camif_data4	gnd_tie	xx01	
	rx1_test_in(9)	gnd_tie	xx10	
gpio 55				pull not programmable (keeper)
	gpio_out_reg(55)	gpio_oe_reg(55)	xx00	
	camif_data3	gnd_tie	xx01	
	rx1_test_in(8)	gnd_tie	xx10	
gpio 54				pull not programmable (keeper)
	gpio_out_reg(54)	gpio_oe_reg(54)	xx00	
	camif_data2	gnd_tie	xx01	
	rx0_test_in(17)	gnd_tie	xx10	
gpio 53				(default 11 = PU)
	gpio_out_reg(53)	gpio_oe_reg(53)	xxx0	
	rx0_test_in(16)	gnd_tie	xxx1	
gpio 52				(default 11 = PU)
	gpio_out_reg(52)	gpio_oe_reg(52)	xxx0	
	rx0_test_in(15)	gnd_tie	xxx1	

Table 8-1 GPIO2 functions: GPIO66-GPIO39 (Continued)

GPIO2_PAGE selects	Function	Output enable	GPIO2_CFG[5:2] value	GPIO2_CFG[1:0] value
gpio 51				(default 11 = PU)
	gpio_out_reg(51)	gpio_oe_reg(51)	xxx0	
	rx0_test_in(14)	gnd_tie	xxx1	
gpio 50				(default 11 = PU)
	gpio_out_reg(50)	gpio_oe_reg(50)	xxx0	
	rx0_test_in(13)	gnd_tie	xxx1	
gpio 49				(default 11 = PU)
	gpio_out_reg(49)	gpio_oe_reg(49)	xxx0	
	rx0_test_in(12)	gnd_tie	xxx1	
gpio 48				(default 11 = PU)
	gpio_out_reg(48)	gpio_oe_reg(48)	xxx0	
	rx0_test_in(11)	gnd_tie	xxx1	
gpio 47				(default 11 = PU)
	gpio_out_reg(47)	gpio_oe_reg(47)	xxx0	
	rx0_test_in(10)	gnd_tie	xxx1	
gpio 46				(default 11 = PU)
	gpio_out_reg(46)	gpio_oe_reg(46)	xxx0	
	rx0_test_in(9)	gnd_tie	xxx1	
gpio 45				(default 10 = PD)
	gpio_out_reg(45)	gpio_oe_reg(45)	xxx0	
	rx0_test_in(8)	gnd_tie	xxx1	
gpio 44				(default 11 = PU)
	gpio_out_reg(44)	gpio_oe_reg(44)	xxx0	
	gp_clk	’1’	xxx1	
gpio 43				(default 11 = PU)
	gpio_out_reg(43)	gpio_oe_reg(43)	xxx0	
	gstep_ch1	’1’	xxx1	
gpio 42				(default 10 = PD)
	gpio_out_reg(42)	gpio_oe_reg(42)	xxx0	
	gstep_ch0	’1’	xxx1	
gpio 41				(default 11 = PU)
	gpio_out_reg(41)	gpio_oe_reg(41)	xxx0	
	system_reset	gnd_tie	xxx1	
gpio 40				(default 11 = PU)
	gpio_out_reg(40)	gpio_oe_reg(40)	xxx0	

Table 8-1 GPIO2 functions: GPIO66-GPIO39 (Continued)

GPIO2_PAGE selects	Function	Output enable	GPIO2_CFG[5:2] value	GPIO2_CFG[1:0] value
gpio 39				(default 11 = PU)
	gpio_out_reg(39)	gpio_oe_reg(39)	xxx0	

0x0040 GPIO_IN_2

Type: Read
Clock: Async

Bits	Name	Description
15:0	GPIO_IN[54:39]	The register is the value of GPIO[54:39]. These values can be read at any time.

0x0044 GPIO_IN_3

Type: Read
Clock: Async

Bits	Name	Description
11:0	GPIO_IN[66:55]	The register is the value of GPIO[66:55]. These values can be read at any time.

0x0048 GPIO_INT_STATUS_2

Type: Read
Clock: Async

Bits	Name	Description
15:0	GPIO_INT_STATUS[54:39]	The status bit value for GPIO interrupt sources [54:39].

0x004C GPIO_INT_STATUS_3

Type: Read
Clock: Async

Bits	Name	Description
11:0	GPIO_INT_STATUS[66:55]	The status bit value for GPIO interrupt sources [66:55].

0x0050 KEYSENSE_RD

Type: Read
Clock: Async

The keysense register provides the value of GPIO pins that can be used as keysense pins. The values can be read at anytime.

Bits	Name	Description
4:0	KEYSENSE_RD[4:0]	gpio48 : keysense4_n gpio47 : keysense3_n gpio46 : keysense2_n gpio63 : keysense1_n gpio62 : keysense0_n

9 Analog SBI

9.1 Analog SBI registers

For the analog portion of the MSM6550 / MSM6150 device, the SBI is used to select the modes of operation, including test modes.

There is no power-up default state for the SBI registers. All SBI registers have to be initialized to be in a known state after power-up. The software defaults are only possible typical values. The actual value is set by the current MSM6550 / MSM6150 software.

The SBI registers have a 6-bit address. Refer also to Figure 9-1, Wideband codec block diagram.

Table 9-1 CODEC power-down register

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_PD1	0x20	Reserved.	Software default: 1
		CODEC_PD1[6]	PD_Txck: 1: Power down codec clock to Tx path
		CODEC_PD1[5]	PD_Rxck_L 1: Power down codec clock to Rx left channel
		CODEC_PD1[4]	PD_biqd_L: 1: Power down left channel biquad filter
		CODEC_PD1[3]	PD_E1: 1: Power down earpiece amp
		CODEC_PD1[2]	PD_HPH_L: 1: Power down left channel head phone amp (HPH_L)
		CODEC_PD1[1]	PD_AUX_L: 1: Power down left channel auxiliary amp (Line_L)

Table 9-1 CODEC power-down register (Continued)

Register Name	Reg_add[5:0]	Data[7:0]	Function
		CODEC_PD1[0]	CODEC_SLEEP: 1: Power down codec core For complete codec power down, codec_ck must be turned off.
CODEC_PD2	0x21	Reserved.	Software default: 1
		CODEC_ PD2[6]	PD_biqd_R: 1: power down right channel biquad filter
		CODEC_ PD2[5]	PD_Rxck_R: 1: power down codec clock to Rx right channel
		CODEC_ PD2[4]	PD_HPH_R: 1: power down right channel headphone amp (HPH_R)
		CODEC_ PD2[3]	PD_AUX_R 1: power down right channel auxiliary amp (Line_R)
		CODEC_ PD2[2]	PD_MOD: 1: power down modulator
		CODEC_ PD2[1]	PD_MIC1: 1: Power down MicAmp1
		CODEC_ PD2[0]	PD_MIC2: 1: Power down MicAmp2
CODEC_PD3	0x22	CODEC_ PD3[7]	PD_Tx_RefBuf 1: Power down Tx reference buffers
		CODEC_ PD3[2]	PD_BG : 1: Power down bandgap ckt
		CODEC_ PD3[1]	PD_REF: 1: power down voltage and current reference circuits
		CODEC_ PD3[0]	PD_MB: 1: Power down microphone bias

Table 9-2 CODEC Configuration register1

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_C1	0x23	CODEC_C1[7]	TX_Modulator_OTA_CNTL 0: select OTA_BW1 when input is 4KHz 1: Select OTA_BW2 when input is 8KHz
		CODEC_C1[6]	G2: 1: Set MicAmp gain to -2dB Note: only select one of G2, G6, and G8
		CODEC_C1[5]	G6: 1: Set MicAmp gain to 6dB Note: only select one of G2, G6, and G8
		CODEC_C1[4]	G8: 1: Set MicAmp gain to 8dB Note1: only select one of G2, G6, and G8 Note2: G2=G6=G8=0⁄16dB
		CODEC_C1[3]	MIC1_SEL: 1: Selects Mic1 input
		CODEC_C1[2]	MIC2_SEL: 1: Selects Mic2 input
		CODEC_C1[1]	MICA_SEL: 1: Selects Auxiliary Mic input
		CODEC_C1[0]	MOR_DISABLE 1: Disable MicAmp auto start up circuit

Table 9-3 CODEC Configuration register2

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_C2	0x24	CODEC_C2[7]	TX_data_sync_polarity Codec_do is valid on 0: falling edge of codec_clk 1: rising edge of codec_clk
		CODEC_C2[6]	RX_data_sync_polarity Both Codec_di_L and Codec_di_R are valid on 0: falling edge of codec_clk 1: rising edge of codec_clk

Table 9-3 CODEC Configuration register2 (Continued)

Register Name	Reg_add[5:0]	Data[7:0]	Function
		CODEC_C2[5:4]	Programming DAC_ref in Rx path 00: Vref=2.0V 01: Vref=1.682V 10: Vref=1.736V 11: Vref=1.417V
		CODEC_C2[3:0]	0000: select 80MHz OTA (fs=12.288&11.2896MHz) 1000: select 55MHz OTA (fs=8.192MHz) 1100: select 42MHz OTA (fs=6.144&5.6448&4.096MHz) 1110: select 22MHz OTA (fs=2.048MHz) Note: all other combinations of these bits produce illegal states

Table 9-4 CODEC Configuration register 3

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_C3	0x25	CODEC_C3[7]	Sel_Aux_Linein_L 1: select left channel Aux_line_in
		CODEC_C3[6]	Sel_Aux_Linein_R 1: select right channel Aux_line_in
		CODEC_C3[5:2]	1000 : set Aux_linein buffer amplifier gain to –6.5dB 0100 : set Aux_linein buffer amplifier gain to –3.5dB 0010 : set Aux_linein buffer amplifier gain to –0.5dB 0001: set Aux_linein buffer amplifier gain to 2.5dB 0000: set Aux_linein buffer amplifier gain to 5.5dB
		CODEC_C3[1]	Aux_startup_enable 1: enable startup circuit for Aux PGA
		CODEC_C3[0]	MIC_RESET: 1: Reset MicAmp input to VSS

Table 9-5 CODEC Test MUX register

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_TMUX	0x26	Reserved.	Software default: 00000000

Table 9-6 CODEC Configuration Register 4

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_TEST	0x27	CODEC_TEST[7]	Sel_mica_mux3: 1: Micamp2 outputs are connected to sigma-delta modulator inputs.
		CODEC_TEST[6]	Sel_mica_mux2: 1: Micamp1 outputs are connected to sigma-delta modulator input (Micamp2 is bypassed)

Table 9-6 CODEC Configuration Register 4 (Continued)

Register Name	Reg_add[5:0]	Data[7:0]	Function
		CODEC_TEST[5]	Sel_mica_mux1: 1: Micamp1 outputs are connected to MICOUTP/MICOUTN Pins
		CODEC_TEST[0]	CODEC_LOOPBACK_L: 1: Feeds ADC bit stream to left channel 1b DAC going to SC biquad filter

Table 9-7 CODEC Configuration Register 5

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_SEL	0x28	CODEC_SEL[7]	Pamp-input-sel_L 1: along with SEL[3], allow signals to be connected from right channel to left channel or left channel to right channel, and used by different power amplifiers
		CODEC_SEL[6]	Sel-biqd_L-output 1: use left channel biquad filter output for power amplifiers
		CODEC_SEL[5]	Sel-biqd_R-output 1: use right channel biquad filter output for power amplifiers
		CODEC_SEL[4]	Line-mode-off 1: select differential-mode for AUX power amplifiers
		CODEC_SEL[3]	Pamp-input-sel_R 1: along with SEL[7], allow signals to be connected from right channel to left channel or left channel to right channel, and used by different power amplifiers
		CODEC_SEL[0]	CODEC_LOOPBACK_R: 1: Feeds ADC bit stream to right channel 1b DAC going to SC biquad filter

Table 9-8 CODEC Configuration Register 6

Register Name	Reg_add[5:0]	Data[7:0]	Function
CODEC_SEL_2	0x35	CODEC_SEL_2[7]	Headset switch detect to Hkadc 1: connect headset switch detect voltage to Hkin7
		CODEC_SEL_2[4]	Ear1_add 1: Add Aux_line_inL signal to Ear1 amp
		CODEC_SEL_2 [3]	HPH_L_add 1: Add Aux_line_inL signal to left channel headphone amp
		CODEC_SEL_2[2]	HPH_R_add 1:1:Add Aux_line_inR signal to right channel headphone amp
		CODEC_SEL_2[1]	Aux_L_add 1: Add Aux_line_inL signal to left channel auxiliary amp
		CODEC_SEL_2[0]	Aux_R_add 1: Add Aux_line_inR signal to right channel auxiliary amp

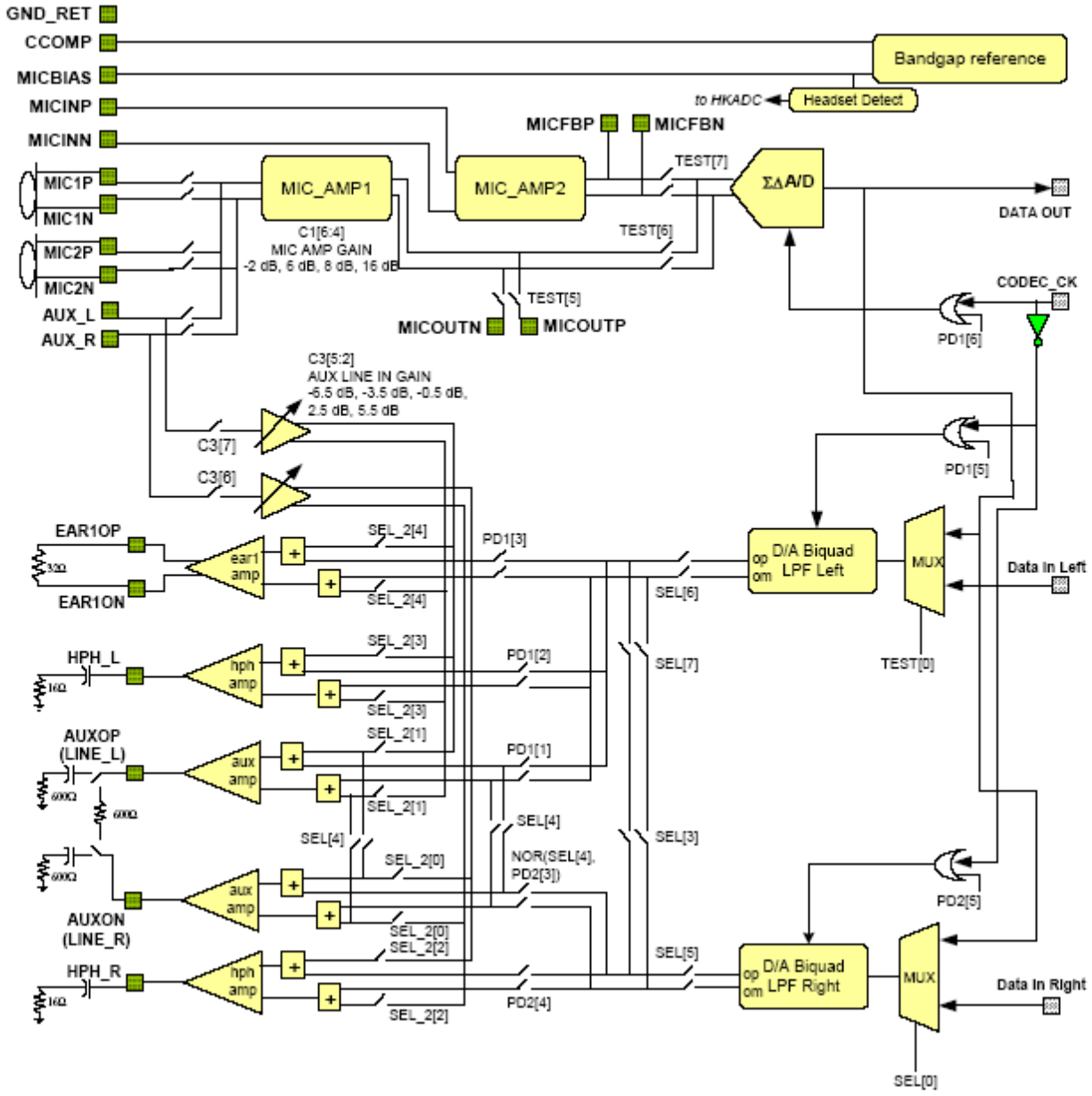


Figure 9-1 Wideband codec block diagram

Table 9-9 HKADC configuration register

Register Name	Reg_add[5:0]	Data[7:0]	Function
HKADC_CONFIG	0x30	Reserved.	Software default: 0
		HKADC_CONFIG[6:4]	VREF_SEL: Voltage Reference Selection 000 : V2 (1.86 V) 001 : Vbg (1.24 V) 010 : Vbg/2 (0.62 V) 011 : HKAIN6 (external reference) 1XX : VDDA
		HKADC_CONFIG[3:1]	MUX_SEL_IN: HKADC Analog Input Selection 000: HKAIN0 001: HKAIN1 010: HKAIN2 011: HKAIN3 100: HKAIN4 101: HKAIN5 110: HKAIN6 111: HKAIN7
		HKADC_CONFIG[0]	FAST SAMPLE: Determines length of sampling time (the time the sample and hold capacitor is allowed to charge). 0: Sample Period = 48 TCXO/4 clock cycles (9.8 us when TCXO is 19.68 MHz) 1: Sample Period = 24 TCXO/4 clock cycles (4.9 us when TCXO is 19.68 MHz)

Table 9-10 HKADC setup register

Register Name	Reg_add[5:0]	Data[7:0]	Function
HKADC_SETUP	0x31	HKADC_ SETUP [7]	1: connect Vdd_msm to Hkin7
		HKADC_ SETUP [6:5]	DIV: These bits determine the ADC comparison clock frequency. 00: TCXO/4 divided by 2. 01: TCXO/4 divided by 4. 10: TCXO/4 divided by 8. 11: TCXO/4 divided by 16.
		HKADC_ SETUP [4]	FAST RESET: This bit determines how long the comparator in the HKADC is reset between comparisons. 0: Reset lasts for 1/(4*comparison clock frequency). 1: Reset lasts for 1 TCXO/4 clock period. Comparison clock is tcxo/(4*DIV)
		HKADC_ SETUP [3]	BYPASS: 0: Normal Operation 1: Bypass Operation
		HKADC_ SETUP [2]	HKADC POWER UP: Power-up Override : 1: HKADC continuously powered-up. 0: HKADC powers up only during conversions. **Note**: Setting this bit causes the HKADC to be powered up continuously. It has the secondary effect of disabling the 5us warm up time as long as the CONVERT_ON_CONFIG_WR bit is low. HKADC_POWER_UP=SW_HKADC_POWER_UP AND SLEEPB

Table 9-10 HKADC setup register (Continued)

Register Name	Reg_add[5:0]	Data[7:0]	Function
		HKADC_ SETUP [1]	CONVERT_ON_CONFIG_WR : This bit controls whether or not a conversion is started when writing to the HKADC_CONFIG register. 0: Feature Disabled 1: Feature Enabled
		HKADC_ SETUP [0]	SW_RESET A 0-1-0 toggling sequence is required to reset the HKADC digital section.

Table 9-11 HKADC start of conversion register

Register Name	Reg_add[5:0]	Data[7:0]	Function
HKADC_CONV_START	0x32	COMMAND_WR	A write to this register starts a conversion.

Table 9-12 HKADC conversion data register

Register Name	Reg_add[5:0]	Data[7:0]	Function
HKADC_DATA_RD	0x33	HKADC_DATA[7:0]	READ ONLY Register This register holds the HKADC conversion data result. Data is valid when the ADC_EOC bit goes high. The data remains valid until the end of the next conversion.

Table 9-13 HKADC status register

Register Name	Reg_add[5:0]	Data[7:0]	Function
HKADC_STATUS	0x34	HKADC_ STATUS [7:1]	Reserved
		HKADC_ STATUS [0]	ADC_EOC: HKADC End of Conversion Bit 0: Conversion In Progress 1: Conversion Done **Note:** This bit goes low on a reset or when the HKADC has been started by a HKADC_CONV_START or a HKADC_CONFIG write (when CONVERT_ON-CONFIG_WR = 1). It goes high when the HKADC conversion data is valid, and remains high until a reset or start of a new conversion occurs. This bit does not need to be polled as long as SW waits for the time specified on the HKADC conversion time's tables.

10 Real Time Counter

10.1 ARM registers

The ARM real time counter registers are separated into two sections, according to their respective clocks:

- CDMA_CHIPX8 clock registers:
  - Offset = CHIP_BASE+0x6000 MAX=CHIP_BASE+0x60FC
- GPS_CHIPX8 clock registers:
  - Offset = CHIP_BASE+0x6100 MAX=CHIP_BASE+0x61FC

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: RTC
- Number of word addresses: 128
- Byte Address: 0x6000 - 0x61FC (total space)
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

10.1.1 CDMA_CHIPX8 clock registers

These real time counter ARM registers are clocked by CDMA_CHIPX8.

0x0000 CDMA_RTC_SYNC

Type: Read
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	CDMA_RTC_SYNC	This register latches the CDMA RTC (wall timer) at the strobe time of the sync80m, which is provided by the 1xdemod combiner. ARM can read this register when we first switch to GPS mode so that Firmware can synchronize the DSP system time based on this *CDMA_RTC_SYNC* value and the *GPS_RTC_CNT* register.

0x0004 RTC_CNT_1X

Type: Read
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	RTC_CNT_1X	This register contains the 1XRTC counter value at the time of accessing.

0x0008 RTC_SLEEP

Type: Write
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	RTC_SLEEP	This register provides the value for the 1XRTC to compare against, so that a strobe is generated when the RTC counter reaches the value in this register. In order for this strobe to trigger the sleep controller, *RTC_CTL:SLEEP_EN* bit needs to be set (1) beforehand.

0x000C RTC_OFFSET

Type: Write
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	RTC_OFFSET	This register provides the offset value for RTC to be adjusted. Software configures this register and then issues the *RTC_CTL:RTC_ADJUST* command to add this offset into the RTC counter. Software must calculate this register value based on the RTC value at the status dump command and the intended value after adjustment.

0x0010 RTC_LOAD

Type: Write
Clock: CDMA_CHIPX8

This register holds the value to be loaded into the 1X RTC counter upon wake-up or test resynchronization. Software specifies a value in chipx8 resolution.

Bits	Name	Description
17:0	RTC_LOAD	Value in chipx8 resolution to be loaded into the RTC upon wake up from sleep, or upon assertion of the *test_cdma_resync* pin (test mode).

0x0014 RTC_CTL

Type: Write
Clock: CDMA_CHIPX8

This register controls the RTC block.

Bit[0] is used to provide the RTC re-loading command. Upon the writing of this register with bit[0] as '1', the programmed value in *RTC_OFFSET[17:0]* is added to current *RTC_CNT[17:0]*, which advances *RTC_CNT* by the programmed number of *chipx8* cycles.

Bit[1] gives the option for software to disable or enable the ability of triggering MSM into sleep mode when the programmed value in register *RTC_SLEEP* is reached.

Bit[2] enables the general purpose interrupt generation when the programmed counter value in *RTC_GP_COMPARE1* is reached. Otherwise, interrupt generation is disabled.

Bit[3] enables the general purpose interrupt generation when the programmed counter value in *RTC_GP_COMPARE1* is reached. Otherwise, interrupt generation is disabled.

Bit[4] enables the interrupt generation when the 18-bit 1X RTC rolls over.

Bit[6:5] are the reset command register for 1X and HDR counter, write '1' to either of these two bits put the corresponding RTC into reset state and stay in reset. A '0' is required to be written to this same location to bring the corresponding RTC out of reset state.

RTC_CTL

Bits	Name	Description
6	RESET_HDR	Write '1' at this location puts the HDR RTC counter into reset state, write '0' again into this location to bring HDR counter out of reset state.
5	RESET_1X	Write '1' at this location puts the 1X RTC counter into reset state, write '0' again into this location to bring 1X counter out of reset state.
4	ROLLOVER_INT_EN	Set to '1 to enable the interrupt generation at rollover of the 18 bit 1X RTC counter. Software detect this interrupt and increment the rollover counter maintained in software.
3	COMPARE2_MASK	Set to '1 to enable the *RTC_GP_COMPARE2* interrupt. clear to '0' to disable it.
2	COMPARE1_MASK	Set to '1 to enable the *RTC_GP_COMPARE1* interrupt. clear to '0' to disable it.
1	SLEEP_EN	When set (1), a sleep triggering strobe is generated when the programmed RTC in *RTC_SLEEP* register value is reached. This serves as the similar purpose of combiner *pn_roll* to trigger MSM into sleep.
0	RTC_ADJUST	Set (1) this bit to advance RTC counter by a number of *chipx8* cycles specified in *RTC_OFFSET[17:0]* register.

0x0018 RTC_GP_COMPARE1

Type: Write
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	RTC_GP_COMPARE1	This register specifies an RTC counter value where a general purpose interrupt will be generated.

0x001C RTC_GP_COMPARE2

Type: Write
Clock: CDMA_CHIPX8

Bits	Name	Description
17:0	RTC_GP_COMPARE2	This register specifies an RTC counter value where a general purpose interrupt will be generated.

0x0020 RTC_DIFF_1XHDR

Type: Read
Clock: CDMA_CHIPX8

Bits	Name	Description
18:0	RTC_DIFF	Reading this register returns the difference between the 1X and HDR RTC counters, the returned value is in the format of 2's complement, positive when 1X RTC counter value is larger, otherwise, it's negative.

10.1.2 GPS_CHIPX8 clock registers

These real time counter ARM registers are clocked by GPS_CHIPX8.

0x0000 GPS_RTC_CTL

Type: Write
Clock: GPS_CHIPX8

Bits	Name	Description
0	RESET_GPS	Writing ‘1’ at this location puts the GPS RTC counter into reset state; write ‘0’ again into this location to bring GPS counter out of reset state.

0x0004 GPS_RTC_CNT

Type: Read
Clock: GPS_CHIPX8
Reset State: 0x0000

Because there is only one read register in the RTC_GPS chipx8 address range, reads from other addresses in this range will return this value also.

Bits	Name	Description
31:13	CNT_1MS	This upper 18 bit counter counts how many times the lower 13 bit counter rolls over. This upper counter essentially records the time in the unit of millisecond about the elapsed GPS time since the last reset operation. Reading this register returns the counter value at the time of accessing. It is reset at the following sync80m strobe after GPS mode is armed. This register is initialized at the same time when *CDMA_RTC_SYNC* is updated. Information inside this register together with *CDMA_RTC_SYNC* register yields the absolute GPS system time.
12:0	CNT_CX8	This lower 13 bit counter counts the gps cx8 cycles elapsed. It rolls over when 1022 chips is reached. Reading this register returns the counter value at the time of accessing. It is reset at the following sync80m strobe after GPS mode is armed or when a software reset command is issued. This register is initialized at the same time when *CDMA_RTC_SYNC* is updated. Information inside this register together with *CDMA_RTC_SYNC* register yields the absolute GPS system time.

11 Receive Front-End

11.1 ARM registers

The registers in this section are accessible only by the ARM.

The Receive front-end block consists of two duplicate sets of registers, one for each Rx chain. Rather than repeat identical descriptions for each entry, the following list uses notation (“c”) that defines the register as belonging to either Rx chain 0 or Rx chain 1. The “c,” representing 0 or 1, appears at the end of the first descriptive word in the register name, and thus should be interpreted as, for example, RXF0/RXF1.

The two sets of registers are separated by 512 bytes of address space, or 0x200 hexadecimal. Each Rx front chain has been assigned 128 Word addresses, or 128 possible registers each. Thus the offset from Rx0 to Rx1 should be 128 words, or 512 bytes. Each address is notated using the offset plus 0x200*c, where c is the 0 or 1 chain.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block names: RXFRONT0, RXFRONT1
- Number of word addresses: 128 for each block
- Byte Address:
  - RXFRONT0 = 0x0A00 - 0x0BFC
  - RXFRONT1 = 0x0C00 - 0x0DFC
- CHIP_BASE = 0x80000000
- chipaddr range = 13:2

For example, the RXF0_RESET register (RXFc_RESET register for chain 0, c = 0) is at address 0x0A00, while the RXF1_RESET register (RXFc_RESET register for chain 1, c = 1) is at address 0x0C00. The RXF0_CTL register (RXFc_CTL register for chain 0, c = 0) is at address 0x0A04, while the RXF1_CTL register (RXFc_CTL register for chain 1, c = 1) is at address 0x0C04.

11.1.1 Rx Front registers

0x0+0x200c RXFc_RESET, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK
Reset State: 0x1

This register is used to hold the rx_front block in reset during configuration. It is reset to '0' anytime the asynchronous reset input from the clock block (clk) is asserted (for example after a hardware reset).

Bits	Name	Description
0	DATA	Set (1) to hold the entire rx_front chain in reset (including the rx_agc block). Clear (0) to remove the rx_front chain from reset. This register is cleared (0) if the asynchronous reset input to rx_front from the clock block (clk) is asserted.

0x4+0x200c Reserved.

0x8+0x200c Reserved.

0xC+0x200c Reserved.

0x10+0x200c Reserved.

0x14+0x200c Reserved.

0x18+0x200c RXFc_DVGA_CTL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

These are parameters used by the digital variable gain amplifier to set appropriate gain levels.

RXFc_DVGA_CTL

Bits	Name	Description
20:19	DIGITAL_VGA_GAIN_SEL	Selects which gain value is used by DVGA. This must change with mode and should match the RX_AGCc_MODE_CTL. 00: undefined 01: CDMA 1X Gain (also used in GPS) 10: HDR Gain 11: HDR FLC mode (Dual DVGA uses HDR and CDMA 1X Gains)

RXFc_DVGA_CTL (Continued)

Bits	Name	Description
18:4	VGA_GAIN_OFFSET	Micro-programmable Digital VGA gain offset, 15-bit two's complement format. With a resolution of (input resolution / (2*vga_gain_scale)).
3:0	VGA_GAIN_SCALE	Micro-programmable Digital VGA gain scale, 4-bit positive number format. This is chosen to create a constant resolution of 1/120 dB for every gain value and mode.

0x1C+0x200c Reserved.

0x20+0x200c Reserved.

0x24+0x200c Reserved.

0x28+0x200c Reserved.

0x2C+0x200c Reserved.

0x30+0x200c Reserved.

0x34+0x200c Reserved.

0x38+0x200c Reserved.

0x3C+0x200c Reserved.

0x40+0x200c Reserved.

0x44+0x200c Reserved.

0x48+0x200c Reserved.

0x4C+0x200c Reserved.

0x50+0x200c Reserved.

0x64+0x200c Reserved.

0x78+0x200c Reserved.

0x7C+0x200c Reserved.

0x80+0x200c Reserved.

0x84+0x200c Reserved.

0x88+0x200c Reserved.

0x8C+0x200c Reserved.

0x90+0x200c Reserved.

0x98+0x200c Reserved.

0xAC+0x200c Reserved.

0xB0+0x200c Reserved.

0xB4+0x200c Reserved.

11.1.2 Rx AGC Registers

0xC0+0x200c RX_AGCc_RESET, c=[0,1]

Type: Write/Command
Clock: RX_FRONT_CLK

Bits	Name	Description
0	AGC_SW_RESET	Writing a 1 to this bit causes the corresponding agc chain to be reset. The chain will not be held in reset. Writing a 0 to this bit has no affect.

0xC4+0x200c RX_AGCc_MODE_SEL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
1:0	AGC_MODE	These bits specify the mode in which RX AGC will function. 00 : Inactive 01 : CDMA 1x/GPS Mode (Fast)32 chip updates 10 : HDR HLC Mode (Slow)1024 chip updates 11 : HDR FLC Mode (Hybrid)32 and 1024 chip updates

0xC8+0x200c RX_AGCc_LGLUT_LVAL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register holds the lower 8 values of the LOOP GAIN LUT. The LUT is used in HDR FLC Mode only (hybrid mode)

Bits	Name	Description
31:28	LOOP_GAIN_LUT_7	Loop gain LUT entry 7
27:24	LOOP_GAIN_LUT_6	Loop gain LUT entry 6
23:20	LOOP_GAIN_LUT_5	Loop gain LUT entry 5
19:16	LOOP_GAIN_LUT_4	Loop gain LUT entry 4
15:12	LOOP_GAIN_LUT_3	Loop gain LUT entry 3
11:8	LOOP_GAIN_LUT_2	Loop gain LUT entry 2
7:4	LOOP_GAIN_LUT_1	Loop gain LUT entry 1
3:0	LOOP_GAIN_LUT_0	Loop gain LUT entry 0

0xCC+0x200c RX_AGCc_LGLUT_HVAL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register holds the upper 8 values of the LOOP GAIN LUT. The LUT is used in HDR FLC Mode only (hybrid mode)

Bits	Name	Description
31:28	LOOP_GAIN_LUT_15	Loop gain LUT entry 15
27:24	LOOP_GAIN_LUT_14	Loop gain LUT entry 14
23:20	LOOP_GAIN_LUT_13	Loop gain LUT entry 13
19:16	LOOP_GAIN_LUT_12	Loop gain LUT entry 12
15:12	LOOP_GAIN_LUT_11	Loop gain LUT entry 11
11:8	LOOP_GAIN_LUT_10	Loop gain LUT entry 10
7:4	LOOP_GAIN_LUT_9	Loop gain LUT entry 9
3:0	LOOP_GAIN_LUT_8	Loop gain LUT entry 8

0xD0+0x200c RX_AGCc_GAIN_CTL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register specifies the CDMA AGC loop gains during CDMA 1X and GPS modes (32 chip updates).

RX_AGCc_GAIN_CTL

Bits	Name	Description
5:4	GAIN_CONSTANT_IM	These bits control the CDMA AGC loop time constant during IM times. Bit values [7:6] — Time constant (s) 11 — 73 10 — 98 01 — 196 00 — Use GAIN_CONSTANT value

RX_AGCc_GAIN_CTL (Continued)

Bits	Name	Description
3:0	GAIN_CONSTANT	These bits control the CDMA AGC loop time constant. Bit values [5:2] — Time constant(s) 1111 — 73 0000 — 98 0001 — 147 0010 — 196 0011 — 294 0100 — 392 0101 — 587 0110 — 783 0111 — 1175 1000 — 1566 1001 — 2349

0xD4+0x200c RX_AGCc_LNA_CTL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register controls the various modes of the LNA state machine.

RX_AGCc_LNA_CTL

Bits	Name	Description
6	LNA_RANGE_OVERRIDE_N	■ Clearing (0) this bit disables the final LNA Range output (mapped from the LNA Range Decision) and replaces it with the value of LNA_RANGE in RX_AGCc_LNA_DATA register. ■ Setting (1) this bit enables the final LNA_Range output to be mapped normal from the LNA Range decision.
5	LNA_DECISION_OVERRIDE	■ Setting (1) this bit disables the hard LNA Range Decision block output and replaces it with the value of LNA_DECISION in the RX_AGCc_LNA_DATA register. ■ Clearing (0) this bit enables the hard LNA Range decision block.
4	LNA_RANGE_FILT_SEL	■ Setting (1) this bit routes the output of the LNA_FILT block to the input of the hard decision LNA control block. ■ Clearing (0) this bit feeds an unmodified AGC_VALUE to the input of the LNA control block.
3	AGC_VALUE_OVERRIDE	■ Set (1) this bit to freeze the AGC_VALUE accumulator in the loop gain integrator and permit writing it directly via AGC_VALUE_WR. ■ Clear (0) this bit to permit normal operation of the AGC_VALUE accumulator.

RX_AGCc_LNA_CTL (Continued)

Bits	Name	Description
2	LNA_FILT_OVERRIDE	■ Set (1) this bit to hold the value on LNA_FILT and provide override capability of LNA_FILT via LNA_FILT_WR. ■ Clear (0) this bit for normal automatic LNA_FILT operation.
1:0	LNA_FILT_BW	These bits define the behavior of the low-pass filter structure, which provides input information for possible processing by the hard LNA decision block. The following bit values select the corresponding time constant for the filter: LNA_FILT_BW — Time constant 00 — 1.6 ms 01 — 3.3 ms 10 — 6.7 ms 11 — 13.3 ms

0xD8+0x200c RX_AGCc_LNA_DATA, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
6:4	LNA_RANGE	When RX_AGCc_LNA_CTL:LNA_RANGE_OVERRIDE_N is cleared (0), the final LNA Range output is overridden with this value. For normal operation, RX_AGCc_LNA_CTL:LNA_RANGE_OVERRIDE_N should be set to '1'. The valid values for LNA_RANGE are "000", "001", "010", "011" and "111", corresponding to LNA stage from one to five.
3:0	LNA_DECISION	Overrides the LNA_DECISION output (states of LNA state machine) of the LNA_STEP_GAIN block when RX_AGCc_LNA_CTL:LNA_DECISION_OVERRIDE bit is set (1). The valid values for it are "0000", "0001", "0011", "0111" and "1111", corresponding to an LNA stage from 0 to 4.

0xDC+0x200c RX_AGCc_VALUE_WR, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	The AGC_VALUE_WR register can be used to write the AGC accumulator value in the loop gain integrator when the RX_AGCc_LNA_CTL:AGC_VALUE_OVERRIDE bit is set.

0xE0+0x200c RX_AGCc_LNA_FILT_WR, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	The LNA_FILT_WR register can be used to write the lna filter accumulator value in the LNA_FILT block when the RX_AGCc_LNA_CTL:LNA_FILT_OVERRIDE bit is set.

0xE4+0x200c RX_AGCc_DC_GAIN, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register provides the AGC_DC_GAIN value and the enable bit. The default value is'0' after reset.

Bits	Name	Description
6	AGC_VALUE_MIN_EN	Set (1) this bit to limit the minimum output from the AGC_VALUE accumulator based on the values in RX_AGCc_VALUE_n_MIN.
5	AGC_LOOP_FREEZE_EN	Set'1' to freeze agc loop, which means gating rssi_data input to the Loop integrator to be 0.
4	AGC_DC_GAIN_EN	Set'1' to enable AGC loop gain to switches to AGC_DC_GAIN_VAL when agc_dc_gain_sel port is asserted as'1' by the DC offset cancellation block.
3:0	AGC_DC_GAIN_VAL	AGC loop gain value when DC loop is in acquisition mode. If set to'1001', the time constant reaches its maximum value '2349 us'. See the description of GAIN_CONSTANT in RX_AGCc_GAIN_CTL.

0xE8+0x200c RX_AGCc_VALUE_n_MIN, c=[0,1], n=[1..4] +4n-4

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
6:0	DATA	These registers provide the minimum power output from the AGC accumulator when the RX_AGCc_DC_GAIN:AGC_VAL_MIN_EN bit is set and the five-stage LNA is in it's nth stage (stages numbered 0-4). It has a resolution of 1/3 dB (in the 85.3-dB mode) or 4/10 dB (in the 102.4-dB mode). The minimum value is 00, and the maximum value is 7F. The implicit sign bit is '1'.

0xF8+0x200c RX_AGCc_VALUE_MAX, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

The AGC_VALUE_MAX register provides the enable and maximum acceptable value for the AGC_VALUE accumulator.

Bits	Name	Description
7	AGC_VALUE_MAX_EN	Setting (1) this bit enables this register to limit the maximum output from the AGC_VALUE accumulator.
6:0	AGC_VALUE_MAX	When AGC_VALUE_MAX_EN=1, this register provides the maximum power output from the AGC_VALUE accumulator. For the bit alignment, a zero sign bit is appended to this register to form an eight bit quantity, which is left justified to limit the AGC_VALUE output.

0xFC+0x200c RX_AGCc_IM_LEVELn, c=[0,1], n=[1..4] +4n-4

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
7:0	DATA	These registers contain an 8-bit IM anti-jamming threshold data in 2's complement format for the nth stage of the five-stage LNA (stages numbered 0-4). It has a resolution of 1/3 dB (85.3 dB mode) or 4/10 dB (102.4 dB mode). Minimum value is 80. Maximum value is 7F.

0x10C+0x200c+4n-4 RX_AGCc_LNA_n_FALL, c=[0,1], n=[1..4]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
7:0	DATA	These registers contain an 8-bit falling threshold in 2's complement format. The threshold is for the n-1 stage when the five-stage LNA is in a stage greater than n-1 (stages numbered 0-4). This register has a resolution of 1/3 dB (85.3 dB mode) or 4/10 dB (102.4 dB mode). Minimum value is 80. Maximum value is 7F.

0x11C+0x200 RX_AGCc_LNA_n_RISE, c=[0,1], n=[1..4] c+4n-4

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
7:0	DATA	These registers contain an 8-bit rising threshold in two's complement format. The threshold is for the nth stage when the five-stage LNA is in a stage lower than n (stages numbered 0-4). This register has a resolution of 1/3 dB (in the 85.3-dB mode) or 4/10 dB (in the 102.4 dB- mode). The minimum value is 80, and the maximum value is 7F.

0x12C+0x200 RX_AGCc_LNA_n_OFFSET, c=[0,1], n=[1..4] c+4n-4

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	These registers contain the offset value in unsigned format (positive number) to be subtracted from the agc value when the 5-stage LNA is in it's nth stage (stages numbered 0-4). It has a resolution of 1/12 (85.3 dBmode) and 1/10 (102.4 dB mode). Minimum value is 000. Maximum value is 3FF.

0x13C+0x200 RX_AGCc_LNA_BP_TIMER_n, c=[0,1], n=[0..3] c+4n

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
12:0	DATA	These registers provide the "bypass" timer values for the five-stage LNA. The timer value is for transitions to stage n+1 from stages lower than stage n+1 (stages numbered 0-4). The timer range is from 0 to 214 ms with resolution of about 26 us.

0x14C+0x200 RX_AGCc_LNA_NBP_TIMER_n, c=[0,1], n=[0..3] c+4n

Type: Write
Clock: CDMA_AGC_CLK

Bits	Name	Description
12:0	DATA	These registers provide the "non-bypass" timer values for the five-stage LNA. The timer value is for transitions to stage n from stages higher than stage n (stages numbered 0-4). The timer range is from 0 to 214 ms with resolution of about 26 us.

0x15C+0x200*c RX_AGCc_LNA_RANGE_DELAY, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
5:0	DATA	This register provides the delay time for LNA_DECISION to switch LNA_OFFSET after an lna state change. When the digital filter is used, the LNA offset to the AGC_VALUE needs to be delayed to compensate for the SBI transaction delay and the ZIFRIC LNA switching delay. The timer start time is determined by the AGC_SBI_BYPASS bit in the RX_AGCc_SBI_CTL register. It is chipx2 resolution. When the AGC_SBI_BYPASS bit is cleared (normal operation), 0 is not a valid value (value must be greater than 0).

0x160+0x200*c RX_AGCc_SBI_CTL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register contains SBI control bits. The default values of all bits are ‘0’ after reset.

Bits	Name	Description
5:3	AGC_SBI_LNA_RANGE	Contains the lna range value used to generate sbi transactions (both requests and packet data) and Vref outputs when the AGC_SBI_LNA_OVERRIDE bit is set(1).
2	AGC_SBI_LNA_OVERRIDE	When set (1), the lna_range value used to generate sbi transactions (both requests and packet data) and Vref outputs is overridden with the value in AGC_SBI_LNA_RANGE.
1	AGC_SBI_BYPASS	When set (1), the programmable delays for LNA_OFFSET and Vref are triggered whenever the respective lna state input changes, otherwise they are triggered by AGC_SBI_DONE signal (which indicates the completion of an rx_agc SBI transaction). This bit should be cleared when AGC SBI transmission is enabled.
0	AGC_SBI_EN	Set (1) to enable the AGC SBI interface.

0x164+0x200c RX_AGCc_SBI_PACKET_DATA, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

This register provides the SBI address and SBI data for LNA gain stages.

Bits	Name	Description
25	SBI_PACKET_ADDR	The SBI address for ZIF and LNA in Interrupt Transfer Mode
24:20	SBI_PACKET_4	SBI packet data when the five-stage LNA is in its fifth stage (stage 4, “111”)
19:15	SBI_PACKET_3	SBI packet data when the five-stage LNA is in its forth stage (stage 3, “011”)
14:10	SBI_PACKET_2	SBI packet data when the five-stage LNA is in its third stage (stage 2, “010”)
9:5	SBI_PACKET_1	SBI packet data when the five-stage LNA is in its second stage (stage 1, “001”)
4:0	SBI_PACKET_0	SBI packet data when the five-stage LNA is in its first stage (stage 0, “000”)

0x168+0x200c RX_AGCc_VREF_DELAY_TIMER, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK

Bits	Name	Description
5:0	DATA	This register provides the delay time for the Vref output to switch after an lna state change. The Vref switch needs to be delayed to compensate for the SBI transaction delay and the ZIFRIC LNA switching delay. The timer start time is determined by the AGC_SBI_BYPASS bit in the RX_AGCc_SBI_CTL register. It is chipx2 resolution. When the AGC_SBI_BYPASS bit is cleared (normal operation), 0 is not a valid value (must be greater than 0).

0x16C+0x200*c RX_AGCc_VREF_VAL, c=[0,1]

Type: Write
Clock: RX_FRONT_CLK
Reset State: 0x00

This register contains the Vref values for each particular LNA stage. This register is reset to all 0's after the asynchronous reset input from the clock block (clk) is asserted (NOT affected by the RXFc_RESET or RX_AGCc_RESET registers).

Bits	Name	Description
4	VREF_VAL_4	Vref value when the five-stage LNA is in its fifth stage (stage 4, "111").
3	VREF_VAL_3	Vref value when the five-stage LNA is in its fourth stage (stage 3, "011").
2	VREF_VAL_2	Vref value when the five-stage LNA is in its third stage (stage 2, "010").
1	VREF_VAL_1	Vref value when the five-stage LNA is in its second stage (stage 1, "001").
0	VREF_VAL_0	Vref value when the five-stage LNA is in its first stage (stage 0, "000").

0x170+0x200*c Reserved.

0x174+0x200*c RX_AGCc_1X_VALUE_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	This register returns the current AGC VALUE output of the accumulator. The value returned is the 32-chip agc value (agc_value_1x) in cdma1x/gps mode, the 1024-chip agc value (agc_value_hdr written by the dsp) in HDR HLC mode, and the 32-chip agc value in HDR FLC mode (which could be the 1024-chip agc value or the 32-chip agc value, whichever is greater).

0x178+0x200c RX_AGCc_HDR_VALUE_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	This register returns the current HDR AGC_VALUE written by the mDSP (updated every 1024 chips).

0x17C+0x200c RX_AGCc_1X_VGA_GAIN_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	This register returns the CDMA_VGA_GAIN value (cdma1x/gps vga_gain value updated every 32 chips) sent to the dual DVGA block from rx_agc.

0x180+0x200c RX_AGCc_HDR_VGA_GAIN_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	This register returns the HDR VGA_GAIN value (updates every 1024 chips based on mDSP updates of HDR AGC_VALUE) sent to the dual DVGA block from rx_agc.

0x184+0x200c RX_AGCc_LNA_FILT_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
9:0	DATA	This register returns the lna filter accumulator value.

0x188+0x200c RX_AGCc_LNA_RANGE_RD, c=[0,1]

Type: Read
Clock: RX_FRONT_CLK

Bits	Name	Description
2:0	DATA	This register returns the lna range value output from the lna state machine. Bit 2 is the mixer range value and bits 1:0 are the lna range value.

12 Searcher

12.1 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: SRCH
- Number of word addresses: 64
- Byte Address: 0x6300 - 0x63FC
- OFFSET=CHIP_BASE+0x6300
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

12.1.1 Search write registers

0x0000 SRCH_ACC_PASS

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
2:0	DATA	The SRCH_ACC_PASS register is the energy accumulation passes register for the searcher engine. This register specifies the number of energy values to be non-coherently accumulated at each offset. After the despread data is integrated, the integrated data for the I and Q channels is squared and summed to obtain the energy data. The number of energy values to be accumulated is equal to the value written to this register, plus one. The functional values of this register range from 0 to 7, corresponding to the 1 to 8 energy values summed.

0x0004 SRCH_CTL

Type: Write
Clock: SRCHCHIPX8_CLK

The SRCH_CTL register is the search control register.

Bits	Name	Description
5	SEARCH_SPEED	This bit selects the speed of the search operations. ■ When this bit is set (1), the searcher searches the designated PN space eight times faster than the MSM2.2 searcher (when not using early dump capabilities). ■ When this bit is clear (0), the search speed is the same as the MSM2.2 searcher. **Search speed** **Performance level** 0 1x 1 8x
4	DMA_DISABLE	Internal use only ■ Setting (1) this bit disables searcher DMA requests using 1x speed. ■ Clearing (0) this bit enables searcher DMA requests using 1x speed. DMA is not available using 8x speed. Clear (0) this bit.
3	STATUS_DUMP	This bit initiates a status dump command without a search operation, or before a search operation is completed. When this bit is set (1), a SEARCH_DONE interrupt is generated within 15 microseconds.
2	SEL_ANT	This bit selects the antenna to search data from. 0 : antenna 0 1 : antenna 1
1:0	SEARCHER_GAIN	This bit selects the gain value for the searcher engine. The gain value is a multiplier for the despread data that is generated before the data is accumulated. Bit [1:0] Gain value 00 : 1/16 01 : 2/16 10 : 4/16 11 : 8/16 For no RX diversity case, 8x searcher SEARCHER_GAIN for 0 (x 1/16), 1 (x 2/16), 2 (x 4/16), 3 (x 8/16) is equivalent to offline searcher SCALING 4 (x 1/32), 3 (x 1/16), 2 (x 1/8), 1 (x 1/4), respectively, because the 8x searcher despreader input is 4 bits offset 2's complement, instead of 5 bits 2's complement in offline searcher.

0x0008 SRCH_INTG_TIME

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
7:0	DATA	The SRCH_INTG_TIME register specifies the number of despread data samples to be coherently integrated at each offset before the data is converted to an energy value. The number of PN chips to be integrated is equal to the value written to this register multiplied by four. The values for this register range from 3 to 255, which corresponds to 12 to 1020 PN chips.

0x000C SRCH_MASK_I

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
14:0	DATA	The SRCH_MASK_I register is the I channel pilot PN mask register for the searcher engine. This register defines the I channel pilot PN mask of the searcher engine. Bit i of this register carries out XOR operations with the searcher engine I channel pilot PN generator, state $x^{(i+1)}$. State x^i is the signal state of the I channel pilot PN generator stage i, where i ranges from 0 to 14.

0x0010 SRCH_MASK_Q

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
14:0	DATA	The SRCH_MASK_Q register is the Q channel pilot PN mask register for the searcher engine. This register defines the Q channel pilot PN mask of the searcher engine. Bit i of this register carries out XOR operations with the searcher engine Q channel pilot PN generator, state $x^{(i+1)}$. State x^i is the signal state of the Q channel pilot PN generator stage i, where i ranges from 0 to 14.

0x0014 SRCH_MAX_SELECT

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
1:0	DATA	The SRCH_MAX_SELECT register is the local maxima select register for the searcher engine. This register selects one of four local maxima energy and offset pairs to be returned by the SEARCH_MAX_ENERGY and SEARCH_MAX_INDEX registers. The possible values to select for the SRCH_MAX_SELECT register are as follows: **Bit[1:0] value** **Operation** 00 : Best search result 01 : Second best search result 10 : Third best search result 11 : Fourth best search result

0x0018 SRCH_MAX_ENERGY

Type: Read
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
15:0	DATA	The SRCH_MAX_ENERGY register is the maximum energy value for the searcher engine. This register contains the SRCH_MAX_SELECT, which is the local maxima energy in the search window for the last completed search. Results are valid from the assertion of SRCH_DONE_INT until the start of the next search.

0x001C SRCH_MAX_INDEX

Type: Read
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
15:0	DATA	The SRCH_MAX_INDEX register is the Maximum Energy Index Value for the Searcher Engine. This register contains the index value for the SEARCH_MAX_SEL, which is the largest local maxima offset in the search window for the last completed search. This register returns the search index which is the offset relative to the earliest offset in the search window with units of 1/2 PN chips per LSBit. Values are valid from the assertion of SEARCH_DONE_INT until the start of the next search.

0x0020 SRCH_NUM

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
15:0	DATA	The SRCH_NUM register is the search number register for the searcher engine. This register specifies the number of offsets to search in the next search operation; it specifies the size of the search window. At each offset, the searcher engine processes both the on-time and late energy values. The late energy value is an energy calculation with the pilot PN sequence delayed 1/2 chip from the on-time energy calculation. After processing at an offset is complete, the searcher engine slews to the next offset by retarding the timing by one PN chip (positive slew). When writing to SEARCH_NUM, clear (0) the LSB. The remaining bits have a 1 PN chip resolution. SEARCH_DONE_INT is asserted after all the processing for the specified number of offsets is completed. When the searcher is active, clearing this register aborts a search in progress. When the searcher is idle, clearing (0) this register, and writing a value to SEARCH_SLEW, slews the searcher engine without initiating a search command. The maximum width of the search window is 32767 PN chips. To use an example, if the search window is 128 (0x80) PN chips, write 0x0100 to SEARCH_NUM.

0x0024 SRCH_OFFSET

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
14:0	DATA	The SRCH_OFFSET register is the binary offset register for the searcher engine. This register specifies a representation of the pilot PN sequence offset of the particular base station to be searched. This register specifies the pilot PN offset, in units of PN chips, for both the I and Q channels. The SRCH_OFFSET, SEARCH_PN_I_MASK, and SEARCH_PN_Q_MASK registers together account for the pilot PN sequence offset. Program this register with [Pilot_channel_offset x 64] –2, where Pilot_channel_offset is 0...511; in accordance with IS-95-A, section 7.1.3.2.1.

0x028 SRCH_QOF_SEL

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
1:0	DATA	The SRCH_QOF_SEL register is the pilot QOF value for the 8X searcher. This register was added to support an auxiliary pilot with QOF searching.

0x002C SRCH_SLEW

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
18:2	DATA	The SRCH_SLEW register is the slew value register for the searcher engine. The slew value register directs the searcher engine to perform a slew from its current position. This register must be written to start a search operation, even if slewing zero offsets. The search operation begins at the offset the searcher slews to and ends at the last offset in the search window, plus one PN chip. A zero-valued slew starts the search operation where the last search operation left off. All searches start from the search window's earliest offset (the offset slewed to) and move to the latest offset. The slew value is specified in units of 1/2 PN chips relative to the searcher's current position. Write a 2's complement number to this register. The maximum positive slew, which retards the timing, is 32767 1/2 PN chips and the maximum negative slew, which advances the timing, is 32768 PN chips.
1:0	RESERVED_BITS1_0	These bits have no effect when written to.

0x0030 SRCH_TH_ENERGY

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
15:0	DATA	The SRCH_TH_ENERGY register specifies the energy dump threshold for the searcher engine. During coherent integration, the energy value of the intermediate integrated data is calculated. This intermediate energy is then compared with the dump threshold energy that is specified by this register. If the intermediate energy value is greater than the threshold, the integration continues. Otherwise, the searcher engine retards to the next offset.

0x0034 SRCH_TH_TIME

Type: Write
Clock: SRCHCHIPX8_CLK

The SRCH_TH_TIME register is the early dump integration time register for the searcher engine. This register specifies the number of despread data values to be coherently integrated at each offset before the data is converted to intermediate energy value. At an intermediate point in the coherent integration, which is set by SRCH_TH_TIME, the energy value of the intermediate coherently integrated data is calculated. This intermediate energy value is then compared with the dump threshold energy specified by the SRCH_TH_ENERGY register. If the intermediate energy value is within either dump threshold energy, the integration continues. Otherwise, the searcher engine immediately retards to the next offset. The values of this register range from 4 to (SEARCH_INTEGRATE_TIME – 2). The corresponding value is 16 to (SEARCH_INTEGRATE_TIME *x* 4 – 8) PN chips.

Bits	Name	Description
7:0	DATA	This register is used only when the searcher is in 1x mode, such that the SEARCH_SPEED bit of the SRCH_CTL register is clear (0).

0x0038 SRCH_WALSH_NUM

Type: Write
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
8:0	DATA	The SRCH_WALSH_NUM register is the pilot Walsh channel number for the 8x searcher.

Reserved.

0x003C SRCH_DMA_DATA

Type: Read
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
15:0	DATA	The SRCH_DMA_DATA register is the DMA data for the searcher engine. This register returns the energy obtained for each offset evaluated in the search window through a DMA transfer mechanism when DMA is enabled (DMA_DISABLE=0), and searcher is running at 1x speed. For each DMA assertion, energy obtained for two offsets spaced a 1/2 PN chip apart are transferred in the following order: lower byte on time; upper byte on time; lower byte late; upper byte late. Zero energy is reported for those offsets that the searcher early dumped. The first DMA assertion corresponds to the first, earliest offset searched and proceeds to the latest offset in the search window. DMA is not available using 8x speed.

Reserved.

0x0040 SRCH_DMA_ERROR

Type: Read
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
0	DATA	The SRCH_DMA_ERROR register is the searcher DMA error register. This register indicates the error status of the DMA port. ■ When set (1), it indicates that the existing set of search results at an offset was not read before the next set of search results arrived. ■ When clear (0), no error has occurred. This register is automatically cleared (0) after a read.

Reserved.

0x0044 SRCH_POSITION

Type: Read
Clock: SRCHCHIPX8_CLK

Bits	Name	Description
17:0	DATA	The SRCH_POSITION register is bits 17:0 of the time offset position for the searcher engine. This register captures the PN offset of the searcher engine at the most recent status dump. After an EPOCH command is received, the slewing control hardware for the searcher engine is reset. The position of the searcher engine (relative offset of the pilot PN sequence) is adjusted whenever the searcher engine slews to a new position. This slewing occurs from microprocessor-directed slewing (writes to the SRCH_SLEW register) or by automatic slewing during search operations. The cumulative change in the searcher's position is recorded and can be read from this register. This change is in reference to the searcher engine's original position when the most recent EPOCH command was generated. The value read from this register is in units of 1/2 PN chips; the value is always positive and the maximum value is 32767 1/2 PN chips. The value returned from this register is in reference to the most recent EPOCH time stamp and is not a true time offset.

13 1X Demodulator

13.1 ARM registers

The ARM registers that follow are divided by clock into two main sections :

- DEMOD_1X_RXCHIPX8 registers:
  - Offset = CHIP_BASE+0x6400 MAX=CHIP_BASE+0x64FC
- DEMOD_1X_OFFLINE registers:
  - OFFSET=CHIP_BASE+0x6500 MAX=CHIP_BASE+0x68FC

Addresses in this section are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: DEM1X
- Number of word addresses: 320
- Byte Address: 0x6400 - 0x068FC (total area)
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

13.1.1 RXCHIPX8 clock registers

0x0000 DEM1X_RESET

Type: Write
Clock: RXCHIPX8_CLK

The rxchipx8 clock must be running for the software reset to be recognized by the reverse link subsystem. Similarly, the FFE block cannot be reset unless the offline clock is running.

Bits	Name	Description
0	RESET	When set(1), all 1X demodulator blocks are reset (equivalent to setting all other bit). Write ‘0’ to clear the reset.

0x0004 DEM1X_SYNC

Type: Write (command)
Clock: RXCHIPX8_CLK

Generates simultaneous commands to the offline reference counter, combiner timer and long code generator. It enables software to synchronize demodulator timing to support processing of 'offline' samples.

Bits	Name	Description
2	REF_COUNT	When set (1), the offline reference counter (a replacement for the RTC) starts counting from a programmed starting value (see DEM1X_OFFLINE) when in offline mode.
1	LC_STATE	When set (1), commands the long code generator to load the pending long code state.
0	COMB_TIME	When set (1), commands the combiner timer to load the pending count value.

0x0008 DEM1X_LATCH

Type: Write (command)
Clock: RXCHIPX8_CLK

This command latches the RTC, long code state and combiner time in the cycle in which the command is received. This enables software to determine the offset of combiner time from the RTC and, hence, any finger.

Bits	Name	Description
2	REF_COUNT	Latch the reference counter in the current cycle. In offline mode this is the offline counter, otherwise it is the RTC.
1	LC_STATE	Latch the long code state in the current cycle.
0	COMB_TIME	Latch the combiner time in the current cycle.

0x000C DEM1X_TRACKING

Type: Read/Write
Clock: RXCHIPX8_CLK

Specifies the finger that the symbol combiner uses for slamming and time tracking.

Bits	Name	Description
3:1	FINGER	Specifies the finger number, when enabled 000 : 0 001 : 1 010 : 2 011 : 3 100 : 4 101 : 5 Reset state: X
0	ENABLE	When set, the tracking finger is specified by the FINGER field. Reset state: 0.

0x0010 DEM1X_FRAME_OFFSET

Type: Read/Write
Clock: RXCHIPX8_CLK

Bits	Name	Description
3:0	DATA	Specifies the frame offset in units of PCG (0 - 15). It is used to compute the frame count and establish frame boundaries. Reset state: X

0x0014 Reserved.

0x0018 DEM1X_OFFLINE

Type: Read/Write
Clock: RXCHIPX8_CLK

Specifies the start and stop value of the offline reference counter. In offline mode, the free-running RTC is replaced by the offline reference counter. This counter is used to replay a time window for which there are stored samples. For example, assume that the Sample Controller stored samples from RTC time A to time B. The 1X demodulator can process the samples stored between A and B at a later time by setting enabling offline mode, the START_VAL = A and STOP_VAL = B and then issuing a DEM1X_SYNC:REF_COUNT.

Bits	Name	Description
30	ENABLE	When set (1), the offline counter is used instead of the online RTC. On reset this mode is cleared and the online RTC provided by the RTC block is used. Reset state: 0
29:15	START_VAL	Denotes the start value of the offline counter (in cx1 units). This value is latched into the offline counter when written. When the DEM1X_LOAD_TIME_CMD is asserted this counter is enabled and starts counting from this value. Reset state: X
14:0	STOP_VAL	Specifies the terminal count (in cx1 units). When the offline counter hits this value, the offline mode ends and a offline_mode_done interrupt is asserted to the microprocessor. Reset state: X

0x001C DEM1X_REF_COUNT

Type: Read
Clock: RXCHIPX8_CLK

Bits	Name	Description
17:0	LATCHED	Returns the reference counter value from the cycle in which the DEM1X_LATCH_TIME_CMD:REF_COUNT command is issued. The count is in chipx8 units and spans the short PN sequence. In offline mode, the offline reference counter is latched instead of the RTC. Reset state: X

0x0020+4n DEM1X_FINGERn, n = [0,5]

Type: Read/Write
Clock: RXCHIPX8_CLK

A write to this register enables or disables finger n. When enable is set (1), the finger is immediately assigned the specified finger PN position. The PN position, to chipx8 resolution, is the time offset from the reference counter (either RTC or offline reference counter).

Software may modify the PN_POSITION while the finger is enabled.

When finger n is enabled or when software modifies its PN position while enabled, firmware is instructed to re-initialize finger n processing.

A read returns the current PN position of finger n and reflects any time-tracking changes made to the finger by firmware.

Bits	Name	Description
18:1	PN_POSITION	PN position is a negative time offset (in units of 1/8th of a PN chip) from the timing reference provided by the reference counter (RTC or offline ref. count). A larger PN position corresponds to a larger delay of the corresponding tracked multi-path component. Reset state: X
0	ENABLE	When set (1), finger processing for finger n is enabled. Reset state: 0

0x0038 DEM1X_COMBINER_TIME_LOAD

Type: Read/Write
Clock: RXCHIPX8_CLK

Specifies a value to be loaded into the combiner time counter. Issuing the DEM1X_ARM_COMBINER_SLAM_CMD causes this value to be loaded on the PN roll from the finger specified by DEM1X_TRACKING_FINGER. This value is also loaded when the DEM1X_SYNC_CMD:COMB_TIME command is issued. The recommended slam value to be written is 0xBF3E8, which is 387 chips before an 80ms roll.

Bits	Name	Description
20	TIME80MS	Combiner time in 80ms (=0 first 80ms period, =1 second 80ms period). Reset state: X
19:0	TIMECHIPX8	Combiner time in chipx8 units, counting zero to 80ms.

0x003C DEM1X_ARM_COMBINER_SLAM

Type: Write (command)
Clock: RXCHIPX8_CLK

The slam value for the combiner time establishes the combiner delay from the tracking finger.

Bits	Name	Description
0	SET	When set (1), the slam is armed. The combiner time is slammed with the DEM1X_COMBINER_TIME_LOAD value on the PN roll from the finger specified by DEM1X_TRACKING. The slam is disarmed by the PN roll or reset.

0x0040 DEM1X_SLEW_COMBINER_TIME

Type: Write (command)
Clock: RXCHIPX8_CLK

This command slews the combiner time by a specified value. It takes effect immediately. For example, if the slew command is applied at combiner time T then the combiner time is T+offset+1 in the next cycle.

NOTE The combiner frame counter is automatically updated but not the long code state.

Bits	Name	Description
19:0	OFFSET	Offset in chipx8 units, ranging from 0 to 80ms, to be added immediately to the combiner time.

0x0044 DEM1X_LC_STATE_LOAD_LO

Type: Read/Write
Clock: RXCHIPX8_CLK

This register specifies the lower 32 bits of the 42 bit long code state to be loaded. This state is loaded in one of two ways: on a combiner PN roll (following a DEM1X_ARM_LC_STATE command) or when the DEM1X_SYNC:LC_STATE command is issued.

Bits	Name	Description
31:0	DATA	Lower 32 bits of long code state Reset state: X

0x0048 DEM1X_LC_STATE_LOAD_HI

Type: Read/Write
Clock: RXCHIPX8_CLK

This register specifies the upper 10 bits of the 42 bit long code state

.

Bits	Name	Description
9:0	DATA	Upper 10 bits long code state. Reset state: X

0x004C DEM1X_ARM_LC_STATE_LOAD

Type: Write (command)
Clock: RXCHIPX8_CLK

This command schedules the long code state to be loaded on the next combiner PN roll.

Bits	Name	Description
0	SET	When set (1), the long code state load is armed. The long code generator is loaded on the next combiner time PN roll. It is disarmed by load or combiner reset.

0x0050 DEM1X_COMBINER_TIME

Type: Read
Clock: RXCHIPX8_CLK

Returns the combiner time from when the DEM1X_LATCH_TIME_CMD:COMB_TIME command is issued.

Bits	Name	Description
20:0	LATCHED	The full 160ms combiner time count in chipx8 units. Reset state: X

0x0054 DEM1X_FRAME_COUNT

Type: Read
Clock: RXCHIPX8_CLK

Returns the frame count from the cycle in which the DEM1X_LATCH_TIME:COMB_TIME command is issued.

Bits	Name	Description
5:4	FRAME	Frame count (0-3) Reset state: X
3:0	PCG	PCG count (0-15) in the current 20ms frame Reset state: X

0x0058 DEM1X_LC_STATE_LO

Type: Read
Clock: RXCHIPX8_CLK

Bits	Name	Description
31:0	LATCHED	Returns the lower 32 bits of the latched long code state. The state is latched on every PN roll or in the cycle that the DEM1X_LATCH:LC_STATE command is received. Reset state: X

0x005C DEM1X_LC_STATE_HI

Type: Read
Clock: RXCHIPX8_CLK

Bits	Name	Description
9:0	LATCHED	Returns the upper 10 bits of the latched long code state. The state is latched on every PN roll or in the cycle that the DEM1X_LATCH:LC_STATE command is received. Reset state: X

0x0060 DEM1X_TRAFFIC_REV_PWR_CTL

Type: Read/Write
Clock: RXCHIPX8_CLK

Bits	Name	Description
3	POWER_CTL_PCT	This bit controls the puncturing behavior of the CDMA power control. ■ To puncture one symbol, set (1) this bit. ■ To puncture two symbols, clear (0) this bit. Changes to this bit take effect immediately. ■ For 9600 bps rate-set operation, clear (0) this bit. ■ For 14,400 bps rate-set operation, set (1) this bit. This bit takes effect immediately.
2	PUNC_POS_SEL	Clear (0) this bit when receiving RC 3 and 5. Set (1) this bit when receiving RC 1, 2, and 4.
1	PC_EVEN	This bit forces the puncturing position for the power control to be aligned with the even PN word. Set (1) this bit when receiving the RC 4 traffic channel. Clear (0) this bit otherwise.
0	ERASE_EN	This bit controls the insertion of erasures into the combined data stream that is sent to the deinterleaver. ■ To insert erasures into the combined data stream for forward traffic channel operation, set (1) this bit. ■ To disable the insertion of erasures into the combined data stream for sync channel and paging channel operation, clear (0) this bit. This bit takes effect immediately after a write.

0x0064 DEM1X_COMMON_REV_PWR_CTL

Type: Read/Write
Clock: RXCHIPX8_CLK

Bits	Name	Description
15:14	PC_RATE_SEL	These bits determine the power control update rates: '00' : 200 bps '01' : 400 bps '10' : 800 bps '11' encoding is undefined.
13:7	INIT_OFFSET2	This bit field is the initial offset for cell B, which is used to compute the effective offset.
6:0	INIT_OFFSET1	This bit field is the initial offset for cell A, which is used to compute the effective offset.

0x0068 DEM1X_FWD_PWR_CTL_IMMED

Type: Write/Read
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Bits	Name	Description
31:16	WALSH_GAIN	The walsh gain is the scaling factor for the supplemental channel Eb/Nt estimate. The scaling factor is given below in Table 13-2 as a function of (non-TD walsh length * MFI factor).
15:0	FPC_SEL	From FPC_SEL.

Table 13-2 Suggested Walsh scaling factors for supplemental channels

non TD walsh length	512	256	128	64	32	16	8	4
RC3	23170	16384	11585	8192	5793	4096	2896	2048
RC4	16384	11585	8192	5793	4096	2896	2048	1448
RC5	18919	13377	9459	6689	4730	3344	2365	1672

0x006C DEM1X_FWD_PWR_CTL_FRAME

Type: Write/Read
Clock: RXCHIPX8_CLK

Bits	Name	Description
31:16	WALSH_GAIN	The walsh gain is the scaling factor for the supplemental channel Eb/Nt estimate. The scaling factor is given in Table 13-2 as a function of (non-TD walsh length * MFI factor).
15:0	FPC_SEL	From FPC_SEL.

0x0078 DEM1X_FREQUENCY_CTL

Type: Read/Write
Clock: RXCHIPX8_CLK

DEM1X_FREQUENCY_CTL

Bits	Name	Description
18	TRK_LO_ADJ_POLARITY	This bit inverts or maintains the signal polarity of the TRK_LO_ADJ pin in CDMA mode. ■ To maintain the default signal polarity of the TRK_LO_ADJ pin, clear (0) this bit. ■ To invert the signal polarity of the TRK_LO_ADJ pin, set (1) this bit. Using the default signal polarity (TRK_LO_ADJ_POL=0), the TRK_LO_ADJ pin operates as follows: If the received baseband pilot signal has a net positive frequency error, CDMA processor decreases the density of high pulses on the TRK_LO_ADJ pin. The opposite is true if TRK_LO_ADJ_POL is set (1).
17	TRK_LO_ADJ_OE	This bit enables or disables the TRK_LO_ADJ pin in CDMA mode. When disabled, the TRK_LO_ADJ pin is placed in a high impedance state. ■ Set (1) this bit to enable the TRK_LO_ADJ pin for normal operation. ■ Clear (0) this bit to disable the TRK_LO_ADJ pin.
16	TRK_LO_ADJ_SEL	This bit selects the source of the 16-bit input to the TRK_LO_ADJ PDM. If cleared (0), the input to the TRK_LO_ADJ_PDM is provided by the ARM through bits 15:0 of the DEM1X_FREQUENCY_CTL. If set (1), the input to the TRK_LO_ADJ PDM is provided by the mDSP through the COMB_TRK_LO_ADJ_WR register.
15:0	TRK_LO_ADJ	The TRK_LO_ADJ bitfield is a micro register that can override the AGC TRK_LO_PDM input. The TRK_LO_ADJ_SEL bit must be clear (0) to enable the override. Note: This register is used only for offline QPCH operation and not for regular use.

0x007C DEM1X_TIME_INT1_MASK

Type: Read/Write
Clock: RXCHIPX8_CLK

Specifies the time of the SYS_TIME_INT1 interrupt from the combiner time.

Bits	Name	Description
10:7	5MS_MASK	This bitfield enables the interrupt strobe for the 5ms frame. Each bit of this register controls a 5ms frame strobe in the 20ms frame. ■ Set (1) a bit to enable the 5ms interrupt strobe. ■ Clear (0) a bit to disable the 5ms interrupt strobe.
6:5	OFFSET_PCG	This bitfield specifies the PCG offset of the interrupt strobe from the 5ms boundary.
4:0	OFFSET_SYMB	This bitfield specifies the 64-chip symbol offset of the interrupt strobe from the PCG boundary.

0x0080 DEM1X_TIME_INT2_MASK

Type: Read/Write
Clock: RXCHIPX8_CLK

Specifies the time of the SYS_TIME_INT2 interrupt from the combiner time.

Bits	Name	Description
10:7	5MS_MASK	This bitfield enables the interrupt strobe for the 5ms frame. Each bit of this register controls a 5ms frame strobe in the 20ms frame. ■ Set (1) a bit to enable the 5ms interrupt strobe. ■ Clear (0) a bit to disable the 5ms interrupt strobe.
6:5	OFFSET_PCG	This bitfield specifies the PCG offset of the interrupt strobe from the 5ms boundary.
4:0	OFFSET_SYMB	This bitfield specifies the 64-chip symbol offset of the interrupt strobe from the PCG boundary.

0x0084 DEM1X_TIME_INT_PHASE

Type: Read
Clock: RXCHIPX8_CLK

Returns the combiner time 5ms count in the offset 80ms frame at the time of the indicated interrupt.

Bits	Name	Description
7:4	TIME_INT2_PHASE	This bitfield returns the TIME_INT_PHASE value of SYS_TIME_INT2.
3:0	TIME_INT1_PHASE	This bitfield returns the TIME_INT_PHASE value of SYS_TIME_INT1.

0x0088+8n DEM1X_CHn_LC_MASK_LO_IMMED, n=[0,2]

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_SPN_ROLL register.

A write to this register immediately updates the lower 32 bits of the long code mask for channel n. A read of this register returns the lower 32 bits of the long code mask used for channel n.

Bits	Name	Description
31:0	DATA	Lower 32 bits of the long code mask for channel n. Reset state: X

0x008C+8n DEM1X_CHn_LC_MASK_HI_IMMED, n=[0,2]

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_SPN_ROLL register.

A write to this register immediately updates the upper 10 bits of the long code mask for channel n. A read of this register returns the upper 10 bits of the long code mask used for channel n.

Bits	Name	Description
9:0	DATA	Upper 10 bits of the long code mask for channel n. Reset state: X

0x00A0+8n DEM1X_CHn_LC_MASK_LO_SPN_ROLL, n=[0,2]

Type: Read/Write
Clock: RXCHIPX8_CLK

A write to this register stores the lower 32 bits of the long code mask for channel n. This value is copied to the active mask register (addressable by DEM1X_CHn_LC_MASK_LO_IMMED) on the following short PN boundary.

A read of this register returns the value last written to this register.

Bits	Name	Description
31:0	DATA	Lower 32 bits of the long code mask for channel n Reset state: X

0x00A4+8n DEM1X_CHn_LC_MASK_HI_SPN_ROLL, n=[0,2]

Type: Read/Write
Clock: RXCHIPX8_CLK

A write to this register stores the upper 10bits of the long code mask for channel n. This value is copied to the active mask register (addressable by DEM1X_CHn_LC_MASK_HI_IMMED) on the following short PN boundary.

A read of this register returns the value last written to this register.

Bits	Name	Description
9:0	DATA	Upper 10 bits of the long code mask for channel n Reset state: X

0x00B8 DEM1X_COMBINER_CTL

Type: Read/Write
Clock: RXCHIPX8_CLK
Reset this register to zero on combiner_reset to prevent combiner interrupts to MDSP.

DEM1X_COMBINER_CTL

Bits	Name	Description
11:8	FPC_DECISION_MASK	This bitfield is used to program the mask value, which is used to determine when to disable the fpc_decision1 and fpc_decision2 flags to the QDSP. A ‘1’ is used to disable these signals to the QDSP during an entire PCG. Program these 4 bits as given below depending on the gating rate. Program these 4 bits as given below depending on the gating rate. **Mask Value — Gating Rate** **0000 — 1** **0101 — 1/2** **0111 — 1/4**
7:4	PILOT_GATING_MASK	This bitfield is used to program the mask value, which is used to determine when to disable the pc_symbol and not_pc_decision flags to the QDSP. A ‘1’ is used to disable these signals to the QDSP during an entire PCG. Program these 4 bits as given below depending on the gating rate. **Mask Value — Gating Rate** **0000 — 1** **1010 — 1/2** **1101 — 1/4**
3	FPC_EN	This bit enables the power control for the fundamental channel and the high-rate supplemental channel.
2	FREQ_TRACK_EN	ft_enable to DSP

DEM1X_COMBINER_CTL (Continued)

1	Q_CHAN_OFF	When set (1), the demodulator will not send Q-channel soft decisions to the Viterbi decoder. ■ Set (1) this bit when receiving SYNC, paging, or RC 1 or 2 traffic channels. ■ Clear (0) this bit when receiving RC 3, 4, or 5 traffic channels, or a quick paging channel.
0	ENABLE	When set(1), combiner interrupts are sent to the mDSP. Reset state: 0

0x00BC DEM1X_FW_CH_ENABLE_IMMED

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Specifies enabled combiner channels to mDSP. For each channel enabled in this register it is assumed that at least one finger has the corresponding channel enabled.

The channel-enable takes effect at the next 64 PN chip boundary.

NOTE Bit(2) used by deinterleaver. Bits 2&3 cleared by DEM1X_COMBINER_CH1&2_CTL:HW_EN before sending to DSP.

Bits	Name	Description
7:0	ENABLE	When bit n is set (1), it specifies that channel n is to be enabled. 0 : Reset state

0x00C0 DEM1X_FW_CH_ENABLE_FRAME

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE Channels are enabled at the next 20 ms frame boundary.

Bits	Name	Description
7:0	ENABLE	When bit n is set (1), it specifies that channel n is to be enabled. 0 : Reset state

0x00C4 DEM1X_CHANNEL0_IMMED

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls the operation of channel 0 (using DEM1X_CHANNEL0_LC_MASK). All traffic channels carrying reverse link power control bits are assigned to channel 0. QPCH and CPCCH are assigned to channel 0.

Symbol combining for channel 0 is performed by mDSP firmware.

DEM1X_CHANNEL0_IMMED

Bits	Name	Description
28		Reserved.
27		Reserved.
26:18	OOK_POSITION	Specifies the start of QPCH bit in 64 PN chip units. The position is given with respect to the last combiner PN roll.
17	QPCH_EN	Quick paging channel processing enable. Set (1) to enable; clear (0) to disable.
16	CPCCH_EN	This bit enables the CPCCH. 1 : Enable 0 : Disable
15	RC4	Qualifies total pc_decision bit.
14	DCCH_ENABLE	Dedicated control channel enable. When set (1), channel 0 is enabled as DCCH. When clear (0), channel 0 is disabled as DCCH.
13:10		Reserved.
9:8	RATE_SEL	00 - symbol 64 chip 01 - symbol 128 chip 10 - symbol 256 chip 11 - symbol 512 chip
7:0	SW_BETA	The SW_BETA bitfield for is the fundamental channel frame based scaling factor for demod soft decision symbol to the convolutional decoder. This bitfield represents a signed 8-bit number and takes effect at next frame boundary. The following default values are listed in Table 13-3.

Table 13-3 SW_BETA_CH0 default values

Description	Value
RC 1, 2, 3, 5	0x20
RC 4	0x17
9.6 ksps QPCH	0x7F
4.8 ksps QPCH	0x40
CCI	0x20
Sync Channel	0x10

0x00C8 DEM1X_CHANNEL0_FRAME

Type: Read/Write
Clock: RXCHIPX8_CLK

This register holds a selection of channel 0 parameters that take effect on the following 20 ms frame boundary.

Bits	Name	Description
15	RC4	qualifies total PC bit - is it necessary (PC_EVEN / RATE_SEL...)?
14	DCCH_ENABLE	Dedicated control channel enable. When set (1), channel 0 is enabled as DCCH. When clear (0), channel 0 is disabled as DCCH.
13:10	TD_SIGN	0101 - RC3 5 0011 - RC4 0000 - others
9:8	RATE_SEL	00 - symbol 64 chip 01 - symbol128 chip 10 - symbol 256 chip 11 - symbol 512 chip
7:0	SW_BETA	The SW_BETA bitfield is the fundamental channel frame based scaling factor for demod soft decision symbol to the convolutional decoder. This bitfield represents a signed 8-bit number and takes effect at next frame boundary. The following default values are listed in Table 13-3.

0x00CC DEM1X_CHANNEL1_IMMED

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls channel 1 symbol combining, which is performed by either the mDSP firmware or hardware.

DEM1X_CHANNEL1_IMMED

Bits	Name	Description
28:24	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111WL_64 00111WL_32
23:19	RESERVED_BITS23_19	
18:17	SOFTDEC_ACCUM_RND	Rounding factor (number of truncated LSBs) applied to the accumulation of soft decision magnitudec (abs(I)+abs(Q) of QPSK symbol estimates prior to exporting through the COMB_CH1_IQ_ACC_RD DSP register. The desired rounding factor depends on the walsh symbol length and SCH frame length Walsh length Frame length desired rounding 128 80ms 1 128 20/40ms 0 64 any 0 32 any 1 Value Rounding 00 no rounding 01 round off 1 bit 10 round off 2 bits 11 round off 3 bits
16	SOFTDEC_RND	Mapped rounding factor (bit truncation) applied to the soft decisions on insertion to the deinterleaver. The desired value and shift are different for convolutional versus turbo codes. Value Rounding Encoding type 0 round off 10 bits conv 1 round off 8bits turbo
15	HW_EN	When set(1), symbol combining is performed in hardware. This is required for channels with Walsh length less that 64 PN chips. Channel 1 hardware supports a minimum Walsh length of 32 PN chips.
14	DCCH_ENABLE	Dedicated control channel enable. When set (1), channel 1 is enabled as DCCH. When clear (0), channel 1 is disabled as DCCH.

DEM1X_CHANNEL1_IMMED (Continued)

Bits	Name	Description
13:10	TD_SIGN	0101 - RC3 5 0011 - RC4 0000 - others
9:8	RATE_SEL	00 - symbol 64 chip 01 - symbol 128 chip 10 - symbol 256 chip 11 - symbol 512 chip
7:0	SW_BETA	The SW_BETA bitfield is the channel 1 frame based scaling factor for the demod soft decision symbol to the convolutional decoder. This value for this bitfield must be programmed based on whether the DSP or the MAC engine is processing channel 1. If the DSP is processing channel 1 (MAC_EN_CH1 = 0), values from Table 13-4 should be used. If the MAC is processing channel 1 (MAC_EN_CH1 = 1), values from Table 13-5 should be used.

Table 13-4 Programming values for DSP Channel 1 processing

Description	Value
QPCH 9.6 kbps	0x7f
QPCH 4.8 kbps	0x40
QPCH CCI	0x20
RC3, RC5	0x20
RC4	0x17

Table 13-5 Programming values for MAC Channel 1 processing

Description	Value
Convolutional 128 chips	0x5B
Convolutional 64 chips	0x80
Convolutional 32 chips	0x5B

0x00D0 DEM1X_CHANNEL1_FRAME

Type: Read/Write
Clock: RXCHIPX8_CLK

This register holds a selection of channel 1 parameters that take effect on the following 20 ms frame boundary.

DEM1X_CHANNEL1_FRAME

Bits	Name	Description
28:24	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111WL_128 01111WL_64 00111WL_32 00011WL_16 00001WL_8 00000WL_4
23:19	RESERVED_BITS23_19	
18:17	SOFTDEC_ACCUM_RND	Rounding factor (number of truncated LSBs) applied to the accumulation of soft decision magnitudec (abs(I)+abs(Q) of QPSK symbol estimates prior to exporting through the COMB_CH1_IQ_ACC_RD DSP register. The desired rounding factor depends on the walsh symbol length and SCH frame length Walsh length Frame length desired rounding 128 80ms 1 128 20/40ms 0 64 any 0 32 any 1 Value Rounding 00 no rounding 01 round off 1 bit 10 round off 2 bits 11 round off 3 bits
16	SOFTDEC_RND	Mapped rounding factor (bit truncation) applied to the soft decisions on insertion to the deinterleaver. The desired value and shift are different for convolutional versus turbo codes. Value Rounding Encoding type 0 round off 10 bits conv 1 round off 8bits turbo
15	HW_EN	When set(1), symbol combining is performed in hardware. This is required for channels with Walsh length less that 64 PN chips. Channel 1 hardware supports a minimum Walsh length of 32 PN chips.
14	DCCH_ENABLE	Dedicated control channel enable. When set (1), channel 1 is enabled as DCCH. When clear (0), channel 1 is disabled as DCCH.

DEM1X_CHANNEL1_FRAME (Continued)

13:10	TD_SIGN	0101 - RC3 5 0011 - RC4 0000 - others
9:8	RATE_SEL	00 - symbol 64 chip 01 - symbol 128 chip 10 - symbol 256 chip 11 - symbol 512 chip
7:0	SW_BETA	The SW_BETA bitfield is the channel 1 frame based scaling factor for the demod soft decision symbol to the convolutional decoder. This value for this bitfield must be programmed based on whether the DSP or the MAC engine is processing channel 1. If the DSP is processing channel 1 (MAC_EN_CH1 = 0), values from Table 13-4 should be used. If the MAC is processing channel 1 (MAC_EN_CH1 = 1), values from Table 13-5 should be used.

0x00D4 DEM1X_CHANNEL2_IMMED

Type: Read/Write
Clock: RXCHIPX8_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls channel 2 symbol combining, which is performed by either the mDSP firmware or hardware.

DEM1X_CHANNEL2_IMMED

Bits	Name	Description
30:29	REP_FACTOR	Specifies the number of Walsh function repetitions per modulation symbol. The valid values are: Value # of Frames 10 REP_4 01 REP_2 00 REP_1
28:24	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32 00011 WL_16 00001 WL_8 00000 WL_4
23:19	RESERVED_BITS23_19	

DEM1X_CHANNEL2_IMMED (Continued)

18:17	SOFTDEC_ACCUM_RND	Rounding factor (number of truncated LSBs) applied to the accumulation of soft decision magnitudec (abs(I)+abs(Q) of QPSK symbol estimates prior to exporting through the COMB_CH2_IQ_ACC_RD DSP register. The desired rounding factor depends on the walsh symbol length and SCH frame length Walsh length / Frame length / desired rounding 128 / 80ms / 1 128 / 20/40ms / 0 64 / any / 0 32 / any / 1 16 / any / 1 8 / any / 2 4 / any / 2 Value / Rounding 00 no rounding 01 round off 1 bit 10 round off 2 bits 11 round off 3 bits
16	SOFTDEC_RND	Mapped rounding factor (bit truncation) applied to the soft decisions on insertion to the deinterleaver. The desired value and shift are different for convolutional versus turbo codes. Value / Rounding / Encoding type 0 round off 10 bits conv 1 round off 8bits turbo
15	HW_EN	When set(1), symbol combining is performed in hardware. This is required for channels with Walsh length less that 64 PN chips. Channel 2 hardware supports a minimum Walsh length of 4 PN chips.
14:8	RESERVED_BITS14_8	
7:0	SW_BETA	The SW_BETA bitfield is the frame-based scaling factor for the supplemental channel. This scaling factor is provided by the software for demodulator soft decision symbols. This scaling factor is an unsigned 8 bit number and takes effect at next frame boundary. Since this value is always positive, the hardware stuffs a zero on the MSBit to make it a 9-bit signed number.

Table 13-6 SW_BETA_CH2 for Turbo codes, MFI factor = 1

Data rate	RC3	RC4	RC5
2x	0x48	0x90	0x58
4x	0x66	0x66	0x7C
8x	0x48	0x90	0x58

Table 13-6 SW_BETA_CH2 for Turbo codes, MFI factor = 1

Data rate	RC3	RC4	RC5
16x	0x66	0x66	0x7C
32x	--	0x90	--

Table 13-7 SW_BETA_CH2 for Turbo codes, MFI factor = 2

Data rate	RC3	RC4	RC5
2x	0x33	0x66	0x3E
4x	0x48	0x48	0x58
8x	0x33	0x66	0x3E
16x	0x48	0x48	0x58
32x	--	0x66	--

Table 13-8 SW_BETA_CH2 for Turbo codes, MFI factor = 4

Data rate	RC3	RC4	RC5
2x	0x24	0x48	0x2C
4x	0x33	0x33	0x3E
8x	0x24	0x48	0x2C
16x	0x33	0x33	0x3E
32x	--	0x48	--

0x00D8 DEM1X_CHANNEL2_FRAME

Type: Read/Write
Clock: RXCHIPX8_CLK

This register holds a selection of channel 0 parameters that take effect on the following 20 ms frame boundary.

DEM1X_CHANNEL2_FRAME

Bits	Name	Description
30:29	REP_FACTOR	Specifies the number of Walsh function repetitions per modulation symbol. The valid values are: Value # of Frames 10 REP_4 01 REP_2 00 REP_1

DEM1X_CHANNEL2_FRAME (Continued)

28:24	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32 00011 WL_16 00001 WL_8 00000 WL_4
23:19	RESERVED_BITS23_19	
18:17	SOFTDEC_ACCUM_RND	Rounding factor (number of truncated LSBs) applied to the accumulation of soft decision magnitudec (abs(I)+abs(Q) of QPSK symbol estimates prior to exporting through the COMB_CH2_IQ_ACC_RD DSP register. The desired rounding factor depends on the walsh symbol length and SCH frame length Walsh length Frame length desired rounding 128 80ms 1 128 20/40ms 0 64 any 0 32 any 1 16 any 1 8 any 2 4 any 2 Value Rounding 00 no rounding 01 round off 1 bit 10 round off 2 bits 11 round off 3 bits
16	SOFTDEC_RND	Mapped rounding factor (bit truncation) applied to the soft decisions on insertion to the deinterleaver. The desired value and shift are different for convolutional versus turbo codes. Value Rounding Encoding type 0 round off 10 bits conv 1 round off 8bits turbo
15	HW_EN	When set(1), symbol combining is performed in hardware. This is required for channels with Walsh length less that 64 PN chips. Channel 2 hardware supports a minimum Walsh length of 4 PN chips.
14:8	RESERVED_BITS14_8	
7:0	SW_BETA	The SW_BETA bitfield is the frame-based scaling factor for the supplemental channel. This scaling factor is provided by the software for demodulator soft decision symbols. This scaling factor is an unsigned 8 bit number and takes effect at next frame boundary. Since this value is always positive, the hardware stuffs a zero on the MSBit to make it a 9-bit signed number.

0x00DC **Reserved.**

0x00E0 **Reserved.**

0x00E4 **Reserved.**

0x00E8 **Reserved.**

0x00F0 **Reserved.**

0x00F4 **Reserved.**

13.1.2 OFFLINE clock registers

NOTE It is important for the correct functioning of the hardware that the physical addresses for the registers DEM1X_F0_PILOT_IMMED to DEM1X_F5_HW_CH2_IMMED are in one continuous range without any other registers or holes in between.

The same is true of physical addresses for the registers DEM1X_F0_WALSH_ACC0_FRAME to DEM1X_F5_HW_CH2_FRAME. Furthermore, the DEM1X_F0_WALSH_ACC0_IMMED and the DEM1X_F0_WALSH_ACC0_FRAME have to be 128 addresses (+ 4*128 = 0x200 offset apart).

0x00+0x38n DEM1X_Fn_PILOT_IMMED, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

Controls pilot processing for finger n.

DEM1X_Fn_PILOT_IMMED

Bits	Name	Description
11	AUX_PILOT_ON	Enables the auxiliary pilot in place of common 0: Common 1: Auxiliary

DEM1X_Fn_PILOT_IMMED (Continued)

10:9	PILOT_LEN	Specifies the duration in PN chips of early and late pilot accumulations. 00: 64 01: 128 10: 256 11: 512 Recommended settings: non-TD modes: 00 TD-modes:01 auxiliary pilot on: 10 or 11 depending on Walsh length
8:0	PILOT_PN_OFFSET	Specifies the pilot PN sequence offset in units of 64 chips (0 - 511).

0x04+0x38n DEM1X_Fn_DIVERSITY_IMMED, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

Controls diversity processing for finger n.

Bits	Name	Description
3	MMSE_EN	Set this bit to enable calculations for MMSE combining weights.
2	PRIMARY_ANT	For non receive diversity mode, this bit determines the antenna whose samples need to be demodulated. In receive diversity, this bit is ignored. Reset state: X
1:0	DIV_MODE	Specifies the diversity mode of the finger: 00: None 01: RD 10: TD_OTD 11: TD_STS Reset state: X

0x08+0x38n+4m DEM1X_Fn_WALSH_ACCm_IMMED, n=[0,5], m=[0,9]

Type: Read/Write
Clock: OFFLINE_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls Walsh accumulator m for finger n. Writes to this register take effect on the next processing cycle (within 64 PN chips of time). Reads of this register return the currently used settings for the finger. Updates may be pending for the next frame boundary.

Bits	Name	Description
11:3	WALSH_CODE	Walsh code channel W_n^{512} where n: 0 to 511.
2:1	QOF_SEL	QOF selection ranging from 0 to 3.
0	ENABLE	When set (1) accumulation is enabled

0x030+0x38n DEM1X_Fn_HW_CH1_IMMED, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls symbol processing for channel 1 of finger ‘n’

Bits	Name	Description
11:8	RESERVED_BITS11_8	
7:6	RND	Rounding factor used during the front end symbol combining
5:1	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32
0	ENABLE	When set, symbol processing for this channel is performed in hardware.

0x034+0x38n DEM1X_Fn_HW_CH2_IMMED, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

NOTE A write to this register automatically updates the corresponding *_FRAME register.

Controls symbol processing for channel 2 of finger ‘n’.

DEM1X_Fn_HW_CH2_IMMED

Bits	Name	Description
11:10	RESERVED_BITS11_10	
9:8	REP_FACTOR	Specifies the number of Walsh function repetitions per modulation symbol for the channel. The values are: Value # of repetitions 10 REP_4 01 REP_2 00 REP_1
7:6	RND	Rounding factor used during the front end symbol combining
5:1	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32 00011 WL_16 00001 WL_8 00000 WL_4
0	ENABLE	When set, symbol processing for this channel is performed in hardware.

0x208+0x38n +4m DEM1X_Fn_WALSH_ACCm_FRAME, n=[0,5], m=[0,9]

Type: Read/Write
Clock: OFFLINE_CLK.

Controls Walsh accumulator m for finger n. Writes to this register take effect on the next 20ms frame boundary. Reads of this register return the currently used settings for the finger.

Bits	Name	Description
11:3	WALSH_CODE	Walsh code channel W_n^{512} where n: 0 to 511.
2:1	QOF_SEL	QOF selection ranging from 0 to 3.
0	ENABLE	When set (1) accumulation is enabled

0x230+0x38n DEM1X_Fn_HW_CH1_FRAME, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

Controls symbol processing for channel 1 of finger ‘n.’

Bits	Name	Description
11:8	RESERVED_BITS11_8	
7:6	RND	Rounding factor used during the front end symbol combining
5:1	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32
0	ENABLE	When set, symbol processing for this channel is performed in hardware.

0x234+0x38n DEM1X_Fn_HW_CH2_FRAME, n=[0,5]

Type: Read/Write
Clock: OFFLINE_CLK

Controls symbol processing for channel 2 of finger ‘n.’

Bits	Name	Description
11:10	RESERVED_BITS11_10	
9:8	REP_FACTOR	Specifies the number of Walsh function repetitions per modulation symbol for the channel. The values are: Value # of repetitions 10 REP_4 01 REP_2 00 REP_1
7:6	RND	Rounding factor used during the front end symbol combining
5:1	SPR_MASK	Specifies the Walsh length. The supported values are: Value Walsh Length 11111 WL_128 01111 WL_64 00111 WL_32 00011 WL_16 00001 WL_8 00000 WL_4
0	ENABLE	When set (1), symbol processing for this channel is performed in hardware.

0x350 **Reserved.**

0x354 **Reserved.**

0x358 **Reserved.**

0x35C **Reserved.**

0x360 **Reserved.**

0x364 **Reserved.**

0x368 **Reserved.**

.

14 Deinterleaver

14.1 ARM registers

Addresses file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: DEINT
- Number of word addresses: 32
- Byte Address: 0x6900 - 0x69FC
- OFFSET=CHIP_BASE+0x6900
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

14.1.1 Decoder output buffer/turbo write registers

0x0000 DINT_RESET

Type: Write
Clock: DEINT_CLK

Bits	Name	Description
0	DATA	Deinterleaver reset register. Writing to this register causes a reset pulse for the entire deinterleaver.

0x0004 DINT_CFG

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The DINT_CFG register is the deinterleaver configuration register. This register programs the overall deinterleaver memory management parameters. The values programmed in this register affect the deinterleaver write control, and are therefore double-buffered. The programmed values take effect immediately.

DINT_CFG

Bits	Name	Description
15	L_UPPER_RAM_SEL	■ Clear (0) this bit to indicate that the start address for this RAM (RAM L) is 0 (with BCCH or CCCH or IS-95-C convolutional SCH). ■ Set (1) this bit to indicate that the start address for RAM L is 768 (in IS-95-C convolutional SCH).
14	L_ODD_RAM_PAGE_SEL	Clear (0) this bit for RAM L.
13	M_UPPER_RAM_SEL	■ Clear (0) this bit to indicate that the start address for this RAM (RAM M) is ‘0’ (with BCCH or CCCH or IS-95-C convolutional SCH). ■ Set (1) this bit to indicate that the start address for RAM M is 768 (in IS-95-C convolutional SCH).
12	M_ODD_RAM_PAGE_SEL	■ Clear (0) this bit for RAM M. ■ If a small (<= 3072 symbols per frame) frame turbo encoded channel (ch2) is allocated to RAMs L and M, set (1) this bit to indicate an odd page for the writing into the RAM.
11	N_UPPER_RAM_SEL	■ Clear (0) this bit to indicate that the start address for this RAM (RAM N) is 0 (with BCCH or CCCH or IS-95-C convolutional SCH). ■ Set (1) this bit to indicate the start address for RAM N is 6144 (in IS-95-C convolutional SCH).
10	N_ODD_RAM_PAGE_SEL	Set (1) this bit to indicate an even page for writing turbo encoded symbols (only for ch2: IS-95-C turbo SCH).
9	P_UPPER_RAM_SEL	■ Clear (0) this bit to indicate that the start address for this RAM (RAM P) is 0 (with BCCH or CCCH). ■ Set (1) this bit to indicate the start address for RAM P is 6144 (in IS-95-C convolutional SCH).

DINT_CFG (Continued)

Bits	Name	Description
8	P_ODD_RAM_PAGE_SEL	Set (1) this bit to indicate an odd page for writing turbo encoded symbols (only for ch2: IS-95-C turbo SCH).
7:0	CHAN_RAM_ALLOC	These bits allocate a specific channel to a specific deinterleaver RAM partition. 00: CH0 allocated 01: CH1 allocated 10: CH2 allocated [7:6]: Channel allocated to RAM L (LRAM_CHAN_ALLOC) [5:4]: Channel allocated to RAM M (MRAM_CHAN_ALLOC) [3:2]: Channel allocated to RAM N (NRAM_CHAN_ALLOC) [1:0]: Channel allocated to RAM P (PRAM_CHAN_ALLOC) RAMs L and M are smaller in size (can each hold up to 768 symbols per page). RAM N,P are the larger in size (can each hold up to 6144 symbols per page). If the simultaneous active channels have a size that can fit into L, or M, then these channels should be allocated to one of these RAMs (e.g., DCCH, FCH, SYNC, Paging, low-rate CCCH, and possibly low-rate SCH). For BCCH, CCCH, or IS-95-B (up to 7) SCH either RAMs N or P may be able to fit all the symbols in a frame. In the case of all turbo SCH and some high rate conv SCH, both RAM N and P must be used. It is recommended that the CH0 and CH1 be programmed for lower rates, and CH2 be reserved for SCH. size: 1 page RAM L RAM M RAM N RAM P RAM NP MSM6550/6150 768 768 6144 6144 12288

0x0008 +4w DINT_CHw_CFG, w=[0..2]

Type: Write
Clock: DEINT_CLK

The DINT_CH0_CFG, DINT_CH1_CFG, and DINT_CH2_CFG registers are the configuration registers for deinterleaver channels 0, 1, and 2. These channels may be simultaneously active and can only be convolutionally encoded.

The values in these registers affects the deinterleaver write (sequential) control, so they are double-buffered. Programmed values for bits 4:0 take effect on the next SYNC20 pulse.

DINT_CHw_CFG

Bits	Name	Description
7	SYNC26_EN	Set (1) to indicate there is a SYNC channel with a 26 ms boundary, and use sync26. Otherwise, use SYNC_chan_80m_bnd to distinguish whether it is a SYNC channel with 80 ms boundary or not.
6	SYNC_CHAN_80M_BND	■ If this bit is set (1), a SYNC channel is available. ■ If this bit is clear (0), a SYNC channel is not available. This bit takes effect immediately. ■ For SYNC channel, set (1) bit 6 and clear (0) bit 5. ■ For all other channels, clear (0) bit 6. Bits 6:5 take effect immediately (they do not wait for a SYNC20 boundary to take effect).
5	NEXT_20M_BND	This bit indicates whether the next SYNC20 signal is a frame boundary for this channel. Set (1) this bit 1: yes Clear (0) this bit: no This bit takes effect immediately.

DINT_CHw_CFG (Continued)

Bits	Name	Description
4:3	SEQUENCE_REP	These bits indicate whether there is a repetition of the sequence (frames or packets). 00: One sequence repetition 01: Two sequences repeated 10: Four sequence repetition 11: Undefined_Bit **Note:** Currently, sequence repetition is supported only with convolution codes (provided the symbol repetition is 4 or fewer), because a symbol from the previous frame (being read with symbol repetition) must be fetched and added to the current symbol.
2:0	NUM_SUB_CHANS	These bits indicate the number of supplemental channels in the IS-95-B mode (1 to 7, programmed as 000 to 110). These SCHs are referred to as sub-channels and are the soft decision symbols for each of the SCHs, because the soft decision symbols for each SCH appear sequentially (rather than in parallel) from the demodulator on the same physical data bus. Clear (000) these bits for normal one-frame operation in all other channels.

0x0014 **Reserved.**

0x0018 TD_INTLV_CFG_LO

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_CFG_LO register is the lower 15 bits of the deinterleaver channel interleaver configuration parameters. This register effects the deinterleaver read control to the turbo decoder. That is, software must program these values with the channel configuration parameters for a particular frame.

NOTE DO NOT write to this register with the intent of using the parameters for a convolutionally encoded CH2.

Bits	Name	Description
15:13	ROW_WIDTH	In the CODE_TYPE for this channel is turbo coded (1), set these deinterleaving parameters. These bits indicate m = $\log_2$M, where M is the number of rows. This value determines the bit-width for the row counter. 000: 3-bit wide row index (M = 8) 001: 4-bit wide row index (M = 16) 010: 5-bit wide row index (M = 32) 011: 6-bit wide row index (M = 64) 100: 7-bit wide row index (M = 128) 101: 8-bit wide row index (M = 256)
12:0	UNDEFINED_BITS12_0	

0x001C TD_INTLV_CFG_HI

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_CFG_HI register is the upper 15 bits of the deinterleaver channel interleaver configuration parameters. This register affects the deinterleaver read control to the turbo decoder. That is, software must program these values 20 ms after it programs the channel configuration parameters for a particular frame.

NOTE DO NOT write to this register with the intent of using the parameters for a convolutionally encoded CH2.

Bits	Name	Description
15:8	INTLV_COLS	If the CODE_TYPE for this channel is turbo coded (1), set the deinterleaving parameters here. These bits indicate the value J-1, where J is the number of columns in the Row-Major deinterleaver.
7:0	INTLV_ROWS	If the CODE_TYPE for this channel is turbo coded (1), set the deinterleaving parameters here. These bits indicate the value M -1 = 2^m – 1, where M is the number of rows of the Row-Major deinterleaver.

0x0020 TD_INTLV_SIZE_LO

Type: Write
Clock: DEINT_CLK

The TD_INTLV_SIZE_LO register programs parameters for the turbo code interleaver.

Bits	Name	Description
15	EARLY_TERMINATION_EN	This bit enables early termination checking for the TD. 1: enabled 0: disabled
14	UNDEFINED_BIT	
13	TD_HYP_X1_ON	This bit enables X1 hypothesis for turbo decoding during variable rate operation. This bit is necessary if the full rate (no symbol repetition) is to be deinterleaved. So, for fixed-rate decoding, this bit is always disabled.
12:0	TD_INTLV_LEN_X1	These bits provide the length of the turbo code interleaver to the turbo decoder for X1 operation.

0x0024 TD_INTLV_SIZE_HI

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_SIZE_HI register programs parameters for the turbo code interleaver

Bits	Name	Description
15:10	MAX_ITER_NUM	These bits indicate the maximum number of iterations to be run.
9:4	MIN_ITER_NUM	These bits indicate the minimum number of iterations to be run before checking for early termination conditions.
3:0	UNDEFINED_BITS3_0	

0x 0028 TD_PUNCT_LO

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_PUNCT_LO register describes the lower 15 bits of the puncturing pattern and length for the full-rate turbo SCH channel (to be present as CH2) packet. This register affects the deinterleaver read control to the turbo decoder. So, the software must program these values 20 ms after it programs the channel configuration parameters for a particular frame.

NOTE DO NOT write to this register with the intent of using the parameters for a convolutionally encoded CH2.

TD_PUNCT_LO

Bits	Name	Description
15:8	PUNCT_PATTERN	These bits indicate the 8 lower bits of the puncturing pattern (left justified). So, these bits are programmed only if the puncturing length is greater than 16. ■ Set (1) all of these bits to indicate no puncturing. ■ Clear (0) all of these bits to indicate an erasure.
7:3	PUNCT_LENGTH	These bits indicate the length of the puncturing pattern. Valid values are1 through 24, programmed as 0 through 23. When there is no puncturing, set these bits to 11111 (23).

TD_PUNCT_LO (Continued)

Bits	Name	Description
2	TD_DEC_INT_EN	This bit enables the turbo decoder interrupt after the current decode 1: enable 0: disable
1:0	CODE_RATE	These bits indicate the code rate for this channel. 00: 1/2 code rate 01: 1/3 code rate 10: 1/4 code rate 11: Undefined_bit This bitfield does not effect the channel deinterleaver. It is only information that is passed onto the turbo decoder for each packet.This bit is double buffered (for TD write and VD read).

0x002C TD_PUNCT_HI

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_PUNCT_HI register describes the upper 15 bits of the puncturing pattern and length for the full-rate turbo SCH channel (to be present as CH2) packet. This register affects the deinterleaver read control to the turbo decoder. So, the software must program these values 20 ms after it programs the channel configuration parameters for a particular frame.

NOTE DO NOT write to this register with the intent of using the parameters for a convolutionally encoded CH2.

Bits	Name	Description
15:0	PUNCT_PATTERN	■ These bits indicate the16-upper bits of the puncturing pattern (left justified). If the puncturing pattern is 16 or less in length, the entire pattern will fit into this register. Set (1) all of these bits to indicate no puncturing. ■ Clear (0) all of these bits to indicate an erasure. Write the pattern left-justified. For example, if the pattern is 110101, then set bits 31:14 with this, and set the length of pattern as 00101 (5) in PUNCT_LENGTH.

0x0030 TD_PARAMS_LO

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_PARAMS_LO register programs the lower 15 bits of different parameters that are required for the turbo encoded channel CH2. These values do not effect the channel deinterleaver. It is only information passed onto the turbo decoder for each packet. This register takes effect on the next SYNC20 pulse.

Bits	Name	Description
15:14	FRM_20M_RATE_HYP	Values depend on firmware release.
13:9	SLIDE_WIN_LEN	These bits indicate the sliding window length. While it is possible to program this bitfield from 1 to 32, only 24 and 32 are supported.
8:4	PARTIAL_WIN_LENGTH	These bits program the partial window length (fractional part of the number of windows). This value is less than 32.
3:0	BETA_SELECT	These bits program the beta scaling for the turbo codes (this is not done in the demod).

0x0034 TD_PARAMS_HI

Type: Write
Clock: DEINT_CLK

The TD_PARAMS_HI register programs the upper 15 bits of different parameters that are required for the turbo encoded channel CH2. These values do not effect the channel deinterleaver. It is only information passed onto the turbo decoder for each packet. This register takes effect on the next SYNC20 pulse.

Bits	Name	Description
15:10	MIN_NUM_CRC_PASS	These bits indicate the minimum number of consecutive CRC checks which must be passed for early termination. Program N - 1 to get N minimum checks.
9:8	UNDEFINED_BITS9_8	
7:0	NUM_WINDOWS	These bits program the integral part of the number of windows for the turbo decoder. These are programmed from the lookup table that is specified in the turbo decoder.

0x0038 DINT_PKT_OFFSET

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The DINT_PKT_OFFSET register programs the address pointer of the DINT_CFG_RAM to be at the appropriate position, in order to modify a particular packet configuration parameter. The processor write to this register loads a counter, which is the write address pointer for the CFG_RAM.

Bits	Name	Description
6:3	PACKET_OFFSET	These bits program the packet offset: 0000: pkt A (0001: pkt B (0010: pkt C (0011: pkt D (0100: pkt E (0101: pkt F (0110: pkt G (0111: pkt H (1000: pkt I (1001: pkt J (1010: pkt K (1011: pkt L (1100: pkt M (1101: pkt N (1110: pkt P (1111: pkt Q
2:0	PACKET_TYPE	These bits program which part of the packet configuration to program (each packet configuration consists of 128-bit values .000: INTLV_CFG_LO 001: INTLV_CFG_HI 010: BLK_SIZE_LO 011: BLK_SIZE_HI 100: PUNCT_LO 101: PUNCT_HI 110: PL_CRC_LO 111: PL_CRC_HI

0x003C Reserved.

0x0040 DINT_TASK_OFFSET

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The DINT_TASK_OFFSET register programs the address pointer of the deinterleaver task scheduler RAM to be at the appropriate task.

0x0044 DINT_TASK_LIST

Type: Write
Clock: DEINT_CLK

This register loads all of the tasks to be done by the deinterleaver.

0x0048 Reserved.

0x004C Reserved.

0x0050 Reserved.

0x0054 Reserved.

0x0058 DEC_OB_ADDR

Type: Write
Clock: MP_DEC_OB_RAM_ADDR

Bits	Name	Description
9:0	DATA	The DEC_OB_ADDR register is the offset register for the decoder output buffer address.

0x005C Reserved.

0x0060 DINT_OTD_CFG

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The DINT_OTD_CFG register is the deinterleaver configuration register for the OTD mode.

Bits	Name	Description
1	OTD_SEL	When this bit is set (1), the entire deinterleaver is in a two-bank mode of deinterleaving (IS-95-C). When this bit is cleared (0), IS-95-A,B one bank interleaving is used.
0	CH2_CODE_TYPE	This bit indicates the code type for the channel CH2. 0: convolutionally encoded 1: turbo encoded

0x0064 TD_MIN_LLR_THRESH

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_MIN_LLR_THRESH register is the minimum log likelihood ratio (LLR) threshold for the turbo decoder. The LLR is used for the early termination check. If the register value is 0x0, the minimum LLR check is effectively disabled.

Bits	Name	Description
11:0	MIN_LLR_THRESH	Minimum log-likelihood ratio.

0x0068 TD_INTLV_LEN_X2

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_LEN_X2 register programs the parameters into the turbo code interleaver for X2 operation.

Bits	Name	Description
13	TD_HYP_X2_ON	Setting (1) this bit enables X2 hypothesis for turbo decoding during variable rate operation. This bit is cleared (0) on RESOUT_N
12:0	TD_INTLV_LEN_X2	These bits provided the length of the turbo code interleaver to the turbo decoder for X2 operation

0x006C TD_INTLV_LEN_X4

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_LEN_X4 register programs the parameters into the turbo code interleaver for X4 operation.

Bits	Name	Description
13	TD_HYP_X4_ON	Setting (1) this bit enables X4 hypothesis for turbo decoding during variable rate operation. This bit is cleared (0) on RESOUT_N
12:0	TD_INTLV_LEN_X4	These bits provided the length of the turbo code interleaver to the turbo decoder for X4 operation

0x0070 TD_INTLV_LEN_X8

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

The TD_INTLV_LEN_X8 register programs the parameters into the turbo code interleaver for X8 operation.

Bits	Name	Description
13	TD_HYP_X8_ON	Setting (1) this bit enables X8 hypothesis for turbo decoding during variable rate operation. This bit is cleared (0) on RESOUT_N
12:0	TD_INTLV_LEN_X8	These bits provided the length of the turbo code interleaver to the turbo decoder for X8 operation

0x0074 TD_NUM_SLWIN_X2

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

Bits	Name	Description
7:0	DATA	The TD_NUM_SLWIN_X2 register programs the number of sliding windows for the turbo decoder during variable rate operation with X2 symbol repetition. This register takes effect on the next SYNC20 pulse. These bits are to be programmed from the lookup table that is specified in the turbo decoder.

0x0078 TD_NUM_SLWIN_X4

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

Bits	Name	Description
7:0	DATA	The TD_NUM_SLWIN_X4 register programs the number of sliding windows for the turbo decoder during variable rate operation with X4 symbol repetition. This register takes effect on the next SYNC20 pulse. These bits are to be programmed from the lookup table that is specified in the turbo decoder.

0x007C TD_NUM_SLWIN_X8

Type: Write
Clock: DEINT_CLK
Reset State: 0x0

Bits	Name	Description
7:0	DATA	The TD_NUM_SLWIN_X8 register programs the number of sliding windows for the turbo decoder during variable rate operation with X8 symbol repetition. This register takes effect on the next SYNC20 pulse. These bits are to be programmed from the lookup table that is specified in the turbo decoder.

14.1.2 Deinterleaver/decoder output buffer/turbo read registers

0x0000 DEINT_TASK_STATUS

Type: Read
Clock: DEINT_CLK

The DINT_TASK_STATUS register indicates the last task that was completely decoded by the Viterbi decoder. This information helps in reading some data signals or the state of the deinterleaver. It also provides observability at the top level, mainly for debugging purposes.

Bits	Name	Description
13:6	UNDEFINED_BITS13_6	
5:0	TASK_ID	The VD_DEC_DONE interrupt captures into this bitfield the task that was most recently finished by the Viterbi decoder.

0x0004 Reserved.

0x0008 VD_DONE_STATUS

Type: Read
Clock: DEINT_CLK

Bits	Name	Description
0	DATA	The VD_DONE_STATUS register indicates the status of the Viterbi decoder. This register is set every time the decoder finishes decoding a task. In case of IS-95B SCCH, this bit is set only when all of the active SCH decoding has been finished. 1: task was done 0: task not yet completed This register is reset whenever the micro-processor reads from this register or the deinterleaver is reset (a write to the DINT_RESET register occurs). This bit is NOT reset on a SYNC20 pulse.

0x000C TD_DONE_STATUS

Type: Read
Clock: DEINT_CLK

Bits	Name	Description
0	DATA	The TD_DONE_STATUS register indicates the status of the turbo decoder. This register is set (1) if the decoder is finished decoding. In the case of variable rate decoding, this bit is set (1) if all of the variable TD rate hypotheses are completed. 1: task was done 0: task not yet completed This register is reset whenever the micro-processor reads from this register or the deinterleaver is reset (a write to the DINT_RESET register occurs). This bit is not reset on a SYNC20 pulse.

0x0010 DEC_OB_DATA

Type: Read
Clock: RD_DEC_OB_RAM_DATA

Bits	Name	Description
15:0	DATA	The DEC_OB_DATA register returns the decoded bits that were prefetched from the output buffer.

15 SVD

15.1 ARM registers

The Viterbi decoder core processes the soft-decision symbols received from the deinterleaver block. It decodes the symbols to generate the convolutional encoder input bit stream.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: SVD
- Number of word addresses: 64
- Byte Address: 0x6B00 - 0x6BFC
- OFFSET=CHIP_BASE+0x6B00
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x0000 VD_RESET

Type: Write
Clock: SVD_CLK

The VD_RESET register is the reset register for the serial Viterbi decoder.

Bits	Name	Description
0	RESET	Generates an internal reset signal for the Jaguar serial Viterbi decoding unit. After the signal is generated, the decoding unit is ready to be initialized by the microprocessor.

0x0004 VD_MODE

Type: Write
Clock: SVD_CL

The VD_MODE register contains the control bits that determine the operating mode of the Viterbi decoder. Writing to this register during an active decode may invalidate the current frame.

Bits	Name	Description
4	FLUSH	When this bit is set (1), the decoder is forced to flush after the current packet even if it is in continuous mode. This mode is used for fast sync channel acquisition.
3:1	INITMODE[2:0]	Set this field to initialize the ACS states at the beginning of each frame. 000 - no init of states 001 - init state 0. Saturate all other state metrics 010 - init state 0 to 15. Saturate all other state metrics 011 - init states 0 to 31. Saturate all other state metrics 100 - ini all states
0	PACKET	This bit specifies that the incoming data is for the traffic channel. This causes each packet to be treated as an entity to be decoded in its entirety. The chainback is flushed at the end of each packet. Setting (1) then clearing (0) this bit between continuous mode frames may reset the internal decoder states.

0x0008 VD_POLY2IJ

Type: Write
Clock: SVD_CLK

The VDPOLY2IJ register specifies the polynomial to use for the Read = 1/2 i,j symbols. The value that is written does not contain the MSB or LSB, which are assumed to be 1.

Bits	Name	Description
13:7	POLY2J[6:0]	This bitfield is the polynomial for j.
6:0	POLY2I[6:0]	This bitfield is the polynomial for i.

0x000C VD_POLY3IJ

Type: Write
Clock: SVD_CLK

The VDPOLY3IJ register specifies the polynomial to use for the Read = 1/3 i,j symbols. The value that is written does not contain the MSB or LSB, which are assumed to be 1.

Bits	Name	Description
13:7	POLY3J[6:0]	This bitfield is the polynomial for j.
6:0	POLY3I[6:0]	This bitfield is the polynomial for i.

0x0010 VD_POLY3K

Type: Write
Clock: SVD_CLK

The VDPOLY3K register specifies the polynomial to use for the Read = 1/3 k,l symbols. The value that is written does not contain the MSB or LSB, which are assumed to be 1.

Bits	Name	Description
6:0	POLY3K[6:0]	This bitfield is the polynomial for k.

0x0014 VD_POLY4IJ

Type: Write
Clock: SVD_CLK

The VDPOLY4IJ register specifies the polynomial to use for the Read = 1/4 i,j symbols. The value that is written does not contain the MSB or LSB, which are assumed to be 1.

Bits	Name	Description
13:7	POLY4J[6:0]	This bitfield is the polynomial for j.
6:0	POLY4I[6:0]	This bitfield is the polynomial for i.

0x0018 VD_POLY4KL

Type: Write
Clock: SVD_CLK

The VDPOLY4KL register specifies the polynomial to use for the Read = 1/4 k,l symbols. The value that is written does not contain the MSB or LSB, which are assumed to be 1.

Bits	Name	Description
13:7	POLY4L[6:0]	This bitfield is the polynomial for l.
6:0	POLY4K[6:0]	This bitfield is the polynomial for k.

16 HDR Decoder Symbol Buffer / Output Buffer

16.1 Overview

The HDR decoder symbol buffer contains configuration information for all possible HDR turbo coded packets in its Frame Type Table (FTT). The FTT is programmed by the microprocessor through registers listed in the following sections.

The HDR output buffer has a number of software configurable modes as well as programmable interrupt conditions. See Section 16.3 of this chapter for application notes.

16.2 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: HDEC_PERPH
- Number of word addresses: 64
- Byte Address: 0x6A00 - 0x6AFC
- Offset = CHIP_BASE+0x6A00
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x0000 **Reserved.**

0x0004 **Reserved.**

0x0008 **Reserved.**

0x000C **Reserved.**

16.2.1 Output buffer

0x0010 HDR_DEC_OB_RESET

Type: Write (Command)
Clock: MSM_CLK

Bits	Name	Description
0	DEC_OB_RESET	When this register is written, the output buffer's reset signal is strobed high. This register affects both the decoder output buffer logic operating in the decoder clock regime as well as the DBIF logic operating on the ARM clock regime. To allow both resets to propagate correctly, the turbo decoder must be idle when this register is written. In addition, the ARM must wait for two cycles after this register is written to ensure the reset has fully propagated through both blocks.

0x0020 HDR_DEC_OB_HUNT_CHAR

Type: Read/Write (Control)
Clock: MSM_CLK

Bits	Name	Description
7:0	HUNT_CHAR[7:0]	This field is the 8 bits of the hunt character. The output buffer block searches for this character in each packet and, if the character is found, the corresponding HCB bit is set in the header. Reset State: X

0x0024 HDR_DBIF_WATERMARK

Type: Read/Write (Control)
Clock: MSM_CLK

Bits	Name	Description
3:0	WATERMARK[3:0]	The WATERMARK is the four bits of the watermark pointer. A value from 0 to 15 can be set. Each increment is a 1/16 of the DBIF RAM. A value of zero disables the generation of the watermark interrupt. Reset State: X

0x0028 HDR_DBIF_PACKET_CNT

Type: Read/Write (Control)
Clock: MSM_CLK

Bits	Name	Description
5:0	PACKET_CNT[5:0]	The DBIF_PACKET_CNT register is the six-bit count of the number of valid packets in the output buffer. A read of this register determines how many valid packets are in the DBIF buffer. This register is written at the start of a DBIF read cycle. Reset State: X

0x002C HDR_DBIF_STALE_TIMEOUT

Type: Read/Write (Control)
Clock: MSM_CLK

Bits	Name	Description
3:0	STALE_TIMEOUT[3:0]	The DBIF_STALE_TIMEOUT register is the four-bit count of the maximum number of slot rolls, with at least one valid packet in the buffer between interrupts. A value of zero in this register produces an interrupt after each packet. Reset State: X

0x0030 Reserved.

0x0034 Reserved.

0x0038 Reserved.

0x003C Reserved.

0x0040 Reserved.

0x0044 Reserved.

0x0048 **Reserved.**

16.2.2 Output buffer frame header

The decoder output buffer constructs and inserts a frame header at the beginning of each packet in the output buffer. The frame header consists of three 32-bit words which are shown below. The following three tables show the mapping of each 32-bit frame header word.

0x0000 **HDR_FRAME_HEADER_0**

Type: Read (Status)
Clock: DEC_CLK

Bits	Name	Description
31:28	FRAME_ID[3:0]	The Frame ID associated with the respective packet
27:24	FRAME_TYPE[3:0]	The Frame Type value assigned to the respective by the Demod DSP.
23	CRC	CRC Pass flag. When set, the respective packet passed the CRC check.
22:12	MIN_LLR[10:0]	Minimum Log-Likelihood Ratio from the Turbo Decoder from the decoding of the respective packet.
11	HCB0	Hunt Character: First 128-bytes When set, the hunt character was found in the first 128 bytes (bits 0-1023) of the respective packet.
10	HCB1	Hunt Character: Second 128-bytes When set, the hunt character was found in the second 128 bytes (bits 1024-2047) of the respective packet.
9	HCB2	Hunt Character: Third 128-bytes When set, the hunt character was found in the third 128 bytes (bits 2048-3071) of the respective packet.
8	HCB3	Hunt Character: Fourth 128-bytes When set (1), the hunt character was found in the fourth 128 bytes (bits 3072-4095) of the respective packet.
7:4	DATA_LENGTH[3:0]	The data length of the respective packet in 256 bit multiples. 0100: 1024 bits 1000: 2048 bits 1100: 3072 bits 0000: 4096 bits
3:0	ITERATIONS_RUN[3:0]	Number of turbo decoder iterations run.

0x0000 HDR_FRAME_HEADER_1

Type: Read (Status)
Clock: DEC_CLK

Bits	Name	Description
31:16	ENERGY_METRIC[15:0]	Energy metric from the turbo decoder for the respective packet.
15:12	FRAME_ID[3:0]	The Frame ID associated with the respective packet
11:8	FRAME_TYPE[3:0]	The Frame Type value assigned to the respective by the Demod DSP.
7:0	DSP_HEADER_BYTE0[7:0]	DSP Header Byte 0 The DSP appends a 5-byte frame header onto the packet prior to decoding. This header is passed through unaltered to the output buffer.

0x0000 HDR_FRAME_HEADER_2

Type: Read (Status)
Clock: DEC_CLK

Bits	Name	Description
31:24	DSP_HEADER_BYTE1[7:0]	DSP Header Byte 1 The DSP appends a 5-byte frame header onto the packet prior to decoding. This header is passed through unaltered to the output buffer.
23:16	DSP_HEADER_BYTE2[7:0]	DSP Header Byte 2 The DSP appends a 5-byte frame header onto the packet prior to decoding. This header is passed through unaltered to the output buffer.
15:8	DSP_HEADER_BYTE3[7:0]	DSP Header Byte 3 The DSP appends a 5-byte frame header onto the packet prior to decoding. This header is passed through unaltered to the output buffer.
7:0	DSP_HEADER_BYTE4[7:0]	DSP Header Byte 4 The DSP appends a 5-byte frame header onto the packet prior to decoding. This header is passed through unaltered to the output buffer.

16.3 Application notes

16.3.1 Symbol buffer

16.3.1.1 FTT table loading

The frame-type table (FTT) is a small RAM containing up to 16 different turbo decoder configurations, which are loaded into the FTT by the microprocessor. The 6-byte header that is attached to each HDR turbo decoder packet is used to select a turbo decoder configuration from the FTT. Table 16-1 shows the format of each configuration stored in the FTT. Each FTT entry consists of four 16-bit words. Hence, the FTT storage is a 48x16 RAM.

Table 16-1 Frame-type table format

Word	Name	Bits	Description
0	intlv_length	11:0	Turbo interleaver length
1	code_rate	14:12	Code rate 000 : 1/2 001 : 1/3 010 : 1/4 011 : 1/5 1xx : Reserved
	beta_sel	11:8	Beta select
	max_iter	7:4	Maximum number of iterations
	min_iter	3:0	Minimum number of iterations (early termination only)
2	early_termination_en	15	Early termination enable
	min_llr	14:4	Minimum LLR to satisfy early termination check
	consecutive_crc	3:0	Consecutive CRC passes to satisfy early termination check

The FTT uses an address/data port interface to the microprocessor for programming. The microprocessor initializes the HDR_FTTD_ADDR register to the desired starting location in the FTT RAM. The microprocessor can then either write or read to the FTT RAM location specified by HDR_FTTD_ADDR. Once the address pointer is initialized, the microprocessor can make an arbitrary number of accesses to consecutive address locations via the HDR_FTTD_DATA port without having to manipulate the address pointer. The HDR_FTTD_ADDR value automatically increments with each access to HDR_FTTD_DATA.

16.3.2 Decoder output buffer

16.3.2.1 Interrupt generation in the DBIF mode

In the DBIF mode, the processor is interrupted in either of two ways:

- The buffer is filled with X number of packets that exceeds the watermark. The watermark check is done after each packet is received.
- The buffer has at least one packet in it and the watermark is not exceeded, the number of slot (1.67 ms) rolls between interrupts exceeds the value in the DBIF_STALE_TIMEOUT register.

The granularity of the watermark is in increments of 1/16 of the RAM. Four bits are used in the DBIF_WATERMARK register to set the watermark. For example, a value of 0x4 in the DBIF_WATERMARK register generates an interrupt when the buffer is 4/16 full or when at least 393 words of the 1572x32 buffer are in use. A value of zero in the DBIF_WATERMARK register disables the watermark generation.

A value of zero in the DBIF_STALE_TIMEOUT register produces an interrupt after every packet. If the buffer does not have enough room for the next packet, the decoder is stalled until buffer space is freed.

16.3.2.2 Software procedure

In DBIF mode, a hardware requirement exists that when servicing the decoder output buffer, software must read out all of the available packets.

The following process is a summary of DBIF interrupt processing:

1. Interrupt is received
2. The software reads the HDR_DBIF_PACKET_CNT register and determines whether there are enough packets to warrant servicing the output buffer. (optional)
3. The software writes to the HDR_DBIF_PACKET_CNT register, and commits to emptying the output buffer.
4. The software reads the HDR_DBIF_PACKET_CNT register, which returns the number of packets that must be read. Subsequent reads from the HDR_DBIF_PACKET_CNT register return the number of packets in the buffer that have been received after the write to the HDR_DBIF_PACKET_CNT register from the previous step.
5. The software reads packets from the output buffer.

When an interrupt is generated in the DBIF, the software can read the HDR_DBIF_ PACKET_CNT register to determine the number of packets that are currently in the decoder output buffer. If the software determines that it wishes to read the packets out of the decoder output buffer, the software then writes to the HDR_DBIF_PACKET_CNT register. The data value associated with the write does not matter, because the HDR_DBIF_PACKET_CNT register acts as a command register with respect to writes.

The write to the HDR_DBIF_PACKET_CNT register signals the hardware that the software is going to service the decoder output buffer, and it commits the software to empty the output buffer. After writing to the HDR_DBIF_PACKET_CNT register, the software must again read the HDR_DBIF_PACKET_CNT register, in case another packet was written into the output buffer between the original read and the write of DBIF_PACKET_CNT. The software then proceeds to read out the number of packets that are indicated by the second read from the HDR_DBIF_ PACKET_CNT register.

Once an interrupt is generated by either the watermark threshold or the slot roll, no new interrupts are generated until the original interrupt is serviced. In the case of the slot roll counter, the counter is reset and frozen until the interrupt is serviced.

17 Modulator

17.1 Overview

This chapter contains the following register sections:

- 1X modulator software registers
- HDR modulator software registers
- Tx AGC software registers
- Modulator firmware registers
  - HDR modulator firmware registers
  - Tx AGC power control firmware registers

17.1.1 Shared registers of 1X and HDR

Following registers or register bits are used in both 1X and HDR mode.

- MOD_MODE
- MOD_CH2_ENC_CTL_0[15]: FCH_TX_TONE_EN
- MOD_MISC_CTL[13]: TX_BBA
- MOD_MISC_CTL[6]: TX_DATA_FORMAT
- TX_I_CLK
- MOD_CLK_CTL

17.2 ARM registers

There is only one micro_interface for the modulator, so all three sections (1X modulator, HDR modulator, and Tx AGC) share the same BASE offset.

- modltr_1x: offset = CHIP_BASE+0x6D00 MAX=CHIP_BASE+0x70FC
- modltr_hdr: offset= CHIP_BASE+0x6D00 MAX=CHIP_BASE+0x70FC
- TxAGC: offset=CHIP_BASE+0x6D00 MAX=CHIP_BASE+0x70FC

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: MODLTR
- Number of word addresses: 256
- Byte Address: 0x6D00 - 0x70FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

17.2.1 1X Modulator software registers

17.2.1.1 1X modulator functional software registers

0x0000 MOD_MODE

Type: Write/ read
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	0: 1X (modulator 1x mode) 1: HDR (modulator HDR mode)

0x0004 MOD_PRMBL_GAIN

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DATA	The MOD_PRMBL_GAIN register holds the preamble gain.

0x0008 MOD_RESET

Type: Write/Command
Clock: CDMA_TX_CLK

Bits	Name	Description
0	RESET	The MOD_RESET register is the reverse link subsystem reset register. A write to this register generates an internal reset signal for the reverse link subsystem. After a reset signal is generated, the modulator is ready to be initialized by the microprocessor. Data is not required for the write operation. The CHIPX8 clock must be oscillating for the reset to be recognized by the reverse link subsystem.

0x000C MOD_CH1_TIMING_CTL

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH1_TIMING_CTL register is the timing control register for modulator channel 1. Encoding begins at a fixed instant during the last PCG, which corresponds to the encode time of the largest frame (there are several frame sizes). By providing a programmable start time, the window available for the microprocessor to write data into the ERAMs is optimized.

Bits	Name	Description
2	UNUSED_BIT2	
1:0	CH1_BEGIN_ENC_TIME	This bit field specifies the time when channel 1 encoding should begin. This is one of 4 positions in a PCG.

0x0010 MOD_CH2_TIMING_CTL

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH2_TIMING_CTL register is the timing control register for modulator channel 2.

Bits	Name	Description
2	UNUSED_BIT2	
1:0	CH2_BEGIN_ENC_TIME	This bit field specifies the time when channel 2 encoding should begin. This is one of 4 positions in a PCG.

0x0014 MOD_CH3_TIMING_CTL

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH3_TIMING_CTL register is the timing control register for modulator channel 3.

Bits	Name	Description
2	CH3_PCG_CNT	This bit field specifies the PCG at which the encoding of channel 3 should begin. When rate 307.2 with turbo encoding is used, it is required to program (1) to this bit: 1 = Second to last PCG of frame 0 = Last PCG of frame (default) **Note:** If the CH3_TRAFFIC bit of the MOD_MISC_GTL register is set (1), last PCG of frame (default) of this bit is always selected.
1:0	CH3_BEGIN_ENC_TIME	This bit field specifies the time when channel 3 encoding should begin. This is one of 4 positions in a PCG.

0x0018 +4w Reserved.

0x0020 +4w Reserved.

0x0028 U_PN_STATE_0

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The U_PN_STATE_0 register is bits 7:0 of the long code PN initial state register for the reverse link. This register specifies the bits of the reverse link long code PN initial state for the modulator's long code PN generator. Bit i of this register is loaded into stage i of the reverse link long code PN generator at the next 80 ms frame boundary. The value of i ranges from 0 to 41. The time at which these values are loaded into the long code generator is specified by bits [3:2] of the MOD_STMR_CMD. Addresses 0x420, 0x424, 0x428, 0x42C, and 0x430 can be written to in any order, but address 0x434 must be written last.

0x002C U_PN_STATE_1

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The U_PN_STATE_1 register is bits 8:15 of the long code PN initial state register for the reverse link.

0x0030 U_PN_STATE_2

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The U_PN_STATE_2 register is bits 23:16 of the long code PN initial state register for the reverse link.

0x0034 U_PN_STATE_3

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The U_PN_STATE_3 register is bits 31:24 of the long code PN initial state register for the reverse link.

0x0038 U_PN_STATE_4

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The U_PN_STATE_4 register is bits 39:32 of the long code PN initial state register for the reverse link.

0x003C U_PN_STATE_5

Type: Write
Clock: CDMA_TX_CLK

The U_PN_STATE_5 register is bits 41:40 of the long code PN initial state register for the reverse link, as well as the BOZO control bit.

Bits	Name	Description
7	BOZO	This bit enables or disables the bozo mode operation of the reverse link long code PN generator. The bozo mode is a test mode in which the UPN state is loaded into the long PN generator every 80 ms. ■ Clear (0) this bit for non-bozo mode operation. ■ Set (1) this bit for bozo mode operation.
6:2	UNDEFINED_BITS6_2	
1:0	U_PN_STATE[41:40]	These bits specify bits 41 and 40 of the reverse link long code PN initial state for the long code PN generator of the modulator. After the data is written to this register, the reverse link long code PN initial state is loaded into the reverse link long code PN generator at the next internal 80 ms frame boundary of the modulator. The reverse link long code PN initial state is specified by these two bits and U_PN_STATE_0 through U_PN_STATE_5.

0x0040 +4w U_PN_MASK_w, w=[0..5]

Type: Write
Clock: CDMA_TX_CLK

This register specifies the long code PN mask for the reverse link long code PN generator of the modulator.

Bit i of this register corresponds to M_i of the long code PN mask, where i ranges from 0 to 41. For a definition of M_i, refer to IS-95-A.

The time at which these values are loaded into the long code generator is specified by bits [1:0] of the MOD_STMR_CMD.

For w=0,

Bits	Name	Description
7:0	DATA	The U_PN_MASK_0 register consists of bits 7:0 of the long code PN mask register for the reverse link.

For w=1,

Bits	Name	Description
7:0	DATA	The U_PN_MASK_1 register consists of bits 15:8 of the long code PN mask register for the reverse link.

For w=2

Bits	Name	Description
7:0	DATA	The U_PN_MASK_2 register consists of bits 23:16 of the long code PN mask register for the reverse link.

For w=3,

Bits	Name	Description
7:0	DATA	The U_PN_MASK_3 register consists of bits 31:24 of the long code PN mask register for the reverse link.

For w=4,

Bits	Name	Description
7:0	DATA	The U_PN_MASK_4 register consists of bits 39:32 of the long code PN mask register for the reverse link.

For w=5,

Bits	Name	Description
1:0	DATA	The U_PN_MASK_5 register consists of bits 41:40 of the long code PN mask register for the reverse link.

0x0058 FRAME_OFF

Type: Write
Clock: CDMA_TX_CLK

b

The FRAME_OFF register is the frame offset adjustment register.

Bits	Name	Description
9:4	UNUSED_BIT6_2	
3:0	REV_OFF_20M	These bits specify the number of power control groups that the reverse link subsystem frame boundary must be delayed from the zero-offset reverse link subsystem boundary. Advancement of the reverse link subsystem frame boundary from the combiner time reference is specified as show below: **12288 x (SYSFR_STATE – FRAME_OFF[3:0]) + TX_SYNC_ST CHIPX8 clock periods** These bits take effect at the next SYNC80M signal. IS-95-A only supports frame offsets for the forward and reverse traffic channel. The REV_OFF signal is applied to all reverse link channels. Clear REV_OFF for access channel processing.

0x005C PA_WARMUP

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The PA_WARMUP register is the power amplifier warmup period register. This register controls the time defined as WARMUP_PERIOD. When PA_PUNCT goes high prior to CDMA data transmission, the WARMUP_PERIOD is the time interval required to switch on the power amplifier and have the bias of the power amplifier reach a steady state. The value that must be written to this register is defined in ms using Equation 1: **PA_WARMUP = 153.6 X WARMUP_PERIOD + 64.** **(Equation 1)** ■ The range of values allowed for PA_WARMUP is from 65 to 162, which corresponds to a WARMUP_PERIOD range of 0.007 ms to 0.638 ms. Typical values are less than or equal to 162 (0.658 ms). ■ WARMUP_PERIOD is determined using the NV item, NV_SUBPCG_PA_WARMUP_DELAY, as follows: ■ WARMUP_PERIOD (in ms) = NV_SUBPCG_PA_WARMUP_DELAY (in chips) / 1228.8. ■ Valid values for NV_SUBPCG_PA_WARMUP_DELAY are in [8, 784] range, default value being 784. This register controls the PA_PUNCT output when the PA_CTL bit of the MISC_CTL register and the MASK_CTL bit of the TEST_CTL register are defined for the normal function (not set to define PA_PUNCT to a fixed state). **Note:** Updates to the MOD_FCH_CTL register should be completed after the WARMUP_PERIOD, before the beginning of the next frame. If this is not done, the PA_PUNCT signal, which controls the PA_ON output pin of the MSM, which turns on and off the power amplifier, will not function correctly.

0x0060 MOD_STMR_MODIFIER_0

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_STMR_MODIFIER_0 register contains the upper 16 bits of a positive integer that represents the CDMA 80 ms system time in eighths of a chip. System time is represented as a simple binary number with the following permissible range: CDMA 1x: 0 to 786,431 ($3 * 2^{15} * 8 - 1$) The STMR can be commanded to load the value of this field into the CDMA Tx system time counter. The MOD_STRM_MODIFIER_0/1 registers replace the TX_SYNC_ST0/1 and SYSFR_STATE registers. This value is calculated as shown in Equation 2: **12288 X SYSFR_STATE + TX_SYNC_ST + 2** **(Equation 2)**

0x0064 MOD_STMR_MODIFIER_1

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
4:0	DATA	The MOD_STMR_MODIFIER_1 register contains the upper 5 bits of the CDMA 80 ms system time. Because the STMR_MODIFIER value is from 0 to 786,431, the MSB is not used and should be ‘0’ all the time.

0x0068 MOD_STMR_CMD

Type: Write/Command
Clock: CDMA_TX_CLK

The MOD_STMR_CMD register is used to make the loading of LC state, mask, and Tx system time flexible.

Bits	Name	Description
5	LC_LOAD_DIS	If this bit is set (1), the modifier value is not loaded into the system time counter at the next RX 80 ms boundary. **Note:** This bit is a control bit, not a command bit like the other bits in this register. **Note:** The reset state of this bit does not allow loading of system time at Rx80ms therefore write a ‘0’ to this bit while setting up traffic.
4	OFFSET_LD_IMMD	■ If this bit is set (1), the frame offset takes effect immediately. ■ If this bit is cleared (0), the offset takes effect on the 20 ms boundary (offset by the present frame offset)
3:2	TX_LC_STATE_LD	This bit field loads the Tx long code generator with the contents of the upper and lower LC state load registers at the specified action time. This is a one-shot command. If there is a pending load operation (commanded but not yet executed) when a new load is commanded, the old load is cancelled and replaced by the new command. 00: No operation 01: Load (on the next non-offset Tx 80 ms boundary) These encodings are undefined:10 and 11
1:0	TX_MASK_LD	This bit field loads the Tx long code mask working register with the contents of the upper and lower Tx LC mask registers at the specified action time. This is a one-shot command. If there is a pending load operation (commanded but not yet executed) when a new load is commanded, the old load is cancelled and replaced by the new command. 00: No operation 01: Load immediately 10: Load (on the next offset Tx 20 ms boundary) This encoding is undefined: 11

0x006C ENC_INT_ST

Type: Write
Clock: CDMA_TX_CLK

The ENC_INT_ST register is the encoder interrupt timing adjustment register. This register specifies the time offset between the internal TX_FR_INT active signal and the beginning of the encoding of the next frame of reverse link data. The TX_FR_INT signal is advanced by an amount equal to the sum of the value specified by this register and one PN chip.

Bits	Name	Description
8:5	5ms_MASK	Set (1) corresponding bit to enable for the interrupt in the four 5ms bins within a 20ms duration. Bit 5 corresponds to Bin0, bit 6 Bin1, bit 7 Bin2 and bit 8 Bin3.
4:3	ENC_INT_PCG	These bits specify the time offset in units of the number of power control groups within a 5ms duration. ENC_INT_PCG is always positive and the maximum value is 3.
2:0	ENC_INT_WYSM	These bits specify the time offset in units of the number of Walsh symbols (64 Walsh Chips.) ENC_INT_WYSM is always positive and the maximum value is 5.

0x0070 MOD_CH1_ENC_CTL

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH1_ENC_CTL register is the DC channel encoder control for modulator channel 1.

MOD_CH1_ENC_CTL

Bits	Name	Description
13:9	CH1_CRC_LEN	This bit field is the CRC length of the DC channel. These bits are binary representations of the associated CRC polynomial length. Valid polynomial lengths range from 0 to 12. To disable CRC, write “00000” to these bits. Used for cdma2000 DCCH.
8:7	CH1_CODE_RATE	This bit field specifies the code rate of the DC channel. 00: 1/2 01: 1/3 10: 1/4 This encoding is undefined: 11

MOD_CH1_ENC_CTL (Continued)

Bits	Name	Description
6:4	CH1_RADIO_CFG	This bit field configures the DC channel radio. 011: RC3 (cdma2000 1x) 100: RC4 (cdma20001x) 101: RC5 (cdma2000 3x) 110: RC6 (cdma2000 3x) These encodings are undefined: 000,001,010,111.
3	CH1_ENC_RATE	This bit is the encoder rate of the DC channel. 0: 9.6 1: 14.4
2:1	CH1_FRAME_SIZE	This bit field is the frame size for the DC channel. 00: 5ms 01: 10ms 10: 20ms This encoding is undefined: 11
0	CH1_EN	This bit enables the DC channel. ■ Set (1): enabled ■ Clear (0): disabled In IS-95-A, this bit should be cleared (0).

0x0074 MOD_CH1_ENC_DATA

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_CH1_ENC_DATA register is the encoder data register for the dedicated control channel in the modulator. This register supplies data to the convolutional encoder for the reverse link dedicated control channel. The data is provided in either 16-bit or 8-bit chunks to the encoder buffer through this register. The order in which the bits are processed is [7][6][5]...[0][15][14][13]...[8]. In case the data is provided in 8 bits, the last 2 bits of the full address are used to determine which byte of the register are valid and these bits are stored. The number of information bits that need to be written depends on the data rate selected by the CH1_ENC_RATE bit of the MOD_CH1_ENC_CTL register. This register should only be written when the CH1_ENC bit of the MOD_STATUS register is clear (0). The fault status is indicated by the CH1_ENC_WR_ERR bit of the MOD_STATUS register.

0x0078 MOD_CH1_CRC_POLY

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_CH1_CRC_POLY register is the CRC polynomial for the SCH. Polynomials of less than 16 bits have the LSBits zero filled so that they form a 16-bit expression polynomial. This register must be re-written with each rate change. The following shows the register values for the various polynomial lengths: **Register value** — **CRC length** 0xC867 — 16 bit 0xF130 — 12 bit 0xF640 — 10 bit 0x9B00 — 8 bit 0x1C00 — 6 bit RC2) 0x9C00 — 6 bit RC3 to RC6

0x007C MOD_CH1_PUNCT_PATN_1

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH1_PUNCT_PATN_1 register is bits 23:16 of the dedicated control channel puncture pattern register. This register, along with the MOD_CH1_PUNCT_PATN_0 register, specifies the pattern used in the puncturing for modulator channel 1.

Bits	Name	Description
15:8	UNDEFINED_BITS15_8	
7:0	DCCH_PUNCT_PATTERN_23_16	■ For no puncturing (also for IS-95A), this value is 0xFFFFFF (when combined with the MOD_CH1_PUNCT_PATN_0 register). ■ For 8/24 puncturing, this value is 0XEBBAEA (when combined with the MOD_CH1_PUNCT_PATN_0 register).

0x0080 MOD_CH1_PUNCT_PATN_0

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH1_PUNCT_PATN_0 register is bits 15:0 of the puncture pattern for modulator channel 1.

Bits	Name	Description
15:0	DCCH_PUNCT_PATTERN_15_0	See the description for the DCCH_PUNCT_PATN_23_16 bit of the MOD_CH1_PUNCT_PATN_1 register.

0x0084 MOD_CH2_ENC_CTL_0

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH2_ENC_CTL_0 register is the encoder control register for the modulator FCH. In IS-95A, this register is used to control the traffic channel. In cdma2000 (IS-2000), it is used to control the fundamental channel.

MOD_CH2_ENC_CTL_0

Bits	Name	Description
15	FCH_TX_TONE_EN	■ Set (1) this bit to enable and select the internal TX test tone generator. ■ Clear (0) this bit for normal operation. The test generator produces an output tone on the TX IQ data pins with a frequency of the chip rate divided by four (chip/4). When TX_TONE_EN is set, the cdma2000_FCH_EN bit (bit 0 of MOD_FCH_CTL register) should also be set (1). To enable a TX tone, write 0x8001 into MOD_FCH_CTL register.
14:13	UNDEFINED_BITS14_13	
12:8	FCH_CRC_LENGTH	This bit field is the CRC length of the fundamental channel. These bits are binary representations of the associated CRC polynomial length. Valid polynomial lengths range from 0 to 16. To disable CRC, write “0000” to these bits. Used for IS-95-A and cdma2000 FCH. Note: One more bit is added to support a CRC length of 16 bits for FCH = 5ms.
7	UNDEFINED_BIT7	

MOD_CH2_ENC_CTL_0 (Continued)

Bits	Name	Description
6:5	FCH_CODE_RATE	This bitfield specifies the fundamental channel code rate 00: 1/2 01: 1/3 10: 1/4 This encoding is undefined: 11
4	UNUSED_BIT4	
3:2	FCH_ENC_RATE	These bits specify the data rate for processing of the next frame of reverse link data. 00: 1/8 01: 1/4 10: 1/2 11: full (9.6 kbps) Used for IS-95-A and cdma2000 FCH
1	IS_95_C	This bit is used to select between IS-95-A and IS-95-C. 0: IS-95-A 1: IS-95-C The IS_95C_FCH_EN bit should be set (1) when IS-95-A is cleared (0).
0	FCH_EN	■ Set (1) this bit to enable the fundamental channel. ■ Clear (0) this bit to disable the fundamental channel. In IS-95-A, this bit should be set (1).

0x0088 MOD_CH2_ENC_CTL_1

Type: Write
Clock: CDMA_TX_CLK
The MOD_CH2_ENC_CTL_1 register is the encoder control register for the FCH.

Bits	Name	Description
4:2	CH2_RADIO_CFG	This bitfield configures the FCH radio. 001: RC1 (IS95A) 010: RC 2 (IS95A) 011: RC3 (cdma2000 1x) 100: RC4 (cdma2000 1x) These encodings are undefined: 000,101,110, 111
1:0	CH2_FRAME_SIZE	This bitfield is the frame size for modulator channel 2. 00: 5ms 01: 10ms 10: 20ms This encoding is undefined: 11

0x008C MOD_CH2_ENC_DATA

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_CH2_ENC_DATA register is the fundamental channel encoder data register. This register supplies data for the reverse link fundamental channel convolutional encoder. The data is provided in either 16-bit or 8-bit chunks to the encoder buffer via this register. The order in which the bits are processed is [7][6][5]...[0][15][14][13]...[8]. In case the data is provided in 8 bits, the last 2 bits of the full address are used to determine which byte of the register are valid and these bits are stored. The number of information bits that need to be written depends on the data rate selected by the CH2_ENC_RATE bit of the MOD_CH2_ENC_CTL register. This register should only be written when the CH2_ENC bit of the MOD_STATUS register is clear (0). The fault status is indicated by the CH2_ENC_ERROR bit of the MOD_STATUS register.

0x0090 MOD_CH2_CRC_POLY

Type: Write
Clock: CDMA_TX_CLK
Bits: 15:0

Bits	Name	Description
15:0	DATA	The MOD_CH2_CRC_POLY register is the CRC polynomial for the FCH. Polynomials of less than 16 bits should have the LSBits zero filled so that they form a 16-bit expression polynomial. This register must be re-written with each rate change. The following lists the register values for the various polynomial lengths: **Register value** — **CRC length** 0xC867 — 16 bit 0xF130 — 12 bit 0xF640 — 10 bit 0x9B00 — 8 bit 0x1C00 — 6 bit RC2 0x9C00 — 6 bit RC3 to RC6

0x0094 MOD_CH2_PUNCT_PATN_1

Type: Write
Clock: CDMA_TX_CLK
The MOD_CH2_PUNCT_PATN_1 register is bits 23:16 of the puncture pattern for the FCH.

Bits	Name	Description
15:8	UNDEFINED_BITS15_8	
7:0	FCH_PUNCPATTERN_23_16	The following specifies the register values for the various FCH puncture patterns. **Register value** — **Puncturing rate** 0xFFFFFF — No puncture (for IS95A) 0xF00000 — 1/5 0xFF0000 — 1/9 0xEBBAEA — 8/24

0x0098 MOD_CH2_PUNCT_PATN_0

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH2_PUNCT_PATN_0 register is bits 15:0 of the puncture pattern for the FCH.

Bits	Name	Description
15:0	FCH_PUNCPATTERN_15_0	See the description for the MOD_CH2_PUNCT_PATN_1 register.

0x009C MOD_CH3_ENC_CTL_0

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH3_ENC_CTL_0 register is an encoder control register for the SCH.

MOD_CH3_ENC_CTL_0

Bits	Name	Description
15	UNDEFINED_BIT15	
14	SCH_LTU_EN	■ When set (1), enables LTU based encoding. ■ When cleared (0) encoding contains no LTUs. LTU encoding parameters are specified in the MOD_SCH_LTU_CTL register.
13	TURBO_ENCODE	This bit selects between the convolutional encoder and the turbo encoder. 0: Convolutional encoder 1: Turbo encode
12:8	SCH_CRC_LENGTH	These bits specify the CRC length of the SCH. These bits are binary representations of the associated CRC polynomial length. Valid polynomial lengths range from 0 to 16. To disable the CRC, write “0000” to these bits.
7:6	SCH_CODE_RATE	These bits specify the SCH code rate. 00: 1/2 01: 1/3 10: 1/4

MOD_CH3_ENC_CTL_0 (Continued)

Bits	Name	Description
5:1	SCH_ENC_RATE	These bits specify the data rate for processing of the next frame of reverse link data. Note that when multiframe interleaving by a factor N (=1,2 or 4), the data rate to be programmed is given by (actual data rate x N). **SCH_ENC_RATE RC3** 00000 1.5 Kbps 00001 2.7 Kbps 00010 4.8 Kbps 00011 9.6 Kbps 00100 19.2 Kbps 00101 38.4 Kbps 00110 76.8 Kbps 00111 153.6 Kbps 01000 307.2 Kbps These encodings are undefined: 01001 to 01111 **SCH_ENC_RATE RC4** 10000 1.8 Kbps 10001 3.6 Kbps 10010 7.2 Kbps 10011 14.4 Kbps 10100 28.8 Kbps 10101 57.6 Kbps 10110 115.2 Kbps 10111 230.4 Kbps These encodings are undefined: 10111 to 11111
0	SCH_EN	This bit is used to enable or disable the supplemental channel. ■ When set (1) the supplemental channel is enabled. ■ When cleared (0), supplemental channel is disabled.

0x00A0 MOD_CH3_ENC_CTL_1

Type: Write
Clock: CDMA_TX_CLK
The MOD_CH3_ENC_CTL_1 register is an additional encoder control register for the SCH.

MOD_CH3_ENC_CTL_1

Bits	Name	Description
15:14	UNDEFINED_BITS15_14	
13:12	MULTIFRAME	00: N = 1 (No multiframe) 01: N = 2 10: N = 4 This encoding is undefined: 11

MOD_CH3_ENC_CTL_1 (Continued)

Bits	Name	Description
11:9	LOW_REPEAT	000: M = 1 001: M = 2 010: M = 4 011: M = 8 100: M = 16 101: M = 32 These encodings are undefined: 110-111
8	UNUSED_BIT8	
7:5	CH3_INTLV_SIZE	These bits specify the data rate for processing of the next frame of reverse link data. **Bits** **CH TYPE_ACTUAL SIZE** 000 EA/CCCH_384 001 SCH_768 010 SCH_1536 011 S/EA/CCCH_3072 100 SCH_6144 101 SCH_12288 These encodings are undefined: 110-111
4	UNUSED_BIT4	
3:2	CH3_SEL	These bits are used to enable or disable the supplemental channel. 00 disabled (No SCH or CCCH or EACH) 01 SCH 10 CCCH 11 EACH
1:0	CH3_FRAME_SIZE	This bit specifies the frame size of the SCH. 00: 5ms Only for CCCH, EACH 01: 10ms Only for CCCH, EACH 10: 20ms This encoding is undefined: 11

0x00A4 MOD_CH3_ENC_DATA

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_CH3_ENC_DATA register is the encoder data register for modulator channel 3. This register supplies data to the reverse link convolutional encoder for the SCH/EACH/CC channel. The data is provided in 16-bit or 8-bit chunks to the encoder buffer via this register. The order in which the bits are processed is [7][6][5]...[0][15][14][13]...[8]. In case the data is provided in 8-bits the last 2 bits of the full address are used to determine which byte of the register are valid and these bits are stored. The number of information bits that need to be written depends on the data rate selected by the CH3_ENC_RATE bit of the MOD_CH3_ENC_ CTL0 register. This register should only be written when the CH3_ENC bit of the MOD_STATUS register is clear (0). The fault status is indicated by the CH3_ENC_WR_ER bit of the MOD_STATUS register.

0x00A8 MOD_CH3_CRC_POLY

Type: Writer
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The MOD_CH3_CRC_POLY register is the CRC polynomial for modulator channel 3. Polynomials of less than 16 bits should have the LSBits zero filled so that they form a 16-bit expression polynomial. This register must be re-written with each rate change. The following lists the register values for the various polynomial lengths: **Register value** — **CRC length** 0xC867 — 16 bit 0xF130 — 12 bit 0xF640 — 10 bit 0x9B00 — 8 bit 0x9C00 — 6 bit

0x00AC MOD_CH3_PUNCT_PATN_1

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH3_PUNCT_PATN_1 register is bits 23:16 of the puncture pattern for modulator channel 3. This register specifies the puncturing pattern used for puncturing the supplemental channel.

Bits	Name	Description
15:8	UNDEFINED_BITS15_8	
7:0	SCH_PUNCPATTERN_23_16	The following table shows the SCH puncture patterns and corresponding register values: **Register Value** — **Puncturing rate** 0xFFFFFF — No puncture (also for IS95A 0xF00000 — 1/5 0xFF0000 — 1/9 0xEBBAEA — 8/24 0xD9B000 — 4/12 c (for conv enc) 0xDDA000 — 4/12 t (for turbo enc) 0xEFFF80 — 2/18 c (for conv enc) 0xFAFFC0 — 2/18 t (for turbo enc)

0x00B0 MOD_CH3_PUNCT_PATN_0

Type: Write
Clock: CDMA_TX_CLK

The MOD_CH3_PUNCT_PATN_0 register is bits 15:0 of the puncture pattern for modulator channel 3.

Bits	Name	Description
15:0	SCH_PUNCPATTERN_15_0	See the description for the MOD_CH3_PUNCT_PATN_1 register.

0x00B4 MOD_SCH_LTU_CTL

Type: Write
Clock: CDMA_TX_CLK
Bits: 11:0

The MOD_SCH_LTU_CTL register is the supplemental channel LTU control for the modulator. This register is used in convolutionally encoded SCH only. This register is effective only when SCH_LTU bit of MOD_SCH_CTL register is set (1).

Bits	Name	Description
11:0	SCH_LTU_SIZE	This bitfield specifies the number of information bytes per LTU block (including LTU CRC length) within a frame. The hardware assumes that LTU CRC length is the same as the frame CRC length.

0x00B8 MOD_MISC_CTL

Type: Write
Clock: CDMA_TX_CLK
The MOD_MISC_CTL register is the miscellaneous functions control register for the modulator. These settings take effect at the reverse link subsystem's next 5 ms frame boundary.
MOD_MISC_CTL

Bits	Name	Description
15	IS95A_MODE	This is a global bit to specify if operating in IS95A mode. ■ Set (1) this bit if operating in radio configurations 1 or 2. ■ Clear (0) this bit if operating in radio configurations 3 or 4.
14	TRAFFIC	If this bit is set (1), the ch3 encoding always start at the last PCG before frame boundary. If this bit is (0), the encoding time is determined by MOD_CH3_TIMING_CTL[2].
13		Reserved.
12	RETRANS_DIS	■ Set (1) this bit to enable preemption of 5ms frames during the transmission of 20 ms frame and to continue transmission of the 20ms frame if possible. ■ Clear (0) it otherwise.

MOD_MISC_CTL (Continued)

Bits	Name	Description
11	FCH_EIGHTH_GATE_EN	Set (1) this bit to enable eighth rate gating in THE FCH. When this bit is set (1), SCH_EN and DCCH_EN must be cleared (0), and PCH_EN and FCH_EN should be set (1). IS95C_TX_PATN must be programmed with the correct gating pattern. The latest this bit can be updated is PA_WARMUP time before the transmit frame boundary.
10:9	PCH_GATING_RATE	When transmitting only the pilot channel in the gating mode, the mobile station periodically gates off certain PCGs of the pilot channel as specified by the pilot channel gating rate. 00: 1 (Transmission in all PCGs) 01: 1/2 (Transmission in all odd PCGs) 10: 1/4 (Transmission in PCGs 3, 7, 11, 15)
8	CDMA2000_PCH_EN	■ Set (1) this bit to enable pilot channel transmission. ■ In IS-95-A mode, this bit should be cleared (0).
7	UNUSED_BIT7	■
6		Reserved.
5	IS95A_ACCESS_CH	This bit selects the access channel mode for the interleaving process. Different hardware settings for the interleaver are selected for different channel modes. ■ For reverse traffic channel processing, clear (0) this bit. ■ For access channel processing, set (1) this bit. This bit is written PA_WARMUP time before the PN20 frame boundary.
4		Reserved.
3	BYTE_ACCESS	■ Set (1) this bit if the writes into the ERAM are in bytes. ■ Clear (0) this bit for half-word (16-bit) writes.

MOD_MISC_CTL (Continued)

Bits	Name	Description
2	BYTE_POS	This bit indicates which byte of the 16-bit bus is valid, in the case of BYTE_ACCESS. 0: LS byte [7:0] 1: MS byte [15:8] The BYTE_ACCESS bit must be set for this bit to be active.
1:0	PA_CTL	These bits select normal operation or specify the output state of the PA_PUNCT/ TX_PUNCT signal (for these settings, X indicates a discretionary setting): Bit [1:0] PA PUNCT/ TX_PUNCT 01 Normal function 00 Low 1X High In normal functioning of PA_PUNCT/ TX_PUNCT, PA_PUNCT/ TX_PUNCT = 1: Power amplifier transmit s PA_PUNCT/ TX_PUNCT = 0: Power amplifier is shut off The PA_PUNCT/ TX_PUNCT signal is used to turn on and off the power amplifier. The timing of this pin is programmable using the PA_WARMUP/ TX_WARMUP register. The PA_PUNCT/ TX_PUNCT signal is similar in timing to MASK_DATA, except that it is asserted earlier using the PA_WARMUP/ TX_WARMUP register to provide sufficient time for the power amplifier to warm up before the actual data transmission takes place. Assertion of PA_PUNCT/ TX_PUNCT is earlier than the assertion of MASK_DATA (programmable by the PA_WARMUP/ TX_WARMUP register) while deassertion of PA_PUNCT/ TX_PUNCT follows deassertion of MASK_DATA by 19 PNChips.

0x00BC MOD_PCH_GAIN

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	PCH_GAIN	The MOD_PCH_GAIN register holds the reverse pilot channel gain. In the IS-95A mode, write 0x00 into this register.

0x00C0 MOD_DCCH_GAIN

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DCCH_GAIN	The MOD_DCCH_GAIN register is the gain for the reverse dedicated control channel. In IS-95-A, write 0x0000 into this register.

0x00C4 MOD_FCH_GAIN

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	FCH_GAIN	The MOD_FCH_GAIN register is the fundamental channel gain for the reverse link modulator. In the IS-95-A mode, write 0x0000 into this register.

0x00C8 MOD_SCH_GAIN

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	SCH_GAIN	The MOD_SCH_GAIN register is the supplemental channel gain for the reverse link modulator.

0x00CC MOD_PRMBL_CTL_0

Type: Write
Clock: CDMA_TX_CLK
The MOD_PRMBL_CTL_0 register is the control register for the enhanced access channel and reverse traffic channel preambles for RC3 to RC6.

Bits	Name	Description
12:9	PREAMBLE_PCG_POSITION	PCG value during a 20ms duration (0 and 15), when the transmission the preamble should begin.
8:5	PREAMBLE_FRAC_DURATION	Individual preamble transmit burst size in multiples of 1.25 ms in enhanced access channel preamble.
4:1	PREAMBLE_NUM_FRAC	Number of preamble bursts in enhanced access channel preamble or reverse traffic channel preamble.
0	PREAMBLE_ENA	■ To enable preambles, set (1) this bit. ■ To disable, clear (0) this bit.

0x00D0 MOD_PRMBL_CTL_1

Type: Write
Clock: CDMA_TX_CLK
The MOD_PRMBL_CTL_1 register is the preambles for the enhanced access channel and reverse traffic channel for RC3 to RC6.

Bits	Name	Description
8:4	PREAMBLE_ADD_DURATION	Final preamble burst transmit time in multiples of 1.25 ms in enhanced access channel preamble.
3:0	PREAMBLE_OFF_DURATION	Individual preamble burst idle time in multiples of 1.25 ms in enhanced access channel preamble. For traffic channel preambles, clear (0) these bits.

0x00D4 MOD_ROTATOR_MAP

Type: Write
Clock: CDMA_TX_CLK

The MOD_ROTATOR_MAP controls the modulator phase rotator. Rotation is one of 0, 90, 180 or 270 degrees. The contents of this register, along with the value of the PA_STATE signal (an internal signal in the MSM), is sent from the AGC to the modulator to determine the actual angle through which the modulator output is rotated.

Bits	Name	Description
7:6	STATE11_MAP	When PA_STATE = 00, the contents of [7:6] are used as the rotation angle.
5:4	STATE10_MAP	When PA_STATE = 01, the contents of [5:4] are used as the rotation angle.
3:2	STATE01_MAP	When PA_STATE = 10, the contents of [3:2] are used as the rotation angle.
1:0	STATE00_MAP	When PA_STATE = 11, the contents of [1:0] are used as the rotation angle.
		Rotation angle bits — **Rotation angle**
		00 — angle 0
		01 — angle 90
		10 — angle 180
		1 — angle 270

0x00D8 MOD_WSYM_STATE

Type: Write
Clock: CDMA_TX_CLK

The MOD_WSYM_STATE register is the output timing adjustment register for the Walsh symbol clock. This register specifies the position, as a time offset, for the WSYM_CLK active pulse. The WSYM_CLK active pulse is generated during the second Walsh symbol of every power control group. The time offset is measured from the second Walsh symbol boundary of a power control group to the WSYM_CLK active pulse. The WSYM_CLK active pulse is offset, from the beginning of the second Walsh symbol boundary, by the sum of the value specified by this register plus 12 CHIPX8 clock periods. The value of WSYM_STATE is always positive and the range of values allowed is from 0 to 252.

Bits	Name	Description
7:2	WSYM_CLK_WCHIP	This bitfield specifies the time offset in units of the number of Walsh chips.
1:0	WSYM_CLK_PN	This bitfield specifies the time offset in units of the number of PN chips.

0x00DC TX_VERY_EARLY_FRAME_CTL

Type: Write
Clock: CDMA_TX_CLK

The TX_VERY_EARLY_FRAME_CTL register is the transmit very early frame control register. This register specifies when the TX_VERY_EARLY_FRAME signal is sent to the transmit AGC block.

Bits	Name	Description
12:2	VERY_EARLY_FRAME_ADV	The TX_VERY_EARLY_FRAME signal is strobed VERY_EARLY_FRAME_ADV PN chips before the next 5, 10, or 20 ms frame boundary that is specified by the VERY_EARLY_FRAME_PERIOD value. For normal operation program this value to 100.
1:0	VERY_EARLY_FRAME_PER	This bitfield specifies the period of the TX_VERY_EARLY_FRAME signal. The default is the same as the frame size being transmitted. 11: Disables (the TX_VERY_EARLY_FRAME signal) 10: every_5_ms (TX_VERY_EARLY_FRAME occurs once) 01: every_10_ms (TX_VERY_EARLY_FRAME occurs once) 00: every_20_ms (TX_VERY_EARLY_FRAME occurs once)

0x00E0 TX_2_EARLY_PCG_CTL

Type: Write
Clock: CDMA_TX_CLK

The TX_2_EARLY_PCG_CTL register is the Transmit Early Frame Control Register. This register specifies when the EARLY_2_PCG_EN signal is sent to the transmit AGC block. The PCG_EN signal is produced by delaying this signal by THW1 (64 chips).

Bits	Name	Description
12:2	EARLY_2_PCG_EN_ADV	The EARLY_2_PCG_EN signal is strobed EARLY_2_PCG_EN_ADV PN chips before the next PCG, 5, 10, or 20 ms frame boundary that is specified by EARLY_2_PCG_EN_PER value. For normal operation, use THW1 (64 PN chips).
1:0	EARLY_2_PCG_EN_PER	This bitfield specifies the period of the TX_EARLY_2_PCG_EN signal. 11: PCG (Occurs once every PCG, normal operation). 10: 5ms (Occurs once every 5 ms) 01: 10ms (Occurs once every 10ms) 00: 20ms (Occurs once every 20ms) During normal operation, use 11.

0x00E4 MOD_PCH_PCBIT_DATA

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	When the corresponding bit of the MOD_PCH_PCBIT_MASK register is set, the contents of the MOD_PCH_PCBIT_DATA register is sent on the pilot channel during the last 25% of each PCG. The order of transmission is 0, 1, 2,.....through 15.

0x00E8 MOD_PCH_PCBIT_MASK

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The contents of the MOD_PCH_PCBIT_MASK register determine whether the PC bit from the combiner or the corresponding bit in the MOD_PCH_PCBIT_DATA register is sent on the pilot channel during the last 25% of each PCG. If bit i is set (1), the MOD_PCH_PCBIT_DATA(i) is transmitted on the pilot channel instead of the PC bit that is provided by the demodulator.

0x00EC MOD_WCOVER_SEL

Type: Write
Clock: CDMA_TX_CLK

The MOD_WCOVER_SEL register selects the Walsh covers for modulator channels 1, 2, and 3.

Bits	Name	Description
11:8	CH1_WALSHCOVER_SEL	This bit selects one of the 16 Walsh covers for channel 1.
7:4	CH2_WALSHCOVER_SEL	This bit selects one of the 16 Walsh covers for channel 2.
3:0	CH3_WALSHCOVER_SEL	■ This bit selects one of the 16 Walsh covers for channel 2.

0x00F0 TX_WARMUP

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The TX_WARMUP register is the power amplifier warm-up period register. This register controls the time defined as WARMUP_PERIOD. When TX_PUNCT goes high prior to CDMA data transmission, the WARMUP_PERIOD is the time interval required to switch on the power amplifier and have the bias of the power amplifier reach a steady state. The value that must be written to this register is defined in ms using Equation : TX_WARMUP = 153.6 x WARMUP_PERIOD + 64. The range of values allowed for TX_WARMUP is from 65 to 162, which corresponds to a WARMUP_PERIOD range of 0.007 ms to 0.638 ms. Typical values are less than or equal to 162 (0.658 ms). WARMUP_PERIOD is determined using the NV item, NV_SUBPCG_PA_WARMUP_DELAY, as follows: WARMUP_PERIOD (in ms) = NV_SUBPCG_PA_WARMUP_DELAY (in chips) / 1228.8. Valid values for NV_SUBPCG_PA_WARMUP_DELAY are in [8, 784] range, default value being 784. This register controls the TX_PUNCT output when the PA_CTL bit of the MOD_MISC_CTL register and the MASK_CTL bit of the MOD_TEST_CTL register are defined for the normal function (not set to define TX_PUNCT to a fixed state). Updates to the MOD_FCH_CTL register should be completed after the WARMUP_PERIOD, before the beginning of the next frame. If this is not done, the TX_PUNCT signal, which controls the TX_ON output pin of the MSM, which turns on and off the power amplifier, will not function correctly.

0x0228 MOD_STATUS

Type: Read
Clock: CDMA_TX_CLK
The MOD_STATUS register is the modulator status register.

MOD_STATUS

Bits	Name	Description
8:7	TX_FR_INT_SLOT	These bits indicate the 5 ms frame slot during which the last TX frame interrupt was issued.
6	CH1_CH2_ENC_POS_ERR	This bit indicates whether CH1 and CH2 have been programmed to start encoding at the same time.
5	CH1_ENC	This bit indicates whether the modulator channel 1 reverse link convolutional encoder is busy or idle.
4	CH1_ENC_WR_ERROR	This bit indicates the fault status of the reverse link convolutional encoder for modulator channel 1. ■
3	CH3_ENC	This bit indicates whether the modulator channel 3 reverse link convolutional encoder is busy or idle.
2	CH3_ENC_WR_ERROR	This bit indicates the fault status of the reverse link convolutional encoder for modulator channel 3. ■
1	CH2_ENC	This bit indicates whether the modulator channel 2 reverse link convolutional encoder is busy or idle.
0	CH2_ENC_WR_ERROR	This bit indicates the fault status of the reverse link convolutional encoder for modulator channel 2. ■

0x0224 MASK_DATA

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	The MASK_DATA register is the MASK_DATA value register. In IS-95-A operation, this register returns the value of the internal MASK_DATA signal. The MASK_DATA signal indicates whether the current PCG is transmitted or not (based on data burst randomization). If MASK_DATA = 1, the current PCG is to be transmitted. If MASK_DATA = 0, the current PCG is not transmitted. In the cdma2000 mode, this register indicates whether there is transmission on the pilot.

0x020C Reserved.

0x0220 Reserved.

0x0110 Reserved.

0x0114 Reserved.

0x0118 Reserved.

0x011C Reserved.

0x022C Reserved.

17.2.2 HDR Modulator software registers

This section describes the HDR Modulator micro registers

0x0120 REVMOD_TIME_STAMP_CTL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	The REVMOD_TIME_STAMP_CTL register contains control bit(s) associated with the time stamp features of the REVMOD. Writing a '1' value to bit 0 commands a snapshot of the current REVMOD's system time count to be placed into the REVMOD_TIME_STAMP and REVMOD_TIME_STAMP2 registers.

0x023C REVMOD_TIME_STAMP

Type: Read
Clock: CDMA_TX_CLK

REVMOD_TIME_STAMP register is loaded with the latest modulator time count information after a 1 is written to the REVMOD_TIME_STAMP_CTL snapshot bit.

Bits	Name	Description
12:6	FRAME_COUNT	7-bit frame count value.
5:0	UPPER_6_BIT_TIME_COUN	Upper 6-bits of modulator time count value (1/4 slot resolution).

0x0240 REVMOD_TIME_STAMP2

Type: Read
Clock: CDMA_TX_CLK

The REVMOD_TIME_STAMP2 register is the modulator time stamp 2 register. This register contains additional bits of resolution for the latched modulator time count value that is present in this register and in the REVMOD_TIME_STAMP register. The value in this register is loaded with the latest modulator time count information, in parallel with the REVMOD_TIME_STAMP register, after a 1 is written to the time stamp snapshot bit in the REVMOD_TIME_STAMP_CTL register.

Bits	Name	Description
15	EVEN_SECOND_COUNT	Bit 15 is the 1-bit even second count.
14:9	ZEROS	Bits 14:9 are tied to zero.
8:0	LOWER_9_BIT_CHIP_COUNT	The 9 LS-bits (8:0) are the 9 LS-bits of the HDR modulator chip count.

0x0124 REVMOD_SLOT_INT_OFFSET

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
3:0	DATA	The REVMOD_SLOT_INT_OFFSET register is the slot interrupt offset register. This 4-bit register contains a slot number between 0 and 15, which indicates at what slot offset within a frame the slot interrupt is generated. **Note:** Frame and slot timing is aligned with the REVMOD's internal PN rollover boundary.

0x0128 REVMOD_FRAME_OFFSET

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
3:0	DATA	The REVMOD_FRAME_OFFSET register is the traffic channel frame offset register. This register specifies on which of the 16 slot boundaries within a frame that the MS traffic channel begins transmission of a reverse-link data packet.

0x012C REVMOD_MOD_CTL

Type: Write
Clock: CDMA_TX_CLK

The REVMOD_MOD_CTL register is the modulation control register. Each bit in this register is an enable (1) or disable (0) control bit for the unbalanced QPSK modulation scheme used in the reverse link transmit chain.

Bits	Name	Description
0	IQ_MOD_CTL	For normal PN spreading operation, this bit is cleared (0). When this bit is set (1), both the I and Q channel PN spreading inputs are forced low, and spreading is turned off.

0x0130 REVMOD_TENC_CTL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	The REVMOD_TENC_CTL register is the turbo encoder control register. Bit 0 of this register is the EDI write-in-progress flag, which is used as a control flag for the modulator hardware to indicate when the processor is writing a data packet into the TEDI interface. ■ This bit is set (1) while the processor is writing data packet bytes into the TEDI interface. ■ This bit is cleared (0) as soon as the processor has finished writing the data packet.

0x0234 REVMOD_TENC_STATUS

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
2	ENC_WAIT_STATUS	This status bit indicates whether the microprocessor successfully wrote the last data packet into the turbo encoder before the maximum encoding time point that occurs exactly one slot before the frame offset. This bit is set (1), if the EDI write-in-progress flag in the REVMOD_TENC_CTL register has not been cleared (0) at the one-slot maximum encoding time point. The modulator TEDI block holds off transmission of the current data packet until one frame time (26.67 ms) later. This status bit is set for only 26.67 ms and is cleared as soon as data packet transmission begins. The processor must sample the value of the status bit every PN roll (26.67 ms) two slots before the frame offset boundary
1	TDM_EDO_ERR	This error flag indicates whether the turbo encoder met the expected one-slot maximum encode time for the last data packet that was written to the encoder. ■ If this bit is cleared (0), the turbo encoder successfully encoded the last data packet by the time the frame offset slot arrived. ■ If this bit is set (1), an unexpected error occurred within the turbo encoder, causing the encoded data packet to not be ready by the frame offset slot. If this error flag is set, it may be cleared 26.67 ms later. The processor must sample the flag value every PN roll (26.67 ms) before the frame offset boundary
0	EDI_READY	This bit indicates whether the turbo encoder is ready to begin encoding a new data packet. If this bit is set (1), the microprocessor can begin writing a new data packet into the EDI interface. If this bit is cleared (0), the turbo encoder is still in the process of encoding a data packet. The microprocessor must wait an additional frame period (26.67 ms) before sampling this status bit again.

0x0134 REVMOD_RRI_REQUEST

Type: Read/Write
Clock: CDMA_TX_CLK

Bits	Name	Description
3	access_pilot	The access_pilot register is the reverse access pilot request register. This register holds the request. Before writing a data packet into the turbo encoder, the processor writes the request into this register. The request must be written into this register four slots before the frame offset slot on which data transmission begins. If the turbo encoder successfully encoded the corresponding data packet, the value in this register is cleared (0) one-half slot before the frame offset slot.
2:0	RRI_request	The RRI_request register is the reverse rate index request register. This register holds the 3-bit requested reverse link rate index value. Before writing a data packet into the turbo encoder, the processor writes the index value of the requested rate into this register. The requested rate index must be written into this register four slots before the frame offset slot on which data transmission begins. If the turbo encoder successfully encoded the corresponding data packet, the value in this register is cleared (0) one-half slot before the frame offset slot.

Table 17-1 lists the 3-bit index values for the reverse link data rates.

Table 17-1 Index values for reverse link data rates

Value	Data rate
000	Pilot only
001	9.6 kbps
010	19.2 kbps
011	38.4 kbps
100	76.8 kbps
101	153.6 kbps
110	Reserved
111	Reserved

0x0138+4w REVMOD_PN_LONG_STATE_w, w=[0..2], totalbits=42

Type: Write
Clock: CDMA_TX_CLK

The REVMOD_PN_LONG_STATE register is the long code PN state register. Writes to this register specify the long code PN initial state for the user long code PNGEN. This register is broken out into three 16-bit registers with individual addresses.

For w=0,

Bits	Name	Description
9:0	DATA	This field is bits 41:32 of REVMOD_PN_LONG_STATE.

For w=1,

Bits	Name	Description
15:0	DATA	This field is bits 31:16.

For w=2,

Bits	Name	Description
15:0	DATA	This field is bits 15:0

0x0144+4w REVMOD_PN_I_LONG_MASK_w, w=[0..2]

Type: Write
Clock: CDMA_TX_CLK

The REVMOD_PN_I_LONG_STATE register is the long code PN I mask register. This register specifies the long code PN I-channel mask value. This register is broken out into three 16-bit registers with individual addresses.

For w=0,

Bits	Name	Description
9:0	DATA	This field includes bits 41:32 of REVMOD_PN_I_LONG_MASK.

For w=1,

Bits	Name	Description
15:0	DATA	This field includes bits 31:16.

For w=2,

Bits	Name	Description
15:0	DATA	This field includes bits 15:0

0x0150+4w REVMOD_PN_Q_LONG_MASK_w, w=[0..2]

Type: Write
Clock: CDMA_TX_CLK
Bits: 41:0

The REVMOD_PN_Q_LONG_STATE register is the long code PN Q mask register. This register specifies the long code PN Q-channel mask value. This register is broken out into three 16-bit registers with individual addresses.

For w=0,

Bits	Name	Description
9:0	DATA	This field is bits 41:32 of REVMOD_PN_Q_LONG_MASK.

For w=1,

Bits	Name	Description
15:0	DATA	This field is bits 31:16.

For w=2,

Bits	Name	Description
15:0	DATA	This field is bits 15:0

0x015C REVMOD_PN_CTL

Type: Read/Write
Clock: CDMA_TX_CLK

The REVMOD_PN_CTL register is the PN generator control register. All register updates occur on the next 26.67 msec frame rollover. A read of the value of a control bit indicates whether the operation has completed. Additional microprocessor writes to an already-set control bit are ignored.

REVMOD_PN_CTL

Bits	Name	Description
3	BOZO_MODE	0: Bozo mode is enabled (default) 1: Bozo mode is disabled ■ When the Bozo mode is enabled, the long code state is reloaded automatically by hardware on EVERY 26.67 msec frame roll. ■ When the Bozo mode is disabled, the long code state is reloaded only when a PN long code load command (control bit 4) is pending.

REVMOD_PN_CTL (Continued)

Bits	Name	Description
2	PN_LONG_CODE_LOAD	Bits 2:0 of this register contain long and short PN generator control bits. Each load control bit commands a load of a PN generator initial state or mask value. 1: in progress 0: done
1	PN_I_LONG_CODE_LOAD	
0	PN_Q_LONG_CODE_LOAD	

0x0160 REVMOD_PA_CTL

Type: Write
Clock: CDMA_TX_CLK

The REVMOD_PA_CTL register is the power amplifier (PA) control register. This register contains control bits that are used to control the RF transmit chain, including two bits to control the state of the reverse link PA.

REVMOD_PA_CTL

Bits	Name	Description
8	TX_WARMUP_EN	This bit control the tx_punct turn on before frame offset boundary. The turn on time before the frame offset boundary is defined by REVMOD_TX_WARMUP register in chips resolution of 1 slot period. 1: tx_warmup_on (TX_WARMUP function is enable) 0: tx_warmup_off (tx_punct is on at the frame offset boundary)
7	PA_WARMUP_EN	This bit control the pa_punct turn on before frame offset boundary. The turn on time before the frame offset boundary is defined by REVMOD_PA_WARMUP register in chips resolution of 1 slot period. 1: pa_warmup_on (PA_WARMUP function is enable) 0: pa_warmup_off (pa_punct is asserted at the frame offset boundary)
6	PA_POWERUP_SEL	This bit controls the function of the PA_ON2 pin. 1: PA_POWERUP is selected 0: PA_ON2 is selected
5	TX_PUNC_CTL	This bit controls the state of the internal <tx_punct> signal. 0: tx_punct_low (<tx_punct> goes low on the next frame offset boundary, as specified by the value in the frame offset register) 1: tx_punct_high (<tx_punct> goes high on the next frame offset boundary) The <tx_punct> signal is internally ANDed with the <mp_pa_on> signal that is controlled by the PA state control bit. Controlling the <tx_punct> signal on the PN roll boundaries allows instant-off gating of the TX RF PA. Turning off the PA at frame, or transmit data, slot offset boundaries is useful during transmission of reverse link access probe messages . The default state of the TX puncture control bit on power-up is inactive. While the reverse link transmit chain is active, set (1) the TX puncture control bit, so that the PA is active.

REVMOD_PA_CTL (Continued)

Bits	Name	Description
4	DRC_DISABLE	This bit controls the functionality of the DRC channel. 0: drc_enable (the DRC channel is enabled and code division multiplexed onto the traffic/DRC Q channel) 1: drc_gated_off (the DRC channel is gated to all zero bits) Writing to this bit takes immediate effect. The DRC disable control bit can be used to gate off the DRC channel when sending reverse link access probes.
3	PA_PUNC_CTL	This bit controls the state of the internal <pa_punct> signal. 0: pa_punct_low (<pa_punct> goes low on the next frame offset boundary, as specified by the value in the frame offset register) 1: pa_punct_high (<pa_punct> goes high on the next frame offset boundary) The <pa_punct> signal is sent to WEB core to form the <pa_on> and <pa_on2> equation for HDR. Controlling the <pa_punct> signal on the PN roll boundaries allows instant-off gating of the TX RF PA. Turning off the PA at frame, or transmit data, slot offset boundaries is useful during transmission of reverse link access probe messages . The default state of the pa puncture control bit on power-up is inactive. While the reverse link transmit chain is active, set (1) the PA puncture control bit, so that the PA is active.
2	MOD_MASK_CTL	This bit controls the functionality of the REVMOD's I and Q channel data outputs to the FIR. 1: iq_mask_on (the I and Q channel data values are masked to a value of 0 on the next 26.67 msec PN roll boundary) 0: iq_mask_off (the I and Q channel data outputs to the FIR contain normal unmasked data)
1	RRI_DISABLE	This bit controls the functionality of the RRI channel. 0: rri_on (the RRI channel will be enabled and time division multiplexed onto the Pilot/RRI I channel) 1: rri_gated_off (the RRI channel is gated to all zero bits) Writing to this bit takes immediate effect. The RRI disable control bit can be used to gate off the RRI channel when sending reverse link access probes.
0	RESERVED_BIT0	

0x0164 REVMOD_FRM_CNT_OFFSET

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
6:0	DATA	The REVMOD_FRM_CNT_OFFSET register is the frame count offset register. This register contains a 7-bit frame count value that can be loaded into the frame counter for the time stamp block on the next PN rollover or frame boundary. The software can also use this register to synchronize the current modulator frame count to system time, which provides greater time stamping accuracy.

0x0168 REVMOD_FRM_CNT_OFFSET_CTL

Type: Read/Write
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	The REVMOD_FRM_CNT_OFFSET_CTL register is the frame count offset control register. This register contains a control bit [0] that is associated with the REVMOD_FRM_CNT_OFFSET register. Setting (1) this bit commands a load of the time stamp block's frame count register with the value in the REVMOD_FRM_CNT_OFFSET register on the next PN rollover, or frame, boundary. A read of this bit indicates whether the operation has completed (1 = in progress, 0 = done). Additional microprocessor writes to this control bit are ignored until the operation is complete.

0x016C REVMOD_PA_WARMUP

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
10:0	DATA	The REVMOD_PA_WARMUP register is used to turn on tx_punct signal before frame offset boundary. Then the PA_ON can be turned on earlier. REVMOD_PA_WARMUP is defined as chip resolution for 1 slot period. Micro should program this register before 1 slot prior to frame offset boundary. If we want PA_ON be turned on 200 chips before frame offset boundary, we should program 2047-200 = 1847 to this register. This register value is observed by hardware only at 1 slot prior to frame offset boundary.

0x0208 REVMOD_TX_WARMUP

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
10:0	DATA	The REVMOD_TX_WARMUP register is used to turn on tx_punct signal before frame offset boundary. Then the TX_ON can be turned on earlier. REVMOD_TX_WARMUP is defined as chip resolution for 1 slot period. Micro should program this register before 1 slot prior to frame offset boundary. If we want TX_ON be turned on 200 chips before frame offset boundary, we should program 2047-200 = 1847 to this register. This register value is observed by hardware only at 1 slot prior to frame offset boundary.

0x0170 REVMOD_ENC_MODE

Type: Read/Write
Clock: CDMA_TX_CLK

The ENC_MODE register determines the encoder mode.

REVMOD_ENC_MODE

Bits	Name	Description
4	ENDIAN_MODE	This bit determines the endian mode. 0: big endian 1: little endian
3	UNUSED_BIT3	
2	CHN_INT_OFF	Turns off the channel interleaver 0: channel interleaver is on (default after reset). 1: channel interleaver is off
1:0	UNUSED_BITS1_0	

0x0174 REVMOD_EDI_CONTROL

Type: Read/Write (or read-only, as indicated)
Clock: CDMA_TX_CLK

The EDI_CONTROL register is the EDI control register.

Bits	Name	Description
2	EDI_ABORT	Trashes all frame data before end of frame. This signal has no effect if it comes after the end of the frame. Valid in both EDI_HOST_MODEs.

Bits	Name	Description
1	EDI_EOF	Signals the end of the frame 1: end of frame. 0: no end of frame. EDI_EOF is cleared after reset and after the transfer is completed. Write is disabled when not in EDI_HOST_MODE.
0	EDI_READY	Read-only register 1: EDI ready for new data byte. 0: EDI not ready. Valid in both EDI_HOST_MODEs

0x0178 REVMOD_EDI_DATA

Type: Read/Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The EDI_DATA register is the encoder dara input register. It contains the input data bits for encoding in ENC_HOST_MODE (first in first out, MSB first). Writing of this address is effective only when the EDI_READY bit is set in the EDI_CONTROL register. Reading this register returns the last value written to it.

0x017C REVMOD_ENC_RATE

Type: Write
Clock: CDMA_TX_CLK

The ENC_RATE register is the reverse rate indicator.

Bits	Name	Description
2:0	RR_INDEX	000: data rate 0 001: data rate 9.6 kbps 010: data rate 19.2 kbps 011: data rate 38.4 kbps 100: data rate 76.8 kbps 101: data rate 153.6 kbps

0x0210 REVMOD_TX_TIME_LATCH

Type: Write/Command
Clock: CDMA_TX_CLK

Bits	Name	Description
0	LATCH	

0x0238 REVMOD_TX_TIME_RD

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
25:0	DATA	This register returns the HDR Tx time counter value latched at hdr_pn_rollover_2d boundary or the time the REVMOD_TX_TIME_LATCH is issued. The resolution of the counter is chipx8.

17.2.3 Tx AGC software registers

BASE ADDRESS = same as Modulator 0x0400

This file contains the read and write TXAGC registers.

17.2.3.1 Tx AGC software write registers

0x0180 TX_AGC_CTL

Type: Write
Clock: CDMA_TX_CLK

The AGC_CTL register is the digital AGC control register.

TX_AGC_CTL

Bits	Name	Description
15:14	TX_GAIN_ADJ_STEP_SIZE	These bits control the power adjustment step size which is applied after each up/down decision of the TX_GAIN_ADJ accumulator. **Value** **Step size** 00 1 dB (IS-95 compatible mode) 01 0.5 dB 10 0.25 dB This encoding is undefined: 11.
13	LINEARIZER_RANGE_SEL	This bit determines the resolution of AGC_VALUE and TX_OPEN_LOOP (10 bits, 8+2 format). If this bit is clear (0), the 10-bit AGC_VALUE has a resolution of 1/12 dB, and TX_OPEN_LOOP has a resolution of –1/12 dB. If this bit is set (1), AGC_VALUE has a resolution of 1/10 dB, and TX_OPEN_LOOP has a resolution of –1/10 dB. ■ Clear (0) this bit to select the 85.3 dB range ■ Set (1) this bit to select the 102.4 dB range
12	RATCHET_MODE	This bit controls the microprocessor-controlled ratchet mode. ■ Set (1) this bit to force the CAGC circuit to set (1) the ratchet_mode bit output to the demodulator power combiner circuit, **only** if the transmit power **is** greater than the value in the TX_GAIN_LIMIT registers. ■ Clear (0) this bit to enable the CAGC ratchet mode operations. Note that ratcheting is still a function of the up and down cell decisions. So, ratcheting occurs only if the cell decisions are to increase transmit power
11	TX_AGC_ADJ_OE	This bit is the CDMA TX_AGC_ADJ PDM output enable. Clearing (0) this bit causes the TX_AGC_ADJ PDM output to go into a high impedance state.

TX_AGC_CTL (Continued)

Bits	Name	Description
10	TX_AGC_ADJ_POLARITY	■ To output active high signals, clear (0) the appropriate bits. ■ To output active low signals, set (1) the appropriate bits. The bits have the following-pin assignments: **Control signal name** / **Pin name** Bit 10: TX_POLARITY / TX_AGC_ADJ Bit 9: PA_POLARITY / PA_R0, PA_R1
9	PA_R_POLARITY	
8	TX_AGC_ADJ_OVERRIDE_N	■ To receive data from the AGC unit, set (1) these bits. ■ To override data from the AGC unit, clear (0) these bits. The bits have the following-pin assignments: **Control signal name** / **PDM / pin name** Bit 8: TX_AGC_ADJ_OVERRIDE_N / TX_AGC_ADJ Bit 7: PA_R_OVERRIDE_N / PA_R0, PA_R1 **NOTE:** TX_POLARITY bits remain active even when the TX_AGC_ADJ_OVERRIDE_N bits are clear (0). PA_POLARITY is inactive when the PA_R_OVERRIDE_N bits are clear (0).
7	PA_R_OVERRIDE_N	
6:5	PA_R_WR	This bit controls the output value of the PA_R1 and PA_R0 pins. This is valid only if the PA_R_OVERRIDE_N bit is clear (0).
4:3	TX_GAIN_BLIP_DELAY	These bits are the select control signal for blip delay mux, that controls the delay on the PA_RANGE[1:0] whenever the PA power is stepped up and the PA range state machine is changing the gain state from lower to higher. This will help guarantee a negative gain blip while things are settling. The delay is measured from PCG_EN boundary specified by the modulator. The following bit values select the corresponding delay times: 00: no delay 01: 6.5 µs 10: 13 µs (normal operation) 11: 26 µs
2:0	MASK_DELAY	Values from 0x0 to 0x7 where 0x0 corresponds to a delay of 1.25 ms (1 PCG) and 0x7 corresponds to a delay of 10 ms (8 PCGs). The resolution is 1.25 ms /LSB. This register takes care of the MASK_DATA delay for all rate sets. The following table gives the details: MASK_DELAY VALUE: MASK_STREAM 0x0: delay of 1 PCG 0x1: delay of 2 PCGs 0x2: delay of 3 PCGs 0x3: delay of 4 PCGs 0x4: delay of 5 PCGs 0x5: delay of 6 PCGs 0x6: delay of 7 PCGs 0x7: delay of 8 PCGs This delay should be set to 1 (i.e., 2 PCGs) for IS95 A/B.

0x0184 TX_AGC_CTL2

Type: Write
Clock: CDMA_TX_CLK

The AGC_CTL register is the digital AGC control register.

Bits	Name	Description
15:8	RESERVED_BITS15_8	
7	TEMPORAL_HYST	Set (1) this bit to enable the temporal hysteresis control in the PA range control circuit if the TX_MSM3000_MODE bit is set (1). If the TX_MSM3000_MODE is clear (0), this bit must be set (1) regardless if temporal hysteresis is desired. Temporal hysteresis can be disabled by clearing (0x00) PA_R_TIMER.
6	TX_MSM3000_MODE	These two registers contain data used for RX AGC and TX AGC backward compatibility. ■ For MSM3000 operation, set (1) these bits. ■ For MSM3100 operation, clear (0) these bits.
5	TX_MSM3100_MODE	This bit selects the MSM3100 mode. 0: default 1: MSM3100 mode (a backward compatible mode)
4	FREEZE_PA_STATE	Set (1) this bit during the Linearizing-table update, so that PA range is frozen. After the table update is finished, reset by clearing (0) this bit. Used in MSM3100 and MSM3300 mode only.
3	PWR_CTL_EN	This bit controls the TX_GAIN_ADJ accumulator. ■ For closed-loop power control operation, set (1) this bit. ■ To override the data from the power control unit, clear (0) this bit.
2	TX_OPEN_LOOP_OVERRIDE	■ Set (1) this bit to freeze the TX_OPEN_LOOP accumulator and permit writing it directly via TX_OPEN_LOOP_WR. ■ Clear (0) this bit to permit normal operation of the TX_OPEN_LOOP accumulator.
1	RATE_ADJ_READY	This bit is used to indicate whether the TX_RATE_ADJ will be available (ready) for sampling at the VERY_EARLY_FRAME (the VERY_EARLY_FRAME is a signal from the modulator to the CAGC. 0: no 1: yes
0	PA_R_MAP_EN	Set this bit to enable the re-mapping of the PA range control state machine output states to pre-defined pin values as specified in the PA_R_MAP register.

0x0188 TX_AGC_RESET

Type: Write/Command
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	Write the TX_AGC_RESET register to reset the TX AGC.

0x018C Reserved.

0x0190 AGC_TX_MASK_DATA_SEL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	The bit 0 is for MASK_DATA_SEL_N. In normal operation, it is cleared to '0', and the mask_delayed_data signal, which is the delayed version of mask_data, is used as the enable signal of the power control bit accumulator, pwrctl_data_en. Before the modulator starts to transmit RCCCH preamble, the microprocessor programs the programmable delay value for tx_rccch_frame_start, and sets mask_data_sel_n to be '1'. At the beginning of RCCCH frame, tx_rccch_frame_start asserts a pulse, and triggers the delay counter. Once the counter expires, pwrctl_data_en becomes '1', and the power control bit accumulator is enabled. After the RCCCH frame is transmitted, the microprocessor should clear mask_data_sel_n to '0', so that pwrctl_data_en signal switches back to mask_delayed_data.

0x0194 AGC_TX_RCCCH_FRAME_DELAY

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
6:0	DATA	The programmable delay value for TX_RCCCH_FRAME_START from the modulator. Its resolution is per PCG.

0x0198 PA_R_MAP

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	When the PA_R_MAP_EN bit in TX_AGC_CTL2 is set, the PA_R_MAP register can be used to assign the output states of the CDMA AGC PA range control state machine to predefined values on the pins PA_R1 and PA_R0.

0x019C PA_R_TIMER

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The PA_R_TIMER register controls the timer, which controls the temporal hysteresis in the PA range control circuit. This register specifies the duration of the temporal hysteresis timer interval. The duration is the value of this register either multiplied by 1.25 ms - if TX_MSM3000_MODE is set (1) or if both TX_MSM3000_MODE AND TX_MSM3100_MODE are cleared (0), or multiplied by 208.3 ms if the TX_MSM3000_MODE is cleared (0) but TX_MSM3100_MODE is set (1). A zero duration means that the temporal hysteresis is disabled. Temporal hysteresis restricts the PA Range State machine from switching from a higher to a lower PA range state until the time interval, specified by PA_R_TIMER expires. The timer starts when the desired Tx power falls below a power level defined by a PA_Rx_FALL threshold. No such restrictions apply when switching from a lower to a higher PA state.

0x01A0 PA_R1_FALL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 1 falling threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01A4 PA_R1_RISE

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 1 rising threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01A8 PA_R2_FALL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 2 falling threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01AC PA_R2_RISE

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 2 rising threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01B0 PA_R3_FALL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 3 falling threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01B4 PA_R3_RISE

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	This register contains the 8-bit PA range 3 rising threshold in offset binary format. This register has a –1/3 dB resolution in 85.3 dB mode, or -2/5 dB resolution in 102.4 dB mode.

0x01B8+4w PA_Sw_OFFSET, w=[0..3]

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	The PA_S<X>_OFFSET registers contain four different offset values to be subtracted from the input to VGA linearizer table when PA is working in four different ranges. The data are in 2's complement format. The four offset values all have the resolution of –1/12 dB (85.3 dB mode) or –1/10 dB (102.4 dB mode). Minimum value are all 0x200. Maximum value are all 0x1FF.

0x01C8 Reserved.

0x01CC RAS_RAM_WR_ADDR_RESET

Type: Write/Command
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	For normal operation this bit is a don't care. Prior to download of RAS RAM, set this bit to '1' (to reset the RAS RAM address counter). Hardware will generate a chipx8 pulse to reset the RAS RAM address counter. If, at one time, only a portion of the RAS RAM is downloaded, and then to return to the download for the remaining addresses after some time, do not do the reset of address counter. The address counter retains the state so that one can start downloading from the required address.

0x01D0 RAS_RAM_DATA

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
15:0	DATA	The RAS_RAM_DATA register is the 16 bits RAS RAM linearizer data. A write to this register will initiate a load operation to the RAS RAM, and the address counter will increase by 1.Table 17-2 lists the linearizer data that corresponds to a written address.

Table 17-2 Linearizer data corresponding to written address

Address	Description
0x00 to 0x0F	Tx linearizer table for PA_RANGE[1:0] = 00, bits [15:7] are the offset and bits[6:0] are the slope.
0x10 to 0x1F	Tx linearizer table for PA_RANGE[1:0] = 01, bits [15:7] are the offset and bits[6:0] are the slope.
0x20 to 0x2F	Tx linearizer table for PA_RANGE[1:0] = 10, bits [15:7] are the offset and bits[6:0] are the slope.
0x30 to 0x3F	Tx linearizer table for PA_RANGE[1:0] = 11, bits [15:7] are the offset and bits[6:0] are the slope.

0x01D4

Values depend upon DMSS Software release.

0x01D8 TX_AGC_ADJ_WR

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DATA	The TX_AGC_ADJ_WR register is the TX AGC data register. This data is valid only when the TX_AGC_ADJ_OVERRIDE_N bit of the AGC_CTL2 register is clear (0). A write to this register loads the register value into the PDM circuit.

0x01DC TX_ALIGN_DELAY

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The TX_ALIGN_DELAY register contains the timer value for the SYNC_DELAY timer. This register defines how many clock cycles pass before the Tx linearization outputs are turned off when the PA 4.8 KHz comes. For normal operation, use 223 CHIPX8 clock periods.

0x01E0 TX_ATTEN_LIMIT_WR

Type: Write
Clock: CDMA_TX_CLK

This register provides the TX_ATTEN_LIMIT value and the atten_ratchet_mode bit. The default value is '0' for atten_ratchet_mode, and 0x1ff for TX_ATTEN_LIMIT, after reset.

Bits	Name	Description
10	ATTEN_RATCHET_MODE	This bit controls the microprocessor-controlled ratchet mode for attenuation. Set (1) this bit to force the CAGC circuit to set (1) the atten_ratchet_mode bit output to the demodulator power combiner circuit, only if the transmit power is less than the value in the TX_ATTEN_LIMIT registers. Clear (0) this bit to enable the CAGC attenuation ratchet mode operations. Note that ratcheting is still a function of the up and down cell decisions. So, ratcheting occurs only if the cell decisions are to decrease transmit power. Please refer to the RATCHET_MODE bit in AGC_CTL register for comparisons.
9:0	TX_ATTEN_LIMIT	10-bit 2's complement format. Has the resolution of -1/12 dB (85.3 dB mode), or -1/10 dB (102.4 dB mode).

0x01E4 TX_GAIN_ADJ_WR

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DATA	The TX_GAIN_ADJ_WR register is the accumulator write register. This register loads the specified override value into the TX_GAIN_ADJ accumulator when the PWR_CTL_EN bit of the POWER_COMB_CTL register is clear (0). This register value has a –1/4 dB resolution. The PWR_CTL_EN bit of the POWER_COMB_CTL register must be clear (0) before writing to the TX_GAIN_ADJ_WR register.

0x01E8 TX_GAIN_LIMIT_WR

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	A write to the TX_GAIN_LIMIT_WR register causes the hardware to pick up the current value of the transmit gain limit, as specified by this register. The 10-bit TX_GAIN_LIMIT value has a resolution of –1/12 dB (85.3 dB mode) or -1/10 dB (102.4 dB mode). The TX_GAIN_LIMIT_WR register contains the 10-bit TX_GAIN_LIMIT value in a two's complement format.

0x01EC TX_GAIN_CTL_LATCH

Type: Write/Command
Clock: CDMA_TX_CLK

Bits	Name	Description
0	DATA	This bit is the latch signal for TX_GAIN_CTL reads.

0x01F0 TX_OPEN_LOOP_WR

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	When TX_OPEN_LOOP_OVERRIDE=1, a write to the TX_OPEN_LOOP_WR register loads the TX_OPEN_LOOP accumulator.

0x01F4 TX_PDM_DELAY_VAL

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
7:0	DATA	The TX_PDM_DELAY_VAL register contains an 8-bit positive value to be used for the counter in the overdrive part of TX_AGC_ADJUST PDM. It defines for how many clock cycles the overdrive circuitry of PDM is active.

0x01F8 TX_RATE_ADJ

Type: Write
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	The TX_RATE_ADJ register is the TX AGC gain adjustment for the data rate. This register is in a 2's complement format. The LSB has a resolution of 1/12 dB (85.3 dB mode) or 1/10 dB (102.4 dB mode), depending on the AGC dynamic range setting. Clear this register to all zeros when transmitting in RC1 or RC2.This register can take values from 0x200 to 0x1FF which, respectively, map to 42.65dB and -42.65dB (for RC3 - RC6).

0x01FC Reserved.

0x0200 Reserved.

0x0204 Reserved.

17.2.3.2 Tx AGC software read registers

0x0244 TX_PA_RD

Type: Read
Clock: CDMA_TX_CLK

The TX_PA_RD register returns the logic values that drive the PA_R and LNA_RANGE open-drain output pins.

Bits	Name	Description
2		Reserved.
1	PA_R1_RD	This bitfield provides a mechanism for reading back the values which drive the PA_R1 and PA_R0 pins. Since PA_R1 and PA_R0 are open-drain pin, this readback is limited to the logic value which sources those open drain drivers.
0	PA_R0_RD	

0x0230 RATCHET_BIT_DIS

Type: Read
Clock: CDMA_TX_CLK

This register provides the RATCHET_BIT_DOWN and RATCHET_BIT_UP for microprocessor.

Bits	Name	Description
1	RATCHET_BIT_DOWN	The RATCHET_BIT_DOWN bit returns the status of the Tx power limit block. If RATCHET_BIT_DOWN is set (1), the desired Tx power is above the minimum limit. If RATCHET_BIT_DOWN is clear (0), the desired Tx power is at its minimum limit. When this occurs, the Tx power limit block limits its output to be no greater than TX_ATTEN_LIMIT, and all power down closed-loop power control decisions are ignored.
0	RATCHET_BIT_UP	The same as before. The RATCHET_BIT_UP bit returns the status of the Tx power limit block. If RATCHET_BIT_UP is set (1), the desired Tx power is below the maximum limit. If RATCHET_BIT_UP is clear (0), the desired Tx power is at its maximum limit. When this occurs, the Tx power limit block limits its output to be no less than TX_GAIN_LIMIT, and all power up closed-loop power control decisions are ignored.

0x0198 TX_AGC_ADJ_RD

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DATA	The TX_AGC_ADJ_RD register returns the 9-bit CDMA TX_AGC_ADJ PDM value.

0x0214 TX_GAIN_ADJ_RD

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
8:0	DATA	The TX_GAIN_ADJ_RD register is the accumulator read register that returns the contents of the TX_GAIN_ADJ accumulator. This register has a –1/4 dB resolution.

0x0218 TX_GAIN_CTL_RD

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	The TX_GAIN_CTL_RD register contains the CDMA transmit gain value, in a 2's complement format. This register has –1/12 dB per resolution in 85.3 dB mode, or -1/10 dB resolution in 102.4 dB mode. Before reading TX_GAIN_CTL, first latch the signal to the registers by using AGC_CTL: TX_GAIN_CTL_LATCH.

0x021C TX_OPEN_LOOP_RD

Type: Read
Clock: CDMA_TX_CLK

Bits	Name	Description
9:0	DATA	The TX_OPEN_LOOP_RD register returns the TX_OPEN_LOOP accumulator value.

17.2.4 Modulator Firmware registers

17.2.4.1 HDR modulator firmware registers

0x000 REVMOD_DRC_CODE_INDEX

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
3:0	DATA	The REVMOD_DRC_CODE_INDEX register is the DRC codeword index register. The 4-bit value in this register, or the DRC message, selects one of sixteen unique 8-ary, bi-orthogonal (8,4,4) DRC codewords for BS to MS transmission. There is one-to-one mapping between each of the sixteen DRC codewords and a particular forward link data rate. The reverse-link modulator reports the required transmission rate for the forward traffic channel to the BS on the DRC channel of the reverse link multiplexed traffic/DRC Q channel. Any changes to this register take effect on the next half-slot boundary of the internal modulator timing reference. Table 17-3 summarizes the DRC codeword mapping for bits 3:0.

Table 17-3 DRC codeword mapping

DRC value	Data rate	Slots
0000	NULL rate	
0001	38.4 kbps	16
0010	76.8 kbps	8
0011	153.6 kbps	4
0100	307.2 kbps	2
0101	307.2 kbps	4
0110	614.4 kbps	1
0111	614.4 kbps	2
1000	921.6 kbps	2
1001	1228.8 kbps	1
1010	1228.8 kbps	2
1011	1843.2 kbps	1
1100	2457.6	1
1101	Invalid	

Table 17-3 DRC codeword mapping (Continued)

DRC value	Data rate	Slots
1110	Invalid	
1111	Invalid	

0x0004 REVMOD_DRC_WALSH_COVER

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
2:0	DATA	The REVMOD_DRC_WALSH_COVER register selects up to one of seven base stations (BS must be in the active set) to which the forward traffic channel DRC codeword value is sent. A value of 0x00 indicates that the DRC index value is not intended for any BS in the active set. The DRC codeword, is covered with the DRC Walsh Wn8 cover. Any changes to this register take effect on the next half-slot boundary of the internal modulator timing reference.

0x0008 REVMOD_RF_PN_ROT

Type: Read/Write
Clock: DSP_CLK

The REVMOD_RF_PN_ROT register is the RF PN rotation adjustment register. This 8-bit register contains four 2-bit fields, with each field corresponding to one of the four possible PA control bit values, as controlled by <pa_r[1:0]>. Each 2-bit field denotes one of four possible PN rotation values. The I and Q PN spreading bits are rotated by nπ/2 radians, where n is determined as shown in Table 17-4.

Bits	Name	Description
7:6	RF_PN_ADJUSTMENT_LUT_11	These bits hold a 2-bit PN rotation value when the current PA control bits <pa_r[1:0]> are set to 11.
5:4	RF_PN_ADJUSTMENT_LUT_10	These bits hold a 2-bit PN rotation value when the current PA control bits <pa_r[1:0]> are set to 10.
3:2	RF_PN_ADJUSTMENT_LUT_01	These bits hold a 2-bit PN rotation value when the current PA control bits <pa_r[1:0]> are set to 01.
1:0	RF_PN_ADJUSTMENT_LUT_00	These bits hold a 2-bit PN rotation value when the current PA control bits <pa_r[1:0]> are set to 00.

Table 17-4 PN rotation for LUT value n

LUT value (n)	PN rotation
00	None
01	π/2
10	π
11	3π/2

The PN rotation compensates for any phase change that is introduced into the transmit path by the VGA and/or the PA of the RF transmit chain.

0x000C REVMOD_PDM_CTL

Type: Read/Write
Clock: DSP_CLK

The REVMOD_PDM_CONTROL register is the PDM control register. This register contains control bits that enable/disable and select the output polarity of the TX AGC PDM, PDM 0 and PDM1.

REVMOD_PDM_CTL

Bits	Name	Description
10	TX_AGC_PDM_OUT_POL	This bit selects the output polarity of the TX AGC PDM. Refer to the TX AGC adjust PDM for programming details on how positive and negative polarity selection affects the PDM output. 1: positive 0: negative
9	PDM1_OUT_POL	This bit selects the output polarity for PDM 1. Refer to the TX AGC adjust PDM for programming details on how positive and negative polarity selection affects the PDM output. 1: positive 0: negative
8	PDM0_OUT_POL	This bit selects the output polarity for PDM 0. Refer to the TX AGC adjust PDM for programming details on how positive and negative polarity selection affects the PDM output. 1: positive 0: negative
3	TX_AGC_EN_OVERRIDE	When set (1), this bit allows updates to the TX AGC PDM to occur as soon as the write to the TX AGC PDM register occurs. 1: enabled 0: disabled (default)

REVMOD_PDM_CTL (Continued)

Bits	Name	Description
2	TX_AGC_PDM_OUT_EN	This bit enables the PDM output of the TX AGC PDM. The Taz3 ASIC PDM output pin goes tri-state if the PDM output is disabled. 1: enabled 0: disabled (default)
1	PDM1_OUT_EN	This bit enables the PDM output of PDM 1. The Taz3 ASIC PDM output pin goes tri-state if the PDM output is disabled. 1: enabled 0: disabled (default)
0	PDM0_OUT_EN	This bit enables the PDM output of PDM 0. The Taz3 ASIC PDM output pin goes tri-state if the PDM output is disabled. 1: enabled 0: disabled (default)

0x0010 REVMOD_TX_AGC_ODRIVE_DLY

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
15:0	DATA	The REVMOD_TX_ODRIVE_DLY register is a 16-bit register that specifies the duration, in chipx16 clocks, that the TX AGC PDM overdrives its output such that its RC filtered voltage reaches a new value with a quicker response time. The duration of the overdrive interval is programmed to approximately equal to the RC time constant of the RC filter in the TX AGC PDM.

0x0014 REVMOD_TX_AGC_ADJ

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
15:0	DATA	The REVMOD_TX_AGC_ADJ register is the value adjustment register for the TX AGC PDM. The 2's complement, 10-bit value in this register controls the PDM output of the TX AGC adjust PDM. Note that the 10-bit 2's complement value must be MSB aligned before writing it to this register. Bits 5:0 are automatically set to zero. ■ If the TX_AGC_PDM_OUT_POL in the REVMOD_PDM_CONTROL register is clear (0), a maximum positive 2's complement value produces a PDM output with a maximum density of one bit. A maximum negative 2's complement value produces a PDM output with a minimum density of one bit. ■ If the PDM output polarity is set (1), the inverse occurs. ■ If the TX_AGC_EN_OVERRIDE bit is set in the PDM control register (bit 3), then any writes to the TX AGC Adjust register will have an immediate effect on the TX AGC PDM. If the TX_AGC_EN_OVERRIDE bit is clear, then any writes to the TX AGC Adjust register will take effect at any point during the 1st and 3rd quarter slots of each slot time.

0x0018 REVMOD_PDM0

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
15:0	DATA	The REVMOD_PDM0 register is the PDM 0 value register. The 2's complement 10-bit value in this register controls the PDM output of PDM 0. Note that the 10-bit 2's complement value must be MSB-aligned before writing it to this register. Bits 5:0 are automatically be set to zero. See the description for the REVMOD_TX_AGC_ADJ register for more programming information.

0x001C REVMOD_PDM1

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
15:0	DATA	The REVMOD_PDM1 register is the PDM 1 value register. The 2's complement 10-bit value in this register controls the PDM output of PDM 1. Note that the 10-bit 2's complement value must be MSB-aligned before writing it to this register. Bits 5:0 are automatically be set to zero. See the description for the REVMOD_TX_AGC_ADJ register for more programming information.

0x0020 REVMOD_RF_PN_ROLL_TIME

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
8:0	DATA	The REVMOD_RF_PN_ROLL_TIME register holds a 9-bit 2's complement value that controls the skewing of the RF system time PN roll boundary, with respect to the PN roll boundary of the REVMOD system time. The 9-bit value allows retards of up to 256 chip clocks (negative value) and advances up to 255 chip clocks (positive value), with respect to the modulator's system time 26.67 msec rollover boundary. Adjustments to this value have immediate effect on the skewed RF system time, synchronous to the next chipx16 clock.

0x0024 REVMOD_PA_RANGE

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
1:0	DATA	The REVMOD_PA_RANGE register is the PA range control register. This register controls the output status of the <pa_r0> and <pa_r1> PA control bits. ■ Setting (1) one of the PA range control bits forces the respective PA range control signal to go active. Sense of the output pin is determined by bit 1 in the PA RANGE Polarity register. ■ Clearing (0) one of the PA range control bits forces the respective PA range control signal to go inactive. Sense of the output pin is determined by bit 0 in the PA RANGE Polarity register.

0x0028 REVMOD_RATE_INDEX

Type: Read
Clock: DSP_CLK

Bits	Name	Description
3	ACCESS_PILOT	The access_pilot register is the access pilot request register. This register is a read-only register that indicates the access pilot of the next frame. The value in the access pilot register bit is updated exactly one-half slot before the programmed frame offset slot of the next data packet, and remains valid during the duration of transmission of the data packet (one frame times or 26.7 msec). The DSP must read the reverse access pilot register bit sometime after the one-half slot before the frame offset slot and before the actual frame offset slot.
2:0	RRI_INDEX	The RRI_index register is the reverse link rate index register. This register is a read-only register that indicates the reverse link rate of the data being sent over the traffic channel of the next frame. The value in the rate index register is updated exactly one-half slot before the programmed frame offset slot of the next data packet, and remains valid during the duration of transmission of the data packet (one frame times or 26.7 msec). The DSP must read the reverse link rate index register sometime after the one-half slot before the frame offset slot and before the actual frame offset slot. Table 17-5 lists the RRI values and the programmed rates.

Table 17-5 RRI rate index

RRI value	Rate
0	NULL
1	9.6 kbps
2	19.2 kbps
3	38.4 kbps
4	76.8 kbps
5	153.6 kbps
6	Invalid
7	Invalid

0x002C REVMOD_PA_RANGE_POLARITY

Type: Read/Write
Clock: DSP_CLK

The REVMOD_PA_RANGE_POLARITY register contains control bits that select the output polarity of the <pa_r0> and <pa_r1> pins. The matrix in Table 17-6 defines the state of the output pin for each combination of PA range control bit and polarity setting.

Bits	Name	Description
1	pa_r1	Controls the output polarity of <pa_r1>
0	pa_r0	Controls the output polarity of <pa_r0>

Table 17-6 <pa_r0> and <pa_r1> control matrix

PA range state	Polarity setting	<pa_r0/1> pin state
0	0	Hi-Z
0	1	0
1	0	0
1	1	Hi-Z

0x0030 REVMOD_I_GAIN_PILOT

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
11:0	DATA	The REVMOD_I_GAIN_PILOT register is the Pilot/RRI channel gain register. This register is a 12-bit, unsigned gain value. The gain only goes into effect on half-slot or slot boundaries.

0x0034 REVMOD_I_GAIN_ACK

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
11:0	DATA	The REVMOD_I_GAIN_ACK register is the ACK channel gain register. This register is a 12-bit, unsigned gain value. The gain only goes into effect on half-slot or slot boundaries.

0x0038 REVMOD_ACK_CTL

Type: Read/Write
Clock: DSP_CLK

The REVMOD_ACK_CTL register is the ACK channel control register.

Bits	Name	Description
2	DSP_CRC_VAL	CRC value from DSP
0	ACK_EN	When set (1), ACK is enabled When clear (0), ACK is disabled

0x003C REVMOD_PA_POWERUP_CTL

Type: Read/Write
Clock: DSP_CLK

The REVMOD_PA_POWERUP_CTL register is the PA powerup control register.

Bits	Name	Description
11	PA_POWERUP_EN	PA powerup is programmed by the DSP
10:0	PA_WARMUP_TIME	PA warmup time in chip resolution

0x0040 REVMOD_PA_RANGE_DELAY

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
6:0	DATA	The REVMOD_PA_RANGE_DELAY register controls the delay on PA range switch, relative to the half-slot boundary, in chip resolution.

0x0044 REVMOD_Q_GAIN_TRAFFIC

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
11:0	DATA	The REVMOD_Q_GAIN_TRAFFIC register is the traffic channel gain register. This register is a 12-bit, unsigned gain value. The gain only goes into effect on half-slot or slot boundaries.

0x0048 REVMOD_Q_GAIN_DRC

Type: Read/Write
Clock: DSP_CLK

Bits	Name	Description
11:0	DATA	The REVMOD_Q_GAIN_DRC register is the drc channel gain register. This register is a 12-bit, unsigned gain value. The gain only goes into effect on half-slot or slot boundaries .

17.2.4.2 Tx AGC power control firmware registers

0x004C REV_LINKPWR_CTRL_WR

Type: Write/Command
Clock: DSP_CLK

This register is used to send up/down commands to the Tx AGC to control the mobile TX power.

Bits	Name	Description
15	RPC_BIT	Up-down strobe to the Tx AGC.

18 Sleep Controller

18.1 ARM registers

Addresses in the sections that follow are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: SLP_CTL
- Number of word addresses: 51
- Byte Address:0x0500 - 0x06FC (total area)
- OFFSET= CHIP_BASE+0x0500
- CHIP_BASE = 0x80000000
- chipaddr range = 13:2

0x0000 SLEEP_CTL

Type: Write
Clock: MICRO_CLK

All bits of this register are cleared on RESOUT_N.

Bits	Name	Description
0	SLEEP_ARM	Setting (1) this bit enables the ASIC to go into the sleep mode on the next sleep trigger. Reset state: 0

0x0004 SLEEP_CMD

Type: Write/Command
Clock: MICRO_CLK

Bits	Name	Description
0	SLEEP_NOW	This bit forces the 1X sleep controller start the sleep procedure. A SLEEP_CX8_CLK wide pulse is generated when micro programs it (1).

0x0008 SLEEP_CX8_EXPIRE

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
15:0	DATA	Specifies the total number of slpcx8 clock cycles within the total sleep time. This value must be greater than twice the number of slpcx8 clock cycles in one sleep clock period.

0x000C SLEEP_CX8_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
15:0	DATA	Returns the current slpcx8 count. This register is intended to be read during sleep while the slpcx8 counter is inactive. When read during sleep (either the SLEEPING or WARMUP state of SLEEP_STATUS) it returns the number of slpcx8 cycles counted before the rising edge of sleep clock was detected and the sleep counter started.

0x0010 SLEEP_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Returns the current sleep counter value. Note: the sleep counter value is sampled using the micro clk. Multiple reads are required to determine the value (assuming micro clk is much faster than sleep clock)

0x0014 SLEEP_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
3	WARMUP	When set (1), indicates counting through the warmup period
2	SLEEPING	When set (1), indicates counting through the sleep period
1	CX8_RUNNING_OUT	When set (1), indicates the slpcx8 counter is running at the end of sleep
0	CX8_RUNNING_IN	When set (1), indicates the slpcx8 counter is running at the beginning of sleep.

0x0018 HDR_SLEEP_CTL

Type: Write
Clock: MICRO_CLK

All bits of this register are cleared on RESOUT_N.

Bits	Name	Description
0	SLEEP_ARM	Setting (1) this bit enables the ASIC to go into the sleep mode on the next sleep trigger. Reset state: 0

0x001C HDR_SLEEP_CMD

Type: Write/Command
Clock: MICRO_CLK

Bits	Name	Description
0	SLEEP_NOW	This bit forces the HDR sleep controller start the sleep procedure. A SLEEP_CX8_CLK wide pulse is generated when micro programs it (1).

0x0020 HDR_SLEEP_CX8_EXPIRE

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
15:0	DATA	Specifies the total number of slpcx8 clock cycles within the total sleep time. This value must be greater than twice the number of slpcx8 clock cycles in one sleep clock period.

0x0024 HDR_SLEEP_CX8_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
15:0	DATA	Returns the current slpcx8 count. This register is intended to be read during sleep while the slpcx8 counter is inactive. When read during sleep (either the SLEEPING or WARMUP state of SLEEP_STATUS) it returns the number of slpcx8 cycles counted before the rising edge of sleep clock was detected and the sleep counter started.

0x0028 HDR_SLEEP_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Returns the current sleep counter value. Note: the sleep counter value is sampled using the micro clk. Multiple reads are required to determine the value (assuming micro clk is much faster than sleep clock)

0x002C HDR_SLEEP_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
3	WARMUP	When set (1), indicates counting through the warmup period
2	SLEEPING	When set (1), indicates counting through the sleep period
1	CX8_RUNNING_OUT	When set (1), indicates the slpcx8 counter is running at the end of sleep
0	CX8_RUNNING_IN	When set (1), indicates the slpcx8 counter is running at the beginning of sleep.

0x0030 SLEEP_XTAL_TIMER_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
15:0	DATA	Returns current slp_xtal_timer counter value (16-bit up counter) that runs off the sleep xtal clock regime. The counter continuously counts the number of clock cycles equal to the value programmed into this register and generate the SLEEP_XTAL_TIMER_INT. The counter is reset once the interrupt is generated. Note: the sleep counter value is sampled using the micro clk. Multiple reads are required to determine the value (assuming micro clk is much faster than sleep clock)

0x0034 SLEEP_XTAL_TIMETICK_COUNT

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Returns current slp_xtal_timetick counter value (32-bit up counter) that runs off the sleep xtal clock regime. The counter continuously counts the number of clock cycles equal to the value programmed into this register and generate the TIMETICK_INT. The timer roll over at 2^32 1X sleep clock cycles and is not reset once the interrupt is generated. Note: the sleep counter value is sampled using the micro clk. Multiple reads are required to determine the value (assuming micro clk is much faster than sleep clock)

0x0038 SLEEP_XTAL_FREQ_ERR

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
26:0	DATA	The SLEEP_XTAL_FREQ_ERR register returns the number of slpfast clock cycles that were counted in SLEEP_XTAL_CNT clock cycles. The value, a 27-bit number, is stable after the sleep_fee_interrupt and until the SLEEP_XTAL_CNT register is written to again. Once the SLEEP_XTAL_CNT register is written to, the frequency estimation is restated.

0x003C WDOG_RESET

Type: Write/Command
Clock: MICRO_CLK

This register resets the watch dog counter. In normal operation, micro should periodically write this command register to reset watch dog. This command register also disables the autokicker.

Bits	Name	Description
0	WATCH_DOG	This bit generates the WDOG_STB during the non-sleep mode. A pulse is generated on WDOG_STB when this bit is set (1)

0x0040 AUTOKICK_START

Type: Write/Command
Clock: MICRO_CLK

This register starts the autokicker. Micro should write this command register before it goes to sleep. The autokicker will periodically resets the watch dog counter during sleep. The autokicker is reset by programming WDOG_RESET to (1) or the wakeup_int pulse generated by each sleep core, which is controlled by WDOG_WAKEUP_CTL.

Bits	Name	Description
0	AUTO_KICK_START	This bit is used to start the auto-kicker circuit in the watchdog timer. To enable the watchdog timer auto-kicker, set (1) this bit.

0x0044 WDOG_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
15:2	WDOG_COUNT	Counter value [13:0] of watch dog counter. Note: the sleep counter value is sampled using the micro clk. Multiple reads are required to determine the value (assuming micro clk is much faster than sleep clock)
1	AUTOKICK_EN	This bit indicates whether the autokicker is on. 1: on 0: off
0	RESET_STATUS	1: watchdog reset (the source of the last system reset occurs due to the internal watchdog timer) 0: resin_n reset (the source of the last system reset is RESIN_N asserting.

0x0048 WDOG_WAKEUP_CTL

Type: Write/Read
Clock: MICRO_CLK

This register control the wakeup_int of each sleep core to disable the autokicker.

Bits	Name	Description
2	WAKEUP_1X_DIS	When set this bit (0), wakeup_int_1x signal will disable the autokicker. When set this bit (1), autokicker is not disable by wakeup_int_1x.
1	WAKEUP_HDR_DIS	When set this bit (0), wakeup_int_hdr signal will disable the autokicker. When set this bit (1), autokicker is not disable by wakeup_int_hdr.
0	WAKEUP_GSM_DIS	When set this bit (0), wakeup_int_gsm signal will disable the autokicker. When set this bit (1), autokicker is not disable by wakeup_int_gsm.

0x004C GSM_SLEEP_CTL

Type: Read/write
Clock: MICRO_CLK

The SLEEP_CTL register controls most of the options available in the sleep controller.

Following control bits are removed because of the different TimeTick2 hardware:

- TIMER_1_AUTOKICK_ENA
- TIMER_0_AUTOKICK_ENA
- TIMER_1_TRIGGER_ARM

Following watch dog related control bits are moved to WDOG_CTL register.

- SLEEP_CTL_AUTOKICK_ENA
- WATCH_DOG

GSM_SLEEP_CTL

Bits	Name	Description
6:4	SLEEP_N[3:1]	These bits allow the microprocessor to directly control the sleep[3:0]_n pins directly. These bits are effective only when the SLEEP_N_ENA bits in the SLEEP_WU_1_CONFIG and SLEEP_WU_0_CONFIG registers are cleared (0). For example, if the ACT_TIME_0_SLEEP_N_ENA bit of the SLEEP_WU_1_CONFIG and SLEEP_WU_0_CONFIG registers are cleared (0), microprocessor has control over the sleep_n_wr[1] pin through sleep_n_wr[0].
3	RESERVED_BIT3	
2	PICH_MODE	This bit indicates whether the Simba ASIC is in PICH mode. Set (1) this bit for the PICH mode. Clear (0) this bit for the non-PICH mode (when the UE is paged and it must remain awake, the microprocessor clears this bit.
1:0	RESERVED_BITS1_0	

0x0050 GSM_SLEEP_WU_0_TIME_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
9:0	DATA	The GSM_SLEEP_WU_0_TIME_CONFIG register defines the warm up time 0 period, which is the period after the sleep state to the start of the residual tick count. The value for this register is specified in sleep controller clock cycles. The minimum value for this register is 2.

0x0054 GSM_SLEEP_PHASE_CORRECTION_CNT

Type: Read/write
Clock: MICRO_CLK

Bits	Name	Description
12:0	DATA	During a write, the GSM_SLEEP_PHASE_CORRECTION_CNT register defines the terminal count for the CHIPx8 counter. This provides a finer resolution on the sleep period given by the GSM_SLEEP_INTERVAL register. It is specified in chipx8 clock cycles. The minimum value for the GSM_SLEEP_PHASE_CORRECTION_CNT register is 2. The GSM_SLEEP_PHASE_CORRECTION_CNT register is read during the sleep state to determine the phase error between the slpchipx8 clock and the sleep crystal clock. Because the sleep clock counter starts at the rising edge of the sleep clock signal, there is a time difference between when the go_to_sleep signal is generated from the system timer and when the time the sleep clock counter begins counting. This register holds the number of slpchipx8 cycles that have elapsed during this time difference. The value stays in this register during the sleep state. Once the sleep state is over, the value stored in this register does not have any significance to the microprocessor.

0x0058 GSM_SLEEP_WU_0_CONFIG

Type: Read/write
Clock: MICRO_CLK

The GSM_SLEEP_WU_0_CONFIG register configures the generation of the warm up time interrupt and the sleep_n control signal.

Bits	Name	Description
2:1	ACT_TIME_INT_ENA	Setting (1) the appropriate bit in this field enables the sleep controller to generate an interrupt upon the number of CHIPX8 counter counts, as specified by the SLEEP_WU_ACT_TIME_x_CONFIG bits (where X = 0 to 1).
0	ACT_TIME_0_SLEEP_N_ENA	Setting (1) this bit enables the sleep controller to set (1) the sleep1_n pin upon the number of CHIPX8 counter counts, as specified by the SLEEP_WU_O_ACT_ TIME_0_CONFIG bit. When this bit is set (1), the sleep1_n pin is cleared (0) upon entering sleep state. Note: the sleep1_n pin is also controlled by the ACT_TIME_0_SLEEP_N_ENA[0] bit of the SLEEP_WU_1_CONFIG register. Do not set (1) both this bit the and the ACT_TIME_0_SLEEP_N_ENA[0] at any given time. When both bits are clear (0), the state of the sleep1_n pin is specified by the SLEEP_N_WR[1] bit of the SLEEP_CTL register.

0x005C GSM_SLEEP_WU_1_CONFIG

Type: Read/write
Clock: MICRO_CLK

The GSM_SLEEP_WU_1_CONFIG register configures the generation of the warm up time 1 interrupt and the sleep_n control signal.

Bits	Name	Description
8:3	ACT_TIME_INT_ENA	Setting (1) the appropriate bit in this field enables the sleep controller to generate an interrupt upon the number of CHIPX8 counter counts, as specified by the GSM_SLEEP_WU_1_ACT_TIME_X_CONFIG bit (where X = 0 to 5).
2:0	ACT_TIME_SLEEP_N_ENA	Setting (1) the appropriate bit in this field enables the sleep controller to set (1) the sleep[x+1]_n] pin upon the number of CHIPx8 counter counts, as specified by the GSM_SLEEP_WU_1_ACT_TIME_X_CONFIG bit (where X = 0 to 2). When a given bit in this field is set (1), the sleep[x+1]_n pin is cleared (0) upon entering sleep state. Note: the sleep1_n pin is also controlled by the ACT_TIME_0_SLEEP_N_ENA bit of the GSM_SLEEP_WU_0_CONFIG register. Do not set both this bit and the ACT_TIME_0_SLEEP_N_ENA bit at any given time. When both bits are clear (0), the state of the sleep1_n pin is specified by the SLEEP_N_WR[1] bit of the GSM_SLEEP_CTL register.

0x0060 GSM_SLEEP_INT_STATUS_RD

Type: Read
Clock: MICRO_CLK

The GSM_SLEEP_INT_STATUS_RD register is updated by sleep controller whenever an interrupt is issued.

Bits	Name	Description
11	ON_LINE_END_INT	This bit is set (1) upon the end of the sample RAM state.
10	ON_LINE_START_INT	This bit is set (1) upon the end of warm-up time 1. This interrupt indicates the start of the sample RAM start. The system timer value is also reloaded at this time.
9	WU_1_ACT_TIME_5_INT	This bit is set (1) when the CHIPx8 counter counts to the WU_1_ACT_TIME_5_CONFIG value during the warmup time 1 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[8] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
8	WU_1_ACT_TIME_4_INT	This bit is set (1) when the CHIPx8 counter counts to the SLEEP_WU_1_ACT_TIME_4_CONFIG value during the warmup time 1 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[7] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
7	WU_1_ACT_TIME_3_INT	This bit is set (1) when the CHIPx8 counter counts to the SLEEP_WU_1_ACT_TIME_3_CONFIG value during the warmup time 1 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[6] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
6	WU_1_ACT_TIME_2_INT	This bit is set (1) when the CHIPX8 counter counts to the SLEEP_WU_1_ACT_TIME_2_CONFIG value during the warmup time 1 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[5] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
5	WU_1_ACT_TIME_1_INT	This bit is set (1) when the CHIPX8 counter counts to the SLEEP_WU_1_ACT_TIME_1_CONFIG value during the warmup time 1 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[4] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
4	WU_1_ACT_TIME_0_INT	This bit is set (1) when the CHIPx8 counter counts to the SLEEP_WU_1_ACT_TIME_0_CONFIG value during the warmup time 1 state.This interrupt is disabled if the ACTION_TIME_INT_ENA[3] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
3	WU_0_ACT_TIME_1_INT	This bit is set (1) when the sleep counter counts to the SLEEP_WU_0_ACT_TIME_1_CONFIG value during the warmup time 0 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[1] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
2	WU_0_ACT_TIME_0_INT	This bit is set (1) when the sleep counter counts to the SLEEP_WU_0_ACT_TIME_0_CONFIG value during the warmup time 0 state. This interrupt is disabled if the ACTION_TIME_INT_ENA[0] bit of the SLEEP_WU_1_CONFIG register is cleared (0).
1	MICRO_WAKEUP_INT	This bit is set (1) upon the end of the sleep state. This interrupt wakes up the microprocessor after the sleep state to start the wake-up sequence.
0	SLEEP_START_INT	This register bit is set (1) upon the start of the sleep state. This interrupt signals the microprocessor to turn off all the RF components, except for the TCXO and the PLL's.

0x0064 GSM_SLEEP_INT_CLEAR_WR

Type: Write
Clock: MICRO_CLK

The GSM_SLEEP_INT_CLEAR_WR register is used to clear the interrupt and the contents of the GSM_SLEEP_INT_STATUS_RD register.

Bits	Name	Description
11:0	SLEEP_INT_CLEAR	By setting (1) the appropriate bit of this field, the interrupt for the corresponding bit in the SLEEP_INT_STATUS_RD register is cleared (0). The combined interrupt signal, sleep_int, can be cleared (0) only if all of the bits in the SLEEP_INT_STATUS_RD register are cleared (0).

0x0068 GSM_SLEEP_INT_ENABLE

Type: Read/write
Clock: MICRO_CLK

The GSM_SLEEP_INT_ENABLE register controls the interrupt source for the sleep_int signal. To disable one or more interrupt source for the sleep_int signal, the appropriate bits of the GSM_SLEEP_INT_MASK_N[x] field are cleared (0).

Bits	Name	Description
11:0	DATA	Set (1) the appropriate bit in this field to enable the corresponding interrupt in the SLEEP_INT_STATUS_RD register. Clear (0) the appropriate bit in this field to disable the corresponding interrupt in the SLEEP_INT_STATUS_RD register.

0x006C GSM_SLEEP_ON_LINE_TIME_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The GSM_SLEEP_ON_LINE_TIME_CONFIG register defines the amount of time for on-line processing, which includes sample RAM storage, on line search, and demodulation of PICH bits. It is specified in slpchipx8 clock cycles. The minimum value for the SLEEP_ON_LINE_ TIME_CONFIG register is 0x00002

0x0070 GSM_SLEEP_WU_1_TIME_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_TIME_CONFIG register defines the warmup time 1 period, which is the period after the end of residual tick count up to the start of the sample RAM write operation. The value for this register is specified in slpchipx8 clock cycles. The minimum value for this register is 2.

0x0074 GSM_SLEEP_WU_0_ACT_TIME_0_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
9:0	DATA	The SLEEP_WU_0_ACT_TIME_0_CONFIG register defines the comparison value for the sleep clock counter during warmup time 0. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor and set (1) the sleep_n[1] signal, depending on the values of the ACT_TIME_INT_ENA[0] and ACT_TIME_SLEEP_N_ENA[0] bits of the SLEEP_WU_0_CONFIG register. The SLEEP_WU_0_ACT_TIME_0_CONFIG register is specified in sleep controller clock cycles. The minimum value for this register is 2.

0x0078 GSM_SLEEP_WU_0_ACT_TIME_1_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
9:0	DATA	The SLEEP_WU_0_ACT_TIME_1_CONFIG register defines the comparison value for the sleep clock counter during warmup time 0. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor, depending on the value of the ACT_TIME_INT_ENA[1] bit of the SLEEP_WU_0_CONFIG register. The SLEEP_WU_0_ACT_TIME_1_CONFIG register is specified in sleep controller clock cycles. The minimum value for this register is 2.

0x007C GSM_SLEEP_WU_1_ACT_TIME_0_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_0_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor and set (1) the sleep_n[1] signal to the microprocessor, depending on the value of the ACT_TIME_INT_ENA[0] bit of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_0_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2.

0x0080 GSM_SLEEP_WU_1_ACT_TIME_1_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_1_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor and set (1) the sleep_n[2] signal to the microprocessor, depending on the value of the ACT_TIME_INT_ENA[1] and ACT_TIME_SLEEP_N_ENA[1] bits of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_1_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2.

0x0084 GSM_SLEEP_WU_1_ACT_TIME_2_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_2_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor and set (1) the sleep_n[3] signal to the microprocessor, depending on the setting of the ACT_TIME_INT_ENA[2] and ACT_TIME_SLEEP_N_ENA[2] bits of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_2_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2.

0x0088 GSM_SLEEP_WU_1_ACT_TIME_3_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_3_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor based on the value of the ACTION_TIME_INT_ENA[3] bit of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_3_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2.

0x008C GSM_SLEEP_WU_1_ACT_TIME_4_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_4_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor, depending on the value of the ACTION_TIME_INT_ENA[4] bit of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_4_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2.

0x0090 GSM_SLEEP_WU_1_ACT_TIME_5_CONFIG

Type: Write
Clock: MICRO_CLK

Bits	Name	Description
18:0	DATA	The SLEEP_WU_1_ACT_TIME_5_CONFIG register defines the comparison value for the SLEEP CHIPx8 counter during warmup time 1. When the counter reaches this value, the sleep controller may assert an interrupt to the microprocessor, depending on the setting of the ACTION_TIME_INT_ENA[5] bit of the SLEEP_WU_1_CONFIG register. The SLEEP_WU_1_ACT_TIME_5_CONFIG register is specified in slpchipx8 cycles. The minimum value for this register is 2. The SLEEP_TIMER_1_CNT register controls a 16-bit up counter that runs off the sleep controller clock regime (between 30kHz and 60kHz). The counter is disabled upon system power-up and it is enabled if a value other than 0 is written to this register. Once enabled, the counter continuously counts the number of clock cycles that is equal to the value programmed into this register and then generates the sleep_timer_int[1] signal. To disable this counter, clear (0) this register. The current counter value can be read from this same address.

0x0094 SLEEP_WAKEUP_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) SLEEP_WAKEUP data is synchronizing to 1X_SLEEP_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x0098 SLEEP_WAKEUP

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Specifies the sleep count value at which the wakeup interrupt is generated: between SLEEP_WAKEUP + 1 and SLEEP_WAKEUP + 2 sleep clock cycles after sleep is triggered. The wakeup interrupt is generated as the warmup period is entered.

0x009C SLEEP_EXPIRE_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) SLEEP_EXPIRE data is synchronizing to 1X_SLEEP_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x00A0 SLEEP_EXPIRE

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Specifies the sleep count value at which sleep ends. When this count is reached, the slpcx8 counter is re-enabled and counts to the SLEEP_CX8_EXPIRE value. The total sleep time is: T_{total} = (SLEEP_EXPIRE + 1) * SLEEP_CLK_PERIOD + (SLEEP_CX8_EXPIRE + 2) * SLPCX8_CLK_PERIOD

0x00A4 HDR_SLEEP_WAKEUP_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) HDR_SLEEP_WAKEUP data is synchronizing to HDR_SLEEP_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x00A8 HDR_SLEEP_WAKEUP

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Specifies the sleep count value at which the wakeup interrupt is generated: between SLEEP_WAKEUP + 1 and SLEEP_WAKEUP + 2 sleep clock cycles after sleep is triggered. The wakeup interrupt is generated as the warmup period is entered.

0x00AC HDR_SLEEP_EXPIRE_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) HDR_SLEEP_EXPIRE data is synchronizing to HDR_SLEEP_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x00B0 HDR_SLEEP_EXPIRE

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	Specifies the sleep count value at which sleep ends. When this count is reached, the slpcx8 counter is re-enabled and counts to the SLEEP_CX8_EXPIRE value. The total sleep time is: T_{total} = (SLEEP_EXPIRE + 1) * SLEEP_CLK_PERIOD + (SLEEP_CX8_EXPIRE + 2) * SLPCX8_CLK_PERIOD

0x00B4 SLEEP_XTAL_TIMER_ENABLE

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
1	SYNC_STATUS	When set (1) sleep xtal timer enable is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).
0	ENABLE	This register resets/enables the timer counter. Clear (0) this bit to enable timer. 0: reset 1: enable

0x00B8 SLEEP_XTAL_TIMER_MATCH_VAL

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
16	SYNC_STATUS	When set (1) sleep_xtal_timer match data is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).
15:0	DATA	The SLEEP_XTAL_TIMER_MATCH_VAL register defines the terminal value of the slp_xtal_timer counter (16-bit up counter) that runs off the sleep xtal clock regime. The counter continuously counts the number of clock cycles equal to the value programmed into this register and generate the SLEEP_XTAL_TIMER_INT. The counter is reset when the interrupt is generated. To disable and reset this counter, write (0) to the SLEEP_XTAL_TIMER_ENABLE register. (The counter actually counts for a value 1 higher than the terminal count. For example, programming 1 the interrupt asserted every 2 clocks.)

0x00BC SLEEP_XTAL_TIMETICK_MATCH_VAL_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) SLEEP_XTAL_TIMETICK_MATCH_VAL data is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x00C0 SLEEP_XTAL_TIMETICK_MATCH_VAL

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
31:0	DATA	The SLEEP_XTAL_TIMETICK_MATCH_VAL register defines the terminal value of the slp_xtal_timetick counter (32-bit up counter) that runs off the sleep xtal clock regime. The counter continuously counts the number of clock cycles equal to the value programmed into this register and generate the SLEEP_XTAL_TIMETICK_INT. The timer roll over at 2^31 1X sleep clock cycles and is not reset once the interrupt is generated. Timer is a free-running counter and can be reset only by hardware.

0x00C4 SLEEP_XTAL_COUNT

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
16	SYNC_STATUS	When set (1) sleep xtal count data is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).
15:0	DATA	The SLEEP_XTAL_CNT register is the number of sleep crystal clock pulses to count for estimation of the sleep crystal frequency.

0x00C8 WDOG_EXPIRED_WIDTH

Type: Read/Write
Clock: MICRO_CLK

This register defines the width of wdog_expired pulse in the number of sleep_xtal_clk. The default value is 0x0C7C = 3196.

Bits	Name	Description
14	SYNC_STATUS	When set (1) wdog expired width data is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).
13:0	DATA	Wdog counter terminal value to de-assert wdog_expired pulse. Wdog counter is14-bits.

0x00CC WDOG_TEST_LOAD_STATUS

Type: Read
Clock: MICRO_CLK

Bits	Name	Description
0	SYNC_STATUS	When set (1) wdog test load is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).

0x00D0 WDOG_TEST_LOAD

Type: Write/Command
Clock: MICRO_CLK

Bits	Name	Description
0	LOAD	Write to this register (1) loads the WDOG_TEST register value into wdog counter.

0x00D4 WDOG_TEST

Type: Read/Write
Clock: MICRO_CLK

Bits	Name	Description
14	SYNC_STATUS	When set (1) wdog test is synchronizing to SLEEP_XTAL_CLK. Data value is not guaranteed until sync_status bit is clear (0).
13:0	WDOG_TEST_CNT	watchdog counter test load value [13:0]. Wdog counter is14-bits.

0x00D8 GSM_SLEEP_INTERVAL

Type: Read/write
Clock: MICRO_CLK

Bits	Name	Description
22	SYNC_STATUS	When set (1) GSM sleep interval data is synchronizing to GSM_SLEEP_CLK. Data value is not guaranteed until sync_status bit is clear (0).
21:0	DATA	During a write, the GSM_SLEEP_INTERVAL register defines the terminal count of the sleep counter. It is specified in sleep controller clock cycles, where the sleep controller clock runs between 30kHz to 60kHz. This value can be updated even during the sleep state when the sleep counter is operating. Care must be taken when updating the value during the sleep state. Read this register first to determine the current count. Then update the value to be at least 5 larger than the read value. The minimum value for the GSM_SLEEP_INTERVAL register is 1. When read, this register returns the current count of the sleep clock counter.

19 USB

19.1 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: USB
- Memory Space = 0x80002000 - 0x80003FFC
- USB_BASE = 0x80002000
- usbaddr range = 12:2

19.1.1 Core level registers

0x000 USB_HARDWARE_MODE

Type: Read/Write
Clock: MCLK

System register. In certain applications the user can force the CORE to work directly as a host or function. This register is reset by the system reset or a write to the ResetControl Register with the **ResetControlLogic** bit set (1).

USB_HARDWARE_MODE

Bits	Name	Description
31:24	FUNCREV	SW: R, HW: R FunctionRevision. Indicates the revision of the Function Controller code and features.
23:16	HSTREV	SW: R, HW: R HostRevision. Indicates the revision of the Host Controller code and features.

USB_HARDWARE_MODE (Continued)

Bits	Name	Description
15	TSTMDE	SW: RW, HW: R TestMode. Setting this bit will put the Host and Function controllers into test mode. This is used for test pattern generation and should not be used in regular operation.
14	ANASDBEN	OtgAnalogSignalShortDebounceEnable 0: Enable the OTG VBUS (and ID signal long debounce logic). 1: Disable the OTG VBUS (and ID signal long debounce logic).
13:8	RESERVED_BITS13_8	
7:6	OTGXCVR	SW: RW, HW: R OtgTransceiverProperties. To suit various transceiver configurations Transmit / Receive 00b Diff (TxDp, TxDm) / Diff (RxDp, RxDm,RxD) 10b SE (Dat, SE0) / Diff (RxDp, RxDm,RxD) 01b Diff (TxDp, TxDm) / SE (Dat, SE0) 11b SE (Dat, SE0) / SE (Dat, SE0)
5:4	HOSTXCVR	SW: RW, HW: R HostTransceiverProperties. To suit various transceiver configurations. All Host transceivers must share the same properties. Transmit / Receive 00b Diff (TxDp, TxDm) / Diff (RxDp, RxDm,RxD) 10b SE (Dat, SE0) / Diff (RxDp, RxDm,RxD) 01b Diff (TxDp, TxDm) / SE (Dat, SE0) 11b SE (Dat, SE0) / SE (Dat, SE0)
3	RESERVED_BIT3	
2	BEMDE	SW: RW, HW: R BigEndianMode. When set (1) all accesses to the data memory will be in big endian.

USB_HARDWARE_MODE (Continued)

Bits	Name	Description
1:0	CRECFG	SW: RW, HW: R CoreConfiguration **Host/ Funct Mode** 00 Hardware HNP HNP is enabled to negotiate the working mode. 01 Host Only Operation Disable the HNP procedure. Force the OTG to work as a Host. 10 Function Host Operation Disable the HNP procedure. Force the OTG to work as a Function; however, other ports are in Host mode. 11 Software HNP Software-based HNP. The hardware HNP is disabled, and the software is responsible for doing HNP.

0x004 USB_CORE_INT_STATUS

Type: Read/Write
Clock: MCLK

Core interrupt source. This register provides status on various OTG-IP CORE level events that cause hardware interrupts. When an OTGIP

CORE level event occurs, hardware sets the corresponding bit in this register. When a bit becomes set, a hardware interrupt is generated if the interrupt is enabled in the COREInterruptStatusEnable Register.

USB_CORE_INT_STATUS

Bits	Name	Description
31:6	RESERVED_BIT31_6	
5	ASHNPINT	AsynchronousHnpInterrupt When asserted indicates that the asynchronous HNP event has triggered an interrupt. To CLEAR (0) this bit, the MainClockEnable bit must be SET (1) in the Clock Control Register.
4	ASFCINT	AsynchronousFunctionInterrupt When asserted indicates that the asynchronous Function Controller event has triggered an interrupt. To CLEAR this bit, the FunctionClockEnable bit must be SET in the Clock Control Register.
3	ASHCINT	AsynchronousHostInterrupt When asserted indicates that the asynchronous Host Controller event has triggered an interrupt. To CLEAR this bit, the HostClockEnable bit must be SET in the Clock Control Register.

USB_CORE_INT_STATUS (Continued)

Bits	Name	Description
2	HNPINT	SW: R, HW: S HnpInterrupt. When asserted indicates that the HNP has triggered an interrupt. To clear (0) this bit all source flags must be cleared or disabled.
1	FCINT	SW: R, HW: S FunctionInterrupt. When asserted indicates that the Function Controller has triggered an interrupt. To clear (0) this bit all source flags must be cleared or disabled.
0	HCINT	SW: R, HW: S HostInterrupt. When asserted indicates that the Host Controller has triggered an interrupt. To clear (0) this bit all source flags must be cleared or disabled.

0x008 USB_CORE_INT_ENA

Type: Read/Write
Clock: MCLK

Core interrupt mask. This register contains the corresponding enable bits of the COREInterruptStatus Register. An interrupt is only asserted if its enable bit is active. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

USB_CORE_INT_ENA

Bits	Name	Description
31:6	RESERVED_BITS31_6	
5	ASHNPINTEN	AsynchronousHnpInterruptEnable When SET it will allow asynchronous HNP event to trigger an interrupt.
4	ASFCINTEN	AsynchronousFunctionInterruptEnable When SET it will allow asynchronous Function Controller event to trigger an interrupt.
3	ASHCINTEN	AsynchronousHostInterruptEnable When SET it will allow asynchronous Host Controller event to trigger an interrupt.
2	HNPINTEN	SW: RW, HW: R HnpInterruptEnable. When SET it will allow HNP interrupts to be asserted.

USB_CORE_INT_ENA (Continued)

Bits	Name	Description
1	FCINTEN	SW: RW, HW: R FunctionInterruptEnable. When SET it will allow Function interrupts to be asserted.
0	HCINTEN	SW: RW, HW: R HostInterruptEnable. When SET it will allow Host interrupts to be asserted.

0x00C USB_CLOCK_CTL

Type: Read/Write
Clock: MCLK

Allows for disabling and enabling the clocks of the Host and Function. This register controls the clocks of the OTG-IP CORE to reduce power and turn off sections of logic when they are not needed. For example, when the OTG-IP CORE is working in **Host Only Operation**, the application can decide to switch off the clock input to the Function Controller core logic. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit. Upon reset these clocks are disabled and must be enabled before use.

Bits	Name	Description
31:3	RESERVED_BITS31_3	
2	FUNCCLK	SW: RW, HW: R FunctionClockEnable. When set (1) the clocks to the Function controller and SIE will be enabled.
1	HSTCLK	SW: RW, HW: R HostClockEnable. When set (1) the clocks to the Host controller and SIE will be enabled.
0	MAINCLK	MainClockEnable When SET it is assumed that the clock into the core is operational. This bit cannot be CLEARED if either the FunctionClockEnable or HostClockEnable is SET. The AsynchronousHnpInterrupt bit from the COREInterruptStatus Register will be cleared, if asserted, by setting this bit.

0x010 USB_RESET_CTL

Type: Read/Write
Clock: MCLK

Allows for independent resets of each of the blocks in the Host and Function. To give software better control over the resets, this register allows for independent resets of the major blocks of the core. Each bit is self-clearing after the reset is complete.

USB_RESET_CTL

Bits	Name	Description
31:16	RESERVED_BITS31_16	
15		Reserved.
14:6	RESERVED_BITS14_6	
5	RSTCTRL	SW: W, HW: Sc ResetControlLogic. When SET this will reset all of the registers in the CTRL interface to their default condition. This bit is self-clearing.
4	RSTFC	SW: W, HW: Sc ResetFunctionController. When SET this will reset all of the registers in the Function Controller interface to their default condition. This bit is self-clearing.
3	RSTFSIE	SW: W, HW: Sc ResetFunctionSIE. When SET this will reset all of the registers in the Function SIE interface to their default condition. This bit is self-clearing.
2	RSTRH	SW: W, HW: Sc ResetRootHub. When SET this will reset all of the registers in the Root Hub interface to their default condition. This bit is self-clearing.
1	RSTHSIE	SW: W, HW: Sc ResetHostSIE. When SET this will reset all of the registers in the Host SIE interface to their default condition. This bit is self-clearing.
0	RSTHC	SW: W, HW: Sc ResetHostController. When SET this will reset all of the registers in the Host Controller interface to their default condition. This bit is self-clearing.

0x014 USB_FRAME_INTERVAL

Type: Read/Write
Clock: MCLK

Sets the width of a USB Frame in full-speedbit times. The **FrameInterval** indicates the number of bit times between two consecutive SOFs. Software may carry out minor adjustment on the **FrameInterval** by writing a new value over the present one at each SOF. If the **ResetFrame** bit is not used the new frame time will be updated at the next SOF. This provides the programmability necessary for the Host Controller to synchronize with an external clocking resource and to adjust any unknown local clock offset. The **FrameIntervalPeriodic** is the number of bit times in a frame when periodic packets like Isochronous and Interrupt can be sent. Based on the Software application this value can be adjusted. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

USB_FRAME_INTERVAL

Bits	Name	Description
31:30	RESERVED_BIT31_30	
29:16	FRMINPER	SW: RW, HW: R FrameIntervalPeriododic. The value written here sets the amount of time in a frame that Periodic packets can be sent. This value is in the number of full-speed bit times
15	RSTFRM	SW: W, HW: Sc ResetFrame. When SET the FrameRemainingBitWidth Register will be cleared and be set to the value in the **FrameInterval**.
14	RESERVED_BIT14	
13:0	FRMINT	SW: RW, HW: R FrameInterval. The value written here sets the width of a frame. This value is in the number of full-speed bit times This specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999.

0x018 USB_FRAME_REMAINING_BIT_WIDTH

Type: Read/Write
Clock: MCLK

Contains the number of full-speed bit times remaining in the current frame and sets the relationship between *mClk* width and full speed bit time.

The FrameRemainingBitWidth Register is used to program the clock relationship between the *mClk* and the USB 12MHz clock (83.33ns). It also contains a 14-bit down counter showing the bit time remaining in the current Frame.

For example, if your *mClk* is 66MHz then the **MasterClockBitWidth** value should be 6 (0110b). The default value is the ratio for 48MHz. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

Bits	Name	Description
31:30	RESERVED_BIT31_30	
29:16	FRMREMN	SW: R, HW: W FrameRemaining. This number represents the number of full-speed bit times still remaining in the current frame. This counter is decremented at each bit time. When it reaches zero, loading the FrameInterval value specified in FrameRemaining at the next bit time boundary resets it. When entering the UsbOperational state, the Host Controller re-loads the content with the FrameInterval and uses the updated value from the next SOF.
15:0	RESERVED_BIT15_4	

0x01C USB_HNP_CTL_STATUS

Type: Read/Write
Clock: MCLK

Gives the current state of the HNP. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

USB_HNP_CTL_STATUS

Bits	Name	Description
31	RESERVED_BIT31	
30	HNPDAT	SW: R, HW: W HnpDataToggle. When asserted this bit indicates that either the DP or the DM line is asserted. SOFTWARE can use this line to detect Data line pulsing SRP.

USB_HNP_CTL_STATUS (Continued)

Bits	Name	Description
29	VBUSBSE	SW: R, HW: W VBusGreaterThanBSessionEnd When asserted, this bit indicates that the bus has been sampled greater than or equal to b_session_end (0.8V). When not asserted a B-DEVICE can consider the session ended.
28	VBUSABSV	SW: R, HW: W VBusABSessionValid When asserted this indicates that the bus has been sampled greater than or equal to 0.8V, however, between 0.8V and 2.0V. This is in the a_session_valid and the b_session_valid range.
27	VBUSGTAVV	SW: R, HW: W When asserted this indicates that the bus has been sampled greater than or equal to 4.4V. This indicates a_session_valid.
26	RESERVED_BIT26	
25	ARMTHNPE	SW: RW, HW: R ARemoteHnpEnabled. When the OTG-IP CORE is an A_MASTER this bit will allow a B device to request an HNP. This bit in the A-DEVICE is a copy of the b_mnp_en bit in the B-DEVICE. The A-DEVICE sets this bit at the same time as it sets the b_mnp_en bit in the B-DEVICE, and clears this bit when it determines that the B-DEVICE should have cleared the BHnpEnabled bit.
24	BHNPEN	SW: RW, HW: R BHnpEnabled. When the OTG-IP CORE is a B_SLAVE, software can set this bit after receiving permission from the A_MASTER to request an HNP. This bit in the B-DEVICE is set by the A-DEVICE via the Set Feature command to enable the B-DEVICE to perform HNP. The B-DEVICE clears this bit in response to a Clear Feature command, or after the B-DEVICE has assumed bus mastership.
23	RESERVED_BIT23	
22	SLAVE	SW: RW, HW: R HnpSlaveState. **Software HNP**: If Software writes with this bit SET it will cause the OTG-IP CORE to work as a Function **Hardware HNP**: A read with this bit SET indicates that currently the OTG is working as a Slave. This bit is not writeable in Hardware HNP mode.
21	MASTER	SW: RW, HW: R HnpMasterState. **Software HNP**: If Software writes with this bit SET it will cause the OTG-IP CORE to work as a Host **Hardware HNP**: A read with this bit SET indicates that currently this OTG is working as a Master. This bit is not writeable in Hardware HNP mode.

USB_HNP_CTL_STATUS (Continued)

Bits	Name	Description
20	BGEN	SW: RW, HW: R BandgapEnable. When SET enables the charge pump band gap.
19	CMPEN	SW: RW, HW: R ComparatorEnable. When SET enables the charge pump comparator.
18	ISBDEV	SW: R, HW: W IsBDevice. The hardware determines if this device is a B-DEVICE by checking the state of the ID pin. The default state of any OTG device is a B-DEVICE.
17	ISADEV	SW: R, HW: W IsADevice. The hardware determines if this device is an A-DEVICE by checking the state of the ID pin. A device will be considered an A-DEVICE if the ID pin is low.
16	RESERVED_BIT16	
15	SWVBUSPUL	SW: RW, HW: R SoftwareVBusPulse. If Software writes with this bit SET it will cause the VBUS to be pulled low.
14:13	RESERVED_BIT14_13	
12	SWAUTORST	SoftwareAutomaticReset This bit is only valid while in Software HNP. Software can write to this bit so that the hardware will automatically generate a downstream reset upon a connection to the port. This bit is only valid while the HnpMasterState bit is set. When the B-DEVICE receives a SET_FEATURE HNP_ENABLE, and the B-DEVICE removes its Pull-up resistor, software should set this bit to automatically drive reset upon a detection of connection. This bit is self-clearing upon the generation of the USB reset.
11	SWPUDP	SW: RW,HW: R SoftwarePullUpDP. Software can write to this bit to turn on or off the pull-up resistor. When SET the pull-up is enabled. When CLEAR the pull-down is enabled. This register is only used when in both Function Host Mode and Software HNP.
10	RESERVED_BIT10	
9	SWPDDM	SW: RW, HW: R SoftwarePullDownDM. When SET the pull-down on the DM line is enabled.

USB_HNP_CTL_STATUS (Continued)

Bits	Name	Description
8:4	HNPSTATE	SW: R, HW: W HnpState. This value is only valid in Hardware HNP. It represents the current state of the hardware HNP state machine. his field is not used in Software HNP. HNP State Machine State Mapping (HNPSTATE field in the HNP Control and Status Register bitmapping): Refer values below for HNP State Decodes. **HnpState Definition** 1_0000b A_IDLE 1_0001b A_MASTER 1_0010b A_SLAVE 1_0011b A_WAIT_VPULSE 1_0100b A_WAIT_DPULSE 1_0110b A_WAIT_CONN_A 1_0111b A_WAIT_CONN_B 1_1000b A_WAIT_VRISE 1_1001b A_SUSPEND 1_1010b A_WAIT_VFALL 1_1011b A_VBUS_ERR 1_1100b CONN_DEBOUNCE 1_1101b A_WAIT_ABREQ 1_1110b A_WAIT_RST 0_0000b B_IDLE 0_0001b B_MASTER 0_0010b B_SLAVE 0_0101b B_SHORT_DB 0_0110b B_WAIT_CONN_A 0_1010b B_WAIT_RST
3	CLRERROR	SW: W, HW: Sc HnpClearErrorState. A-DEVICE: When SET this bit when it has recovered from an over-current condition. Software needs to issue HnpClearErrorState to clear the error when the ADEVICE HNP is in A_VBUS_ERROR state and does not want to release the bus. As a B-DEVICE this bit is used to force the transition from B_SLAVE to B_IDLE. This is needed when B-DEVICE has generated an SRP, transitioned to B_SLAVE. If after 5 seconds, the Software still does not get an interrupt indicating an SRP success, then Software needs to force the B-DEVICE to come back to B_IDLE.
2	ADROPBUS	SW: RW, HW: R ADropVBus. When SET indicates that the Software of the A-DEVICE wants to power down the VBUS.

USB_HNP_CTL_STATUS (Continued)

Bits	Name	Description
1	ABBUSREQ	SW: RW, HW: RSc ABBusRequest. When SET indicates that the Software is requesting to be Master. This bit remains true when this device wants to use the bus, and is false when it no longer wants to use the bus.
0	RESERVED_BIT0	

0x020 USB_HNP_TIMERS1

Type: Read/Write
Clock: MCLK

Timer values used in HNP operation. This register defines the values of different timers used in the OTG. The time unit is 1ms. The register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

Bits	Name	Description
31:24	BWAITCONN	SW: RW, HW: R BWaitConnectTimer This timer is used by a B-DEVICE in the B_WAIT_CONN_A state, to wait for an A-DEVICE to signal a connection. If the A-DEVICE does not connect before a timeout (b_wait_conn_tmout), then the B-DEVICE stops waiting for a connection.
23:16	AWAITDISC	SW: RW, HW: R AWaitDisconnectTimer. This timer is started by an A-DEVICE when it enters the A_SUSPEND state. If the A-DEVICE does not detect a downstream disconnect before a timeout, then the A-DEVICE can end the session.
15:8	AWAITCONN	SW: RW, HW: R AWaitConnectTimer. This timer is used by an A-DEVICE in the A_WAIT_BCON state, to wait for the B-DEVICE to signal a connection. If the B-DEVICE does not connect before a timeout, then the A-DEVICE stops waiting for a connection. If the InsertionMode bit is not set the timeout will cause the AwaitBConnectTimeOutInterrupt.
7:0	AWAITVRISE	SW: RW, HW: R AWaitVRiseTimer. This timer is used by an A-DEVICE in the A_WAIT_VRISE state to wait for VBUS to rise to valid level (a_vbus_vld = 1). If VBUS does not reach a valid level before a timeout (a_wait_vrise_tmout = 1), then the A-DEVICE knows that the B-DEVICE is drawing too much current.

0x024 USB_HNP_TIMERS2

Type: Read/Write
Clock: MCLK

Timer values used in HNP operation. This register defines the values of different timers used in the HNP. All timer values are in 1ms increments. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

Bits	Name	Description
31:16	BSRPFAIL	SW: RW, HW: R BSrpFailTimer. This timer is set by software to determine when a Session Request has failed. The conditions of a failure are the VBUS must be over the b_session_vaild and the A_MASTER must drive a reset to the device.
15:10	RESERVED_BITS15_10	
9:5	SRPVPULW	SW: RW, HW: R SrpVBusPulseWidth. Defines the Session Request Protocol VBUS pulse length generated by BDEVICE.
4:0	SRPDPULW	SW: RW, HW: R SrpDataBusPulseWidth. Defines the session request protocol data line pulse length generated by BDEVICE.

0x028 USB_HNP_TIMER3_PULSE_CTL

Type: Read/Write
Clock: MCLK

Controls the SRP response methods and timer values used in HNP operation. The HnpTimer3PulseControl Register contains information for the hardware to know which kind of Session Request to respond to. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

Table 19-1 USB_HNP_TIMER3_PULSE_CTL

Bits	Name	Description
31:24	LNGDBNC	SW: RW, HW: R LongDebounceTimer. This timer is the de-bounce time following the detection of a transition on the USB bus. This timer is used only after the BConnectShortDebounceWindow has expired or following the transition from the A_WAIT_VRISE state.

Table 19-1 USB_HNP_TIMER3_PULSE_CTL (Continued)

Bits	Name	Description
23:16	A_SDB_WIN	SW: RW, HW: R BConnectShortDebounceWindow. This value is the maximum time the Hardware HNP should wait to do short de-bouncing of the connection. Once this timer expires it then will use theLongDebounceTimer to detect the connection.
15:7	RESERVED_BITS15_7	
6	INSERTION	SW: RW, HW: R InsertionModeSet. When SET the HNP logic in Hardware HNP will disable the a_wait_bcon_tmout to allow for operation as a legacy host.
5	DATAPULSE	SW: RW, HW: R UseDataPulseSrpDetection. Use data line pulsing to detect session request by the B-DEVICE.
4	VBUSPULSE	SW: RW, HW: R UseVBusPulseSrpDetection. Use VBUS pulsing to detect session request by the B-DEVICE.
3:0	RESERVED_BITS3_0	

0x02C USB_HNP_INT_STATUS

Type: Read/Write
Clock: MCLK

HNP interrupt sources. This register provides status on various HNP level events that cause hardware interrupts. When an HNP event occurs, hardware sets the corresponding bit in this register. When a bit becomes set, an interrupt is generated if the interrupt is enabled in the HnpInterruptEnableStatus Register. This register is reset by a system reset or a write to the ResetControl Register **ResetControlLogic** bit.

USB_HNP_INT_STATUS

Bits	Name	Description
31:16	RESERVED_BITS31_16	
15		Reserved.
14:9	RESERVED_BITS14_9	
8	AWAITBTO	SW: RC, HW: S AWaitBConnectTimeOutInterrupt. This timer is only used in Hardware HNP. When the AWaitConnectTimer expires this interrupt is asserted. The timer is started when in the A_WAIT_BCONN state and when it times out the state transitions to the A_WAIT_VFALL state. This interrupt is only asserted if the InsertionModeSet is cleared.
7	AIDLEBDTO	SW: RC, HW: S AIdleBDisconnectTimeOutInterrupt. This timer is only used in Hardware HNP. This interrupt is asserted when the AWaitDisconnectTimer expires. This timer is started in the A_SUSPEND state and when it expires it will cause the transition to A_WAIT_VFALL.
6	SRPSUCFAIL	SW: RC, HW: S SrpSuccessFailInterrupt. This interrupt is only valid in Hardware HNP. B-DEVICE needs two conditions to know that its session request is successful. 1. The state must be in B_SLAVE 2. It must detect that A-DEVICE is resetting it within the BSrpFailTimer value.
5	SRPINT	SW: RC, HW: S SessionRequestInterrupt. After receiving this interrupt software must decide if it wants to respond to the Session Request. This interrupt is asserted when either a VBUS pulse is detected and/or a Data pulse is detected.
4	VBUSERROR	SW: RC, HW: S VBusErrorInterrupt. This interrupt is only valid in Hardware HNP. A-DEVICE VBUS error, due to over current or battery problem.

USB_HNP_INT_STATUS (Continued)

Bits	Name	Description
3	BSESSVAILD	SW: RC, HW: S BSessionValidChangeInterrupt. This interrupt is only valid in Software HNP and Function Host Mode. When asserted this means that the VBUS has crossed the b_session_vld threshold. Software should read the HnpControlStatus Register to determine the current status of the VBUS.
2	ASESSVAILD	SW: RC, HW: S ASessionValidChangeInterrupt. This interrupt is only valid in Software HNP and Function Host Mode. When asserted this means that the VBUS has crossed the a_vbus_vld level. Software should read the HnpControlStatus Register to determine the current status of the VBUS.
1	MASSLVCHG	SW: RC, HW: S MasterSlaveChangeInterrupt. This interrupt is only valid in Hardware HNP. This bit is asserted when HnpState enters A_SLAVE, A_MASTER, to inform the application so that appropriate action can be taken by software.
0	IDCHANGE	SW: RC, HW: S IDChangeInterrupt. Indicates the state of the ID pin has changed. Software should read the HnpControlStatus Register to determine the current status of the ID pin.

0x030 USB_HNP_INT_ENA

Type: Read/Write
Clock: MCLK

HNP interrupt mask. This register is reset by a system reset or a write to the ResetControl Register ResetControlLogic bit.

USB_HNP_INT_ENA

Bits	Name	Description
31:16	RESERVED_BITS31_16	
15		Reserved.
14:9	RESERVED_BITS14_9	
8	AWAITBTOEN	SW: RW, HW: R AWaitBConnectTimeOutInterruptEnable. Enable bit of AWaitBConnectTimeOutInterrupt.
7	AIDLEBDTON	SW: RW, HW: R AIdleBDisconnectTimeOutInterruptEnable. Enable bit of AIdleBDisconnectTimeOutInterrupt.

USB_HNP_INT_ENA (Continued)

Bits	Name	Description
6	SRPSUCFAILEN	SW: RW, HW: R SrpSuccessFailInterruptEnable. Enable bit of SrpSuccessFailInterrupt.
5	SRPINTEN	SW: RW, HW: R SessionRequestInterruptEnable. Enable bit of SessionRequestInterrupt.
4	VBUSERROREN	SW: RW, HW: R VBusErrorInterruptEnable. Enable bit of VBusErrorInterrupt.
3	ABSESVAILDEN	SW: RW, HW: R ABSessionValidInterruptEnable Enable bit of ABSessionValidInterrupt.
2	AVBUSVAILDEN	SW: RW, HW: R AVBusValidInterruptEnable Enable bit of AVBusValidInterrupt.
1	MASSLVCHGEN	SW: RW, HW: R MasterSlaveChangeInterruptEnable. Enable bit of MasterSlaveChangeInterrupt.
0	IDCHANGEEN	SW: RW, HW: R IDChangeInterruptEnable. Enable bit of IDChangeInterrupt.

19.1.2 Host registers

Effect of resets on host registers

There are several reset sources for the USB Host Controller: system master reset, a transition to Host Controller USBRESET, state and software-initiated controller resets. The software-initiated controller resets have the same effect as the system reset except they can selectively reset particular blocks. These resets are accomplished by writing to the ResetControl Register with **ResetHostSIE**, **ResetRootHub**, and/or **ResetHostController** bits SET.

The USB bus reset is also generated by software. Writing to the HostControl Register and setting the **HostControllerUSBState** to USBRESET, will reset the Host Controller and the Root Hub. After the reset has completed, the Host Controller will automatically enter the USBSUSPEND state.

0x080 USB_HOST_CTL

Type: Read/Write
Clock: MCLK

Host controller configuration.

Bits	Name	Description
31	HCRESET	SW: W, HW: Sc HostControllerReset. Writing a ‘1’ will generate a software-initiated hardware reset. Writing a '0' leaves this bit unchanged.
30:18	RESERVED_BIT31_18	
17:16	SCHEDOVR	SW: R, HW: W SchedulerOverrunCount. This count is incremented each time a Scheduler Overrun event is detected in the Host Controller.
15:5	RESERVED_BIT15_5	
4	RMTWUEN	SW: RW, HW: R RemoteWakeUpEnable. Setting this bit allows the Host to wakeup when a resume event is detected downstream.
3:2	HCUSBSTE	SW: RW, HW: R HostControllerUSBState. Software uses these bits to control the state of the USB state machine. Code State 00b: USBRESET 01b: USBRESUME 10b: USBOPERATIONAL 11b: USBSUSPEND Writing these states will control the Root Hub.
1:0	CTLBLKSR	SW: RW, HW: R ControlBulkServiceRatio. This setting dictates the number of CONTROL packets sent out for each BULK packet. Code Ratio 00b = 1:1 01b = 2:1 10b = 4:1 11b = 8:1

0x088 USB_H_SYSTEM_INT_STATUS

Type: Read/Write
Clock: MCLK

Interrupt status.

Bits	Name	Description
31:7	RESERVED_BIT31_7	
6	PSCINT	SW: RC, HW: S PortStatusChangeInterrupt. When asserted, indicates that a Port Status has changed.
5	FMOFINT	SW: RC, HW: S FrameNumberOverflowInterrupt. When asserted, indicates that the Frame Number register has over flown.
4	HERRINT	SW: RC, HW: S HostErrorInterrupt. When asserted, indicates that the Host has encountered a major scheduling error.
3	RESDETINT	SW: R, HW: C S ResumeDetectedInterrupt. When asserted, this bit indicates that the Host has detected a resume transition on the USB bus.
2	SOFINT	SW: RC, HW: S StartOfFrameInterrupt. When asserted, this indicates that an SOF has been transmitted.
1	DONEINT	SW: RC, HW: S DoneRegisterInterrupt. When asserted, this indicates that there are completed ETDs on the ETDDoneStatus Register.
0	SORINT	SW: RC, HW: S SchedulerOverrunInterrupt. When asserted, indicates that the Host has experienced a Scheduling Overrun condition.

0x08C USB_H_SYSTEM_INT_ENA

Type: Read/Write
Clock: MCLK

Interrupt mask.

Bits	Name	Description
31	RESERVED_BIT31	
6	PSCIEN	SW: RC, HW: S PortStatusChangeInterruptEnable. When SET, this will allow the PortStatusChangeInterrupt to be asserted.
5	FMOFIEN	SW: RC, HW: S FrameNumberOverflowInterruptEnable. When SET, this will allow the FrameNumberOverflowInterrupt to be asserted.
4	HERRIEN	SW: RC, HW: S HostErrorInterruptEnable. When SET, this will allow the HostErrorInterrupt to be asserted.
3	RESDETIEN	SW: RC, HW: S ResumeDetectedInterruptEnable. When SET, this will allow the ResumeDetectedInterrupt to be asserted.
2	SOFIEN	SW: RC, HW: S StartOfFrameInterruptEnable. When SET, this will allow the StartOfFrameInterrupt to be asserted.
1	DONEIEN	SW: RC, HW: S DoneRegisterInterruptEnable. When SET, this will allow the DoneRegisterInterrupt to be asserted.
0	SORIEN	SW: RC, HW: S SchedulerOverrunInterruptEnable. When SET, this will allow the SchedulerOverrunInterrupt to be asserted.

0x098 USB_H_X_BUFFER_INT_STATUS

Type: Read/Write
Clock: MCLK

X buffer Interrupt status.

Bits	Name	Description
31:0	XBUFnINT	SW: RC, HW: S XBuffer<n>Interrupt. When asserted indicates that the X buffer of ETD <n> has been emptied (for OUTs) or filled (for INs) by the host. Writing the asserted bit back to the register clears this bit.

0x09C USB_H_Y_BUFFER_INT_STATUS

Type: Read/Write
Clock: MCLK

Y buffer Interrupt status.

Bits	Name	Description
31:0	YBUFnINT	SW: RC, HW: S YBuffer<n>Interrupt When asserted indicates that the Y buffer of ETD <n> has been emptied (for OUTs) or filled (for INs) by the host. Writing the asserted bit back to the register clears this bit.

0x0A0 USB_H_XY_INT_ENA

Type: Read/Write
Clock: MCLK

X/Y Interrupt enables.

Bits	Name	Description
31:0	XYnIEN	SW: RW, HW: R XY<n>InterruptEnable. When set allows for an interrupt for both X and Y buffers on ETD <n> to be asserted.

0x0A8 USB_H_X_FILLED_STATUS

Type: Read/Write
Clock: MCLK

X buffer filled status. Software toggles the bits in this register by writing to it with the appropriate bits SET. For an OUT ETD, the application indicates that the X Buffer has been filled and is ready for transfer by setting the XFilled<n>Status bit. Once the buffer has been completely emptied by the Host Controller this bit is cleared. For an IN ETD, the host indicates that it has completely filled the X Buffer by setting the appropriate XFilled<n>Status bit. Once the buffer has been completely emptied by Software this bit should be cleared.

Bits	Name	Description
31:0	XFILLn	SW: RC, HW: S XFilled<n>Status. When SET, this indicates that the corresponding X Buffer is Full.

0x0AC USB_H_Y_FILLED_STATUS

Type: Read/Write
Clock: MCLK

Y buffer filled status. Same operation as the XFilledStatus Register (0x05C) except the action is for the Y Buffer.

Bits	Name	Description
31:0	YFILLn	SW: RC, HW: S YFilled<n>Status. When SET, this indicates that the corresponding YBuffer is Full.

0x0C0 USB_ETD_ENA

Type: Read/Write
Clock: MCLK

The Host Controller supports a maximum of 32 ETDs. This register is used by software to tell the Host Controller that an ETD is ready to be processed. When the descriptor is properly configured and the transfer is ready to commence, the corresponding bit should be set in this register. The bit is cleared by the ETD Clear Register.

Bits	Name	Description
31:0	ETDnSET	SW: RW, HW: R ETD<n>SET When SET indicates that an ETD is enabled and ready to be processed by the host.

0x0C4 USB_ETD_ENA_CLEAR

Type: Read/Write
Clock: MCLK

When this bit is set (1), the ETDSet Register will be cleared (0). The ETD register cannot be cleared by writing a 0 to the ETDSet Register.

Bits	Name	Description
31:0	ETDnCLR	ETD<n>Clear When SET indicates that ETD <n> is disabled. This register cannot be read.

0x0CC USB_H_IMMEDIATE_INT

Type: Read/Write
Clock: MCLK

Informs the host controller that all ETDs that are set here should generate an interrupt immediately upon completion, instead of waiting for an SOF. In order to reduce the number of interrupts occurring in a frame, all ETD<n>DoneStatus are flagged on the SOF following the retirement of the ETD. However, if Software wishes to see the done status flagged immediately when the ETD is retired it can by setting the correct bit in this register.

Bits	Name	Description
31:00	IMMINTn	SW: RW, HW: R ImmediateInterrupt<n>. When SET the Host Controller will assert an interrupt immediately upon retirement of an ETD instead of waiting until the SOF.

0x0D0 USB_H_ETD_DONE_STATUS

Type: Read/Write
Clock: MCLK

Asserts which ETDs are done processing. This register indicates which ETD transfers have been completed and are waiting for further Software instructions. The ETDDoneStatus Register is updated immediately upon retirement of an ETD, however, it will not assert an interrupt until the SOF marker unless the corresponding ImmediateInterrupt bit is SET. Software should read this register to determine which ETDs are in need of service.

Bits	Name	Description
31:00	ETDnDONE	SW: RC, HW: S ETD<n>DoneStatus. When asserted indicates that ETD<n>DoneStatus has been retired, either through normal operation or through an error condition. The CompletionCode field of the ETD should be read to determine the reason for retirement. A complete list of completion codes is found in ETD completion code descriptions.

0x0D4 USB_ETD_DONE_ENA

Type: Read/Write
Clock: MCLK

ETD done mask. This register indicates which ETD<n>DoneStatus will generate an interrupt when set. This applies to interrupts generated either immediately or on the SOF marker.

Bits	Name	Description
31:00	ETDnDNEN	SW: RW, HW: R ETD<n>DoneEnable. When SET allows ETD <n> to generate done interrupts. The ETD<n>DoneEnable must be SET for both immediate and SOF marker interrupts.

0x0E0 USB_H_FRAME_NUMBER

Type: Read/Write
Clock: MCLK

The current Frame Number.

Bits	Name	Description
31:16	RESERVED_BIT31_16	
15:0	FRMNUMB	SW: RW, HW: RW FrameNumber. This contains the current frame number for the Host Controller. This number is placed in the SOF packet used to signal the beginning of the new frame.

0x0E4 USB_LOW_SPEED_THRESH

Type: Read/Write
Clock: MCLK

Defines the Low Speed Threshold value.

Bits	Name	Description
31:11	RESERVED_BIT31_11	
10:0	LSTHRESH	SW: RW, HW: R LowSpeedThreshold. This is the number in USB full speed bit times that are required to be remaining in a frame to allow a low speed packet to be transmitted. All low speed packets have a MaxPacketSize maximum of 8 bytes, so this should be set to the number of bit times needed to transmit an 8 byte packet. The default value of 628h should not be changed for normal operation.

0x0E8 USB_ROOT_HUB_DESCRIPTOR_A

Type: Read/Write
Clock: MCLK

Root Hub Characteristics Register 1.

Bits	Name	Description
31:24	PWRTOGOOD	SW: R, HW: R PowerOnToPowerGoodTime. This byte specifies the duration that software must wait before accessing a powered-on port of the Root Hub.
23:13	RESERVED_BIT23_13	
12	NOOVRCURP	SW: R, HW: R NoOverCurrentProtection. This bit describes how the over current status for the Root Hub is reported. In this implementation, the over current status is reported collectively for all downstream ports.
11	OVRCURPM	SW: R, HW: R OverCurrentProtectionMode. This bit describes how the over current status for the Root Hub ports are reported. This implementation uses gang-power and gang-over current reporting.
10	DEVTYPE	SW: R, HW: R DeviceType. This bit specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read as 0.
9	PWRSWTMD	SW: R, HW: R PowerSwitchingMode. This bit is used to specify how the power switching of the Root Hub ports is controlled. In this implementation, each port is powered individually. This allows for power to be controlled on a per-port basis. The port responds only to port power commands (Set/ClearPortPower).
8	NOPWRSWT	SW: R, HW: R NoPowerSwitching. All ports are power switched. This bit will always read as SET.
7:0	NDNSTMPRT	SW: R, HW: R NumberDownstreamPorts. These bits specify the number of downstream ports supported by the Root Hub.

0x0EC USB_ROOT_HUB_DESCRIPTOR_B

Type: Read/Write
Clock: MCLK

Root Hub Characteristics Register 2.

Bits	Name	Description
31:24	RESERVED_BIT31_24	
23:16	PRTPWRCM	SW: R, HW: R PortPowerControlMask. Each bit indicates if a port is affected by a global power control command when PowerSwitchingMode is set. When set, the port's power state is only affected by per-port power control (Set/ClearPortPower). When cleared, the port is controlled by the global power switch (Set/ClearGlobalPower).
15:8	RESERVED_BIT15_8	
7:0	DEVREMOVE	SW: R, HW: R DeviceRemovable. Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable. bit 0: this port can be OTG/ #1 port (during HNP negotiation) bit 1: Device attached to Port #2 bit 2: Device attached to Port #3

0x0F0 USB_ROOT_HUB_STATUS

Type: Read/Write
Clock: MCLK

Root Hub Status and Change Bits.

USB_ROOT_HUB_STATUS

Bits	Name	Description
31	CLRMTWUE	SW: W, HW: Sc ClearRemoteWakeupEnable. Setting this bit clears DeviceRemoteWakeupEnable.
30:18	RESERVED_BIT30_18	
17	OVRCURCHG	SW: RC, HW: R OverCurrentIndicatorChange. This bit is set by hardware when a change has occurred to the OCI field of this register. The HCD clears this bit by writing a ‘1’. Writing a ‘0’ has no effect.

USB_ROOT_HUB_STATUS (Continued)

Bits	Name	Description
16	LOCPWRSC	SW: R, HW: R LocalPowerStatusChange. The Root Hub does not support the local power status feature; thus, this bit is always read as ‘0’.
15	DEVCONWUE	SW: RW, HW: R DeviceConnectWakeupEnable. This bit enables a ConnectStatusChange bit as a resume event, causing a USBSUSPEND to USBRESUME state transition and setting the ResumeDetected interrupt. ConnectStatusChange is encoded as follows: 0 = Not a remote wakeup event. 1 = remote wakeup event. Writing a '1' sets DeviceRemoveWakeupEnable. Writing a '0' has no effect.
14:2	RESERVED_BIT14_2	
1	OVRCURI	SW: R, HW: S OverCurrentIndicator. This bit reports over current conditions when the global reporting is implemented. When set, an over current condition exists. When cleared, all power operations are normal. If per-port over current protection is implemented this bit is always ‘0’
0	LOCPWRS	SW: R, HW: R LocalPowerStatus. The Root Hub does not support the local power status feature; thus, this bit is always read as ‘0’.

0x0F4+4n USB_PORT_STATUS_n, n=[0..2]

Type: Read/Write
Clock: MCLK

Status and Change Bits.

USB_PORT_STATUS_n

Bits	Name	Description
31:21	RESERVED_BIT31_21	
20	PRTRSTSC	SW: RC, HW: S PortResetStatusChange. This bit is SET at the end of the 50-ms port reset signal. Only a write back with this bit SET will clear this bit.

USB_PORT_STATUS_n (Continued)

Bits	Name	Description
19	OVRCURIC	SW: RC, HW: S PortOverCurrentIndicatorChange. This bit is SET when an PortOverCurrentIndicator has changed on this port and is only valid if over current conditions are reported on a per-port basis. Only a write back with this bit SET will clear this bit.
18	PRTSTATSC	SW: RC, HW: S PortSuspendStatusChange. This bit is set when the resume sequence has been fully completed. A write back with this bit SET or ResetStatusChange is SET when will clear this bit.
17	PRTENBLSC	SW: RC, HW: S PortEnableStatusChange. This bit is set when hardware events cause the PortEnableStatus bit to be cleared. Only a write back with this bit SET will clear this bit.
16	CONNECTSC	SW: RC, HW: S ConnectStatusChange. When a connect or disconnect event occurs this bit will be SET. A write with this bit SET will clear this bit. Writing a ‘0’ has no effect. If CurrentConnectStatus is cleared when a SetPortReset, SetPortEnable, or SetPortSuspend write occurs, this bit is SET to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected. Only a write back with this bit SET will clear this bit.
15:10	RESERVED_BIT15_10	
9	LSDEVCON	SW: RW, HW: RW (read) LowSpeedDeviceAttached. This bit indicates the speed of the device attached to this port. When SET, a low speed device is attached to this port. When CLEARED, a full speed device is attached to this port. This field is valid only when the CurrentConnectStatus is set. (write) ClearPortPower. The HCD clears the PortPowerStatus bit by a write with this bit SET. Writing with this bit CLEARED has no effect.
8	PRTPWRST	SW: RW, HW: RW (read) PortPowerStatus. This bit reflects the port’s power status. This bit is cleared if an over current condition is detected. HCD sets this bit by writing SetPortPower. HCD clears this bit by writing ClearPortPower. When port power is disabled, CurrentConnectStatus, PortEnableStatus, PortSuspendStatus, and PortResetStatus should be reset. (write) SetPortPower. The HCD writes a ‘1’ to set the PortPowerStatus bit. Writing a ‘0’ has no effect.
7:5	RESERVED_BITS7_5	

USB_PORT_STATUS_n (Continued)

Bits	Name	Description
4	PRTRSTST	SW: RW, HW: RW (read) PortResetStatus. When this bit is SET by a write to SetPortReset, and will remain asserted until the port reset signaling is completed. When reset is completed, this bit is cleared and PortResetStatusChange is SET. This bit cannot be set if CurrentConnectStatus is cleared. (write) SetPortReset. The HCD sets the port reset signaling by writing a '1' to this bit. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortResetStatus, but instead sets ConnectStatusChange. This informs the driver that it attempted to reset a disconnected port.
3	PRTOVRCURI	SW: RW RW (read) PortOverCurrentIndicator. This bit is only valid when the Root Hub is configured so that over current conditions are reported on a per-port basis. If per-port over current reporting is not supported, this bit is always read as '0'. If cleared, all power operations are normal for this port. If SET, an overcurrent condition exists on this port. (write) ClearSuspendStatus. The HCD writes a '1' to initiate a resume. Writing a '0' has no effect. A resume is initiated only if PortSuspendStatus is SET.
2	PRTSUSPST	SW: RW, HW: RW (read) PortSuspendStatus. When SET bit indicates the port is suspended or in the resume sequence. It is SET by a SetSuspendState write and at the end of the resume interval. This bit cannot be set if CurrentConnectStatus is cleared. This bit is also cleared when PortResetStatusChange is set at the end of the port reset or when the Host Controller is placed in the USBRESUME state. (write) SetPortSuspend. The HCD sets the PortSuspendStatus bit by writing a '1' to this bit. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortSuspendStatus; instead it sets ConnectStatusChange. This informs the driver that it attempted to suspend a disconnected port.

USB_PORT_STATUS_n (Continued)

Bits	Name	Description
1	PRTENABST	SW: RW, HW: RW (read) PortEnableStatus. When SET this bit indicates whether the port is enabled. The Root Hub will clear this bit when an over current condition, disconnect event, the power is switched-off, or operational bus error has occurred. A change in this bit causes PortEnabledStatusChange to be SET. HCD sets this bit by writing SetPortEnable and clears it by writing ClearPortEnable. This bit cannot be set when CurrentConnectStatus is cleared. This bit is also set, if not already, at the completion of a port reset when ResetStatusChange is set or port suspend when SuspendStatusChange is set. 0 : port disabled 1 : port enabled (write) SetPortEnable The HCD sets PortEnableStatus by writing a '1'. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortEnableStatus, but instead sets ConnectStatusChange. This informs the driver that it attempted to enable a disconnected port.
0	CURCONST	SW: RW, HW: RW (read) CurrentConnectStatus. This bit reflects the current state of the downstream port. A read with this bit SET indicates a device is connected. (write) ClearPortEnable The HCD writes a '1' to this bit to clear the PortEnableStatus bit. Writing a '0' has no effect. The CurrentConnectStatus is not affected by any write.

19.1.3 Function registers

Effect of different resets on registers

There are several reset resources for the USB Function Controller: system reset and a software initiated reset. The software initiated reset has the same effect as the system reset, however it can be localized to a particular block.

The different reset options are described in the ResetControl Register (0x010) in section 5.1.5. All of the reset bits are self-clearing (SC). The USB bus reset will trigger an interrupt if enabled to alert Software that a bus reset has been detected. This interrupt is triggered on both the entering and exiting of the USB bus reset state. Software can use this interrupt to then set the soft reset bit to reset the Function Controller, if desired.

0x040 USB_FUNCTION_CMD_STATUS

Type: Read/Write
Clock: MCLK

Current status of the function controller.

Bits	Name	Description
31:8	RESERVED_BITS31_8	
7	SOFTRESET	SW: W, HW: Sc SoftResetFunctionController. Writing with this bit SET will generate a software-initiated hardware reset. Writing a '0' leaves this bit unchanged.
6:4	RESERVED_BIT6_4	
3	BADISOAP	BadISOAccepted Writing with this bit SET will cause the Function Controller to accept an ISO packet with corrupted data, like data CRC error. When this bit is cleared, or by default, bad ISO packets are automatically dropped.
2	SUSPDET	SW: RW, HW: R SuspendDetected. This bit indicates that the Function Controller is in the Suspended state. Writing with this bit SET will cause device to enter the suspended state. Writing with this bit CLEARED leaves this bit unchanged.
1	RSMINPROG	SW: RW, HW: RW ResumeInProgress. When the Function Controller is in suspend with clock running, Software can set this bit to generate a resuming signal to the upstream port by writing a ‘1’. Writing a '0' leaves this bit unchanged.
0	RESETDET	SW: R, HW: W USBBusResetDetected. A Software read with this bit SET indicates that the USB device has detected bus reset.

0x044 USB_DEVICE_ADDRESS

Type: Read/Write
Clock: MCLK

Assigned address for the function controller from last enumeration. The value contained in this register is the Host assigned address. This is the ONLY address to which the function controller will respond. When there is a system reset or the ResetFunctionController bit is SET, this value will be cleared to the register's default value. In the status stage of the standard SET_ADDRESS request, Software should write received device address into this register immediately after sending an empty packet to the Host.

Bits	Name	Description
31:7	RESERVED_BITS31_7	
6:0	DEVADDR	SW: RW, HW: R DeviceAddress. This field contains the device address.

0x048 USB_F_SYSTEM_INT_STATUS

Type: Read/Write
Clock: MCLK

Interrupt vector.

Bits	Name	Description
31:5	RESERVED_BITS31_5	
4	SOFDETINT	SW: RC, HW: S SOF DetectedInterrupt. When asserted indicates that the USB device has received a Start Of Frame.
3	DONEREGINT	SW: RC, HW: S DoneRegisterInterrupt. When asserted this bit indicates one or more bits in EndpointDoneStatus Register (0x070) is set.
2	SUSPDETINT	SW: RC, HW: S SuspendDetectedInterrupt. When asserted indicates that the USB device has detected an active to suspend state on the bus.
1	RSMFININT	SW: RC, HW: S ResumeFinishedInterrupt. When asserted indicates that the USB device has detected a suspend to active state change.
0	RESETINT	SW: RC, HW: S ResetDetectedInterrupt. The interrupt will be asserted on the rising edge and falling edge of the bus reset detection.

0x04C USB_F_SYSTEM_INT_ENA

Type: Read/Write
Clock: MCLK

Interrupt mask.

USB_F_SYSTEM_INT_ENA

Bits	Name	Description
31:5	RESERVED_BITS31_5	
4	SOFDETIEN	SW: RC, HW: R SOFDetectedInterruptEnable. When SET this will allow the SOFDetectedInterrupt to be asserted.
3	DONEREGIEN	SW: RC, HW: R DoneRegisterInterruptEnable. When SET this will allow the DoneRegisterInterrupt to be asserted.
2	SUSPDETIEN	SW: RC, HW: R SuspendDetectedInterruptEnable. When SET this will allow the SuspendDetectedInterrupt to be asserted.
1	RSMFINIEN	SW: RC, HW: R ResumeFinishedInterruptEnable. When SET this will allow the ResumeFinishedInterrupt to be asserted.
0	RESETIEN	SW: RC, HW: R USBBusResetDetectedEnable. When SET this will allow the USBBusResetDetected to be asserted.

0x050 USB_F_X_BUFFER_INT_STATUS

Type: Read/Write
Clock: MCLK

X buffer Interrupt status.

USB_F_X_BUFFER_INT_STATUS

Bits	Name	Description
31	XBUF31ININT	**ALL odd-numbered bits:** SW: RC, HW: S XBuffer<n>InInterrupt. When asserted indicates that the XOUT buffer of endpoint <n> has been filled by the host. Writing the asserted bit back to the register clears this bit.
30	XBUF30OUTINT	**ALL even-numbered bits:** SW: RC, HW: S XBuffer<n>OutInterrupt. When asserted indicates that the XIN buffer of endpoint <n> has been emptied by the host. Writing the asserted bit back to the register clears this bit.
29	XBUF29ININT	
28	XBUF28OUTINT	
27	XBUF27ININT	
26	XBUF26OUTINT	
25	XBUF25ININT	
24	XBUF24OUTINT	
23	XBUF23ININT	
22	XBUF22OUTINT	
21	XBUF21ININT	
20	XBUF20OUTINT	
19	XBUF19ININT	
18	XBUF18OUTINT	
17	XBUF17ININT	
16	XBUF16OUTINT	
15	XBUF15ININT	
14	XBUF14OUTINT	
13	XBUF13ININT	
12	XBUF12OUTINT	

USB_F_X_BUFFER_INT_STATUS (Continued)

Bits	Name	Description
11	XBUF11ININT	
10	XBUF10OUTINT	
9	XBUF9ININT	
8	XBUF8OUTINT	
7	XBUF7ININT	
6	XBUF6OUTINT	
5	XBUF5ININT	
4	XBUF4OUTINT	
3	XBUF3ININT	
2	XBUF2OUTINT	
1	XBUF1ININT	
0	XBUF0OUTINT	

0x054 USB_F_Y_BUFFER_INT_STATUS

Type: Read/Write
Clock: MCLK

Y buffer Interrupt status.

USB_F_Y_BUFFER_INT_STATUS

Bits	Name	Description
31	YBUF31ININT	**ALL odd-numbered bits:** SW: RC, HW: S YBuffer<n>InInterrupt. When asserted indicates that the YOUT buffer of endpoint <n> has been filled by the host. Writing the asserted bit back to the register clears this bit.
30	YBUF30OUTINT	**ALL even-numbered bits:** SW: RC, HW: S YBuffer<n>OutInterrupt. When asserted indicates that the YIN buffer of endpoint <n> has been emptied by the host. Writing the asserted bit back to the register clears this bit.
29	YBUF29ININT	
28	YBUF28OUTINT	
27	YBUF27ININT	
26	YBUF26OUTINT	
25	YBUF25ININT	
24	YBUF24OUTINT	
23	YBUF23ININT	
22	YBUF22OUTINT	
21	YBUF21ININT	
20	YBUF20OUTINT	
19	YBUF19ININT	
18	YBUF18OUTINT	
17	YBUF17ININT	
16	YBUF16OUTINT	
15	YBUF15ININT	
14	YBUF14OUTINT	
13	YBUF13ININT	
12	YBUF12OUTINT	

USB_F_Y_BUFFER_INT_STATUS (Continued)

Bits	Name	Description
11	YBUF11ININT	
10	YBUF10OUTINT	
9	YBUF9ININT	
8	YBUF8OUTINT	
7	YBUF7ININT	
6	YBUF6OUTINT	
5	YBUF5ININT	
4	YBUF4OUTINT	
3	YBUF3ININT	
2	YBUF2OUTINT	
1	YBUF1ININT	
0	YBUF0OUTINT	

0x058 USB_F_XY_INT_ENA

Type: Read/Write
Clock: MCLK

X/Y Interrupt enables.

USB_F_XY_INT_ENA

Bits	Name	Description
31	XY31INEN	**ALL odd-numbered bits:** SW: RW, HW: R XY<n>InInterruptEnable. When set allows for an interrupt on both X/Y IN for endpoint <n> to be asserted.
30	XY30OUTEN	**ALL even--numbered bits:** SW: RW, HW: R XY<n>OutInterruptEnable. When set allows for an interrupt on both X/Y OUT for endpoint <n> to be asserted.
29	XY29INEN	
28	XY28OUTEN	
27	XY27INEN	
26	XY26OUTEN	

USB_F_XY_INT_ENA (Continued)

Bits	Name	Description
25	XY25INEN	
24	XY24OUTEN	
23	XY23INEN	
22	XY22OUTEN	
21	XY21INEN	
20	XY20OUTEN	
19	XY19INEN	
18	XY18OUTEN	
17	XY17INEN	
16	XY16OUTEN	
15	XY15INEN	
14	XY14OUTEN	
13	XY13INEN	
12	XY12OUTEN	
11	XY11INEN	
10	XY10OUTEN	
9	XY9INEN	
8	XY8OUTEN	
7	XY7INEN	
6	XY6OUTEN	
5	XY5INEN	
4	XY4OUTEN	
3	XY3INEN	
2	XY2OUTEN	
1	XY1INEN	
0	XY0OUTEN	

0x05C USB_F_X_FILLED_STATUS

Type: Read/Write
Clock: MCLK

X buffer filled status. Software toggles the bits in this register by writing to it with the appropriate bits set.

For an IN endpoint, the application indicates that the X Buffer has been filled and is ready for transfer by setting the XFilled<n>InStatus bit. Once the buffer has been completely emptied by the Hardware, this bit is cleared. For an OUT endpoint, the host indicates that it has completely filled the Xbuffer by setting the appropriate XFilled<n>OutStatus. Once the buffer has been completely emptied by Software, this bit should be cleared.

USB_F_X_FILLED_STATUS

Bits	Name	Description
31	XFILL31IN	**ALL odd-numbered bits:** SW: S, HW: RC XFilled<n>InStatus. When a bit is asserted, it indicates to the Hardware that the <n> X Buffer has been filled and the function controller can begin sending the data on the next IN.
30	XFILL30OUT	**ALL even-numbered bits:** SW: RC, HW: S XFilled<n>OutStatus. When a bit is cleared, it indicates to the Hardware that the <n> X Buffer has been drained and the function controller can begin receiving data from the next OUT.
29	XFILL29IN	
28	XFILL28OUT	
27	XFILL27IN	
26	XFILL26OUT	
25	XFILL25IN	
24	XFILL24OUT	
23	XFILL23IN	
22	XFILL22OUT	
21	XFILL21IN	
20	XFILL20OUT	
19	XFILL19IN	
18	XFILL18OUT	
17	XFILL17IN	

USB_F_X_FILLED_STATUS (Continued)

Bits	Name	Description
16	XFILL16OUT	
15	XFILL15IN	
14	XFILL14OUT	
13	XFILL13IN	
12	XFILL12OUT	
11	XFILL11IN	
10	XFILL10OUT	
9	XFILL9IN	
8	XFILL8OUT	
7	XFILL7IN	
6	XFILL6OUT	
5	XFILL5IN	
4	XFILL4OUT	
3	XFILL3IN	
2	XFILL2OUT	
1	XFILL1IN	
0	XFILL0OUT	

0x060 USB_F_Y_FILLED_STATUS

Type: Read/Write
Clock: MCLK

Y buffer filled status. This register has the same operation principles as the XFilledStatus Register (0x05C) except for the Y Buffer.

USB_F_Y_FILLED_STATUS

Bits	Name	Description
31	YFILL31IN	**ALL odd-numbered bits:** SW: S, HW: RC YFilled<n>InStatus. When a bit is asserted, it indicates to the Hardware that the <n> Y Buffer has been filled and the function controller can begin sending the data on the next IN.

USB_F_Y_FILLED_STATUS (Continued)

Bits	Name	Description
30	YFILL30OUT	**ALL even-numbered bits:** SW: RC, HW: S YFilled<n>OutStatus. When a bit is cleared, it indicates to the Hardware that the <n> Y Buffer has been drained and the function controller can begin receiving data from the next OUT.
29	YFILL29IN	
28	YFILL28OUT	
27	YFILL27IN	
26	YFILL26OUT	
25	YFILL25IN	
24	YFILL24OUT	
23	YFILL23IN	
22	YFILL22OUT	
21	YFILL21IN	
20	YFILL20OUT	
19	YFILL19IN	
18	YFILL18OUT	
17	YFILL17IN	
16	YFILL16OUT	
15	YFILL15IN	
14	YFILL14OUT	
13	YFILL13IN	
12	YFILL12OUT	
11	YFILL11IN	
10	YFILL10OUT	
9	YFILL9IN	
8	YFILL8OUT	
7	YFILL7IN	
6	YFILL6OUT	
5	YFILL5IN	
4	YFILL4OUT	

USB_F_Y_FILLED_STATUS (Continued)

Bits	Name	Description
3	YFILL3IN	
2	YFILL2OUT	
1	YFILL1IN	
0	YFILL0OUT	

0x064 USB_ENDPOINT_ENA

Type: Read/Write
Clock: MCLK

Informs the function controller which endpoints are enabled. The Function Controller can support up to 32 physical endpoints or 16 bi-directional endpoints. The controller uses the values contained in this register to indicate which requests from the host to respond to. Software can choose to enable only the IN, the OUT or both. After either a system reset or a soft reset all of the endpoints are disabled.

USB_ENDPOINT_ENA

Bits	Name	Description
31	EP31INEN	**ALL odd-numbered bits:** SW: RW, HW: R Endpoint<n>InEnabled. When SET indicates that endpoint <n> is ready to respond to the host on the next IN token. If there is not any data to be sent, the Function Controller will respond with a NAK.
30	EP30OUTEN	**ALL even-numbered bits:** SW: RW, HW: R Endpoint<n>OutEnabled. When SET indicates that endpoint <n> is read to respond to the host on the next OUT token. If the Function Controller is unable to receive any data because the X and/or Y Buffers are full, it will respond with a NAK.
29	EP29INEN	
28	EP28OUTEN	
27	EP27INEN	
26	EP26OUTEN	
25	EP25INEN	
24	EP24OUTEN	
23	EP23INEN	
22	EP22OUTEN	

USB_ENDPOINT_ENA (Continued)

Bits	Name	Description
21	EP21INEN	
20	EP20OUTEN	
19	EP19INEN	
18	EP18OUTEN	
17	EP17INEN	
16	EP16OUTEN	
15	EP15INEN	
14	EP14OUTEN	
13	EP13INEN	
12	EP12OUTEN	
11	EP11INEN	
10	EP10OUTEN	
9	EP9INEN	
8	EP8OUTEN	
7	EP7INEN	
6	EP6OUTEN	
5	EP5INEN	
4	EP4OUTEN	
3	EP3INEN	
2	EP2OUTEN	
1	EP1INEN	
0	EP0OUTEN	

0x068 USB_ENDPOINT_READY

Type: Read/Write
Clock: MCLK

USB_ENDPOINT_READY

Bits	Name	Description
31	EP31INREADY	**ALL odd-numbered bits:** SW: RW, HW: R Endpoint<n>InReady. Software must write to this bit to tell the Core whether or not the Endpoint is ready to be transferred. If this bit is 0, and the EP is enabled, then the function controller will return NAKs. If this bit is 1, then the transfer will commence. This bit is ignored if the EP is not enabled.
30	EP30OUTREADY	**ALL even-numbered bits:** SW: RW, HW: R Endpoint<n>OutReady. Software must write to this bit to tell the Core whether or not the Endpoint is ready to be transferred. If this bit is 0, and the EP is enabled, then the function controller will return NAKs. If this bit is 1, then the transfer will commence. This bit is ignored if the EP is not enabled.
29	EP29INREADY	
28	EP28OUTREADY	
27	EP27INREADY	
26	EP26OUTREADY	
25	EP25INREADY	
24	EP24OUTREADY	
23	EP23INREADY	
22	EP22OUTREADY	
21	EP21INREADY	
20	EP20OUTREADY	
19	EP19INREADY	
18	EP18OUTREADY	
17	EP17INREADY	
16	EP16OUTREADY	
15	EP15INREADY	
14	EP14OUTREADY	
13	EP13INREADY	
12	EP12OUTREADY	

USB_ENDPOINT_READY (Continued)

Bits	Name	Description
11	EP11INREADY	
10	EP10OUTREADY	
9	EP9INREADY	
8	EP8OUTREADY	
7	EP7INREADY	
6	EP6OUTREADY	
5	EP5INREADY	
4	EP4OUTREADY	
3	EP3INREADY	
2	EP2OUTREADY	
1	EP1INREADY	
0	EP0OUTREADY	

0x06C USB_F_IMMEDIATE_INT

Type: Read/Write
Clock: MCLK

Informs the function controller that all endpoints that are set here should generate an interrupt immediately upon completion.

In order to reduce the number of interrupts occurring in a frame, all Endpoint<n>InDoneStatus andEndpoint<n>OutDoneStatus are flagged on the SOF. However, if software wishes to see the done status flagged immediately it can by setting the correct bit in this register.

USB_F_IMMEDIATE_INT

Bits	Name	Description
31	IM31ININT	**ALL even-numbered bits:** SW: RW, HW: R Immediate<n>InInterrupt. When SET the Function controller will assert an interrupt immediately instead of waiting until the SOF.
30	IM30OUTINT	**ALL even-numbered bits:** SW: RW, HW: R Immediate<n>OutInterrupt. When SET the Function controller will assert an interrupt immediately instead of waiting until the SOF.
29	IM29ININT	

USB_F_IMMEDIATE_INT (Continued)

Bits	Name	Description
28	IM28OUTINT	
27	IM27ININT	
26	IM26OUTINT	
25	IM25ININT	
24	IM24OUTINT	
23	IM23ININT	
22	IM22OUTINT	
21	IM21ININT	
20	IM20OUTINT	
19	IM19ININT	
18	IM18OUTINT	
17	IM17ININT	
16	IM16OUTINT	
15	IM15ININT	
14	IM14OUTINT	
13	IM13ININT	
12	IM12OUTINT	
11	IM11ININT	
10	IM10OUTINT	
9	IM9ININT	
8	IM8OUTINT	
7	IM7ININT	
6	IM6OUTINT	
5	IM5ININT	
4	IM4OUTINT	
3	IM3ININT	
2	IM2OUTINT	
1	IM1ININT	
0	IM0OUTINT	

0x070 USB_F_ENDPOINT_DONE_STATUS

Type: Read/Write
Clock: MCLK

Asserts which endpoints are done processing. This register indicates which endpoint transfers have been completed and are waiting for further software instructions. The EndpointDoneStatus Register is updated immediately, however it will not assert an interrupt until the SOF marker occurs, unless the corresponding ImmediateInterrupt bit is SET. Software should read this register to determine which endpoints are in need of service. In the case of a host OUT transfer overflow condition the endpoint is placed on the EndpointDoneStatus Register

USB_F_ENDPOINT_DONE_STATUS

Bits	Name	Description
31	EP31INDONE	**ALL odd-numbered bits:** SW: RC, HW: S Endpoint<n>InDoneStatus. When asserted indicates that endpoint <n> IN has transferred all of the data currently set in the Function Endpoint Descriptor.
30	EP30OUTDONE	**ALL even-numbered bits:** SW: RC, HW: S Endpoint<n>OutDoneStatus. When asserted indicates that endpoint <n> OUT has received all of the data currently set in the Function Endpoint Descriptor or the endpoint has received larger than the MaxPacketSize for this endpoint.
29	EP29INDONE	
28	EP28OUTDONE	
27	EP27INDONE	
26	EP26OUTDONE	
25	EP25INDONE	
24	EP24OUTDONE	
23	EP23INDONE	
22	EP22OUTDONE	
21	EP21INDONE	
20	EP20OUTDONE	
19	EP19INDONE	
18	EP18OUTDONE	
17	EP17INDONE	
16	EP16OUTDONE	

USB_F_ENDPOINT_DONE_STATUS (Continued)

Bits	Name	Description
15	EP15INDONE	
14	EP14OUTDONE	
13	EP13INDONE	
12	EP12OUTDONE	
11	EP11INDONE	
10	EP10OUTDONE	
9	EP9INDONE	
8	EP8OUTDONE	
7	EP7INDONE	
6	EP6OUTDONE	
5	EP5INDONE	
4	EP4OUTDONE	
3	EP3INDONE	
2	EP2OUTDONE	
1	EP1INDONE	
0	EP0OUTDONE	

0x074 USB_ENDPOINT_DONE_ENA

Type: Read/Write
Clock: MCLK

Endpoints done mask. This register indicates which bit in the EndpointDoneStatus Register will generate an interrupt when set. This applies to interrupts generated either immediately or on the SOF marker.

USB_ENDPOINT_DONE_ENA

Bits	Name	Description
31	EP31INDNEN	**ALL odd-numbered bits:** SW: RW, HW: R Endpoint<n>InDoneEnable. When SET allows endpoint <n> IN direction to generate done interrupts. The Endpoint<n>InDoneEnable must be SET for both immediate and SOF marker interrupts.
30	EP30OUTDNEN	**ALL even-numbered bits:** SW: RW, HW: R Endpoint<n>OutDoneEnable. When SET allows endpoint <n> IN direction to generate done interrupts. The Endpoint<n>OutDoneEnable must be SET for both immediate and SOF marker interrupts.
29	EP29INDNEN	
28	EP28OUTDNEN	
27	EP27INDNEN	
26	EP26OUTDNEN	
25	EP25INDNEN	
24	EP24OUTDNEN	
23	EP23INDNEN	
22	EP22OUTDNEN	
21	EP21INDNEN	
20	EP20OUTDNEN	
19	EP19INDNEN	
18	EP18OUTDNEN	
17	EP17INDNEN	
16	EP16OUTDNEN	
15	EP15INDNEN	
14	EP14OUTDNEN	
13	EP13INDNEN	

USB_ENDPOINT_DONE_ENA (Continued)

Bits	Name	Description
12	EP12OUTDNEN	
11	EP11INDNEN	
10	EP10OUTDNEN	
9	EP9INDNEN	
8	EP8OUTDNEN	
7	EP7INDNEN	
6	EP6OUTDNEN	
5	EP5INDNEN	
4	EP4OUTDNEN	
3	EP3INDNEN	
2	EP2OUTDNEN	
1	EP1INDNEN	
0	EP0OUTDNEN	

0x078 USB_ENDPOINT_TOGGLE_BITS

Type: Read/Write
Clock: MCLK

The current toggle bit for the endpoint. On all transfers the PID of data packets alternate between DATA0 and DATA1 to ensure the ordering of the packets. The data toggle must continue across multiple transfers and are different for each physical endpoint. Each control transfer must begin from the host with DATA0.

USB_ENDPOINT_TOGGLE_BITS

Bits	Name	Description
31	EP31INTOG	**ALL odd-numbered bits:** SW: RW, HW: RW Endpoint<n>InToggleBit. The expected value of the current data PID is determined by the hardware from the last PID from the host for that endpoint. Software should change this bit while the EP is currently full or disabled if they wish to change the next data toggle bit.
30	EP30OUTTOG	**ALL odd-numbered bits:** SW: RW, HW: RW Endpoint<n>OutToggleBit. The value of the next data PID to be sent by hardware is determined by this bit. This bit is updated by Hardware after each successful transmission. Software should change this bit while the EP is currently empty or disabled if they wish to change the next data toggle bit.
29	EP29INTOG	
28	EP28OUTTOG	
27	EP27INTOG	
26	EP26OUTTOG	
25	EP25INTOG	
24	EP24OUTTOG	
23	EP23INTOG	
22	EP22OUTTOG	
21	EP21INTOG	
20	EP20OUTTOG	
19	EP19INTOG	
18	EP18OUTTOG	
17	EP17INTOG	
16	EP16OUTTOG	
15	EP15INTOG	

USB_ENDPOINT_TOGGLE_BITS (Continued)

Bits	Name	Description
14	EP14OUTTOG	
13	EP13INTOG	
12	EP12OUTTOG	
11	EP11INTOG	
10	EP10OUTTOG	
9	EP9INTOG	
8	EP8OUTTOG	
7	EP7INTOG	
6	EP6OUTTOG	
5	EP5INTOG	
4	EP4OUTTOG	
3	EP3INTOG	
2	EP2OUTTOG	
1	EP1INTOG	
0	EP0OUTTOG	

0x07C USB_F_FRAME_NUMBER

Type: Read
Clock: MCLK

The number of the last Start Of Frame. Reading this returns the frame number of the last successfully received SOF. The frame number is returned with the least significant byte (LSB) first.

Bits	Name	Description
31:11	RESERVED_BIT31_11	
10:0	FRMNUMB	SW: R, HW: W LastFrameNumber. Hardware will set this value based upon the FrameNumber contained in the last received SOF packet.

0x07C USB_ENDPOINT_READY_CLR

Type: Write
Clock: MCLK

Bits	Name	Description
31:0	EPRDYCLEAR	Setting (1) this register clears the corresponding Endpoint Ready bit but does not affect the Frame Number. This is a workaround to fix a register contention condition in the Core. Since address space was not available, the Frame Number register was multiplexed with this functionality.

19.1.4 DMA registers

0x0800 USB_DMA_REVISION

Type: Read/Write
Clock: MCLK

The DmaRevision Register contains the revision code for the DMA Core block.

Bits	Name	Description
31:8	RESERVED_BITS31_8	
7:0	REVCODE	SW: R, HW: R RevisionCode. Revision code for DMAIP block design.

0x0804 USB_DMA_INT_STATUS

Type: Read/Write
Clock: MCLK

EP and ETD interrupt for erroneous transfers. The DmaInterruptStatus Register reports the interrupt status for the DMAIP.

USB_DMA_INT_STATUS

Bits	Name	Description
31:2	RESERVED_BITS31_2	

USB_DMA_INT_STATUS (Continued)

Bits	Name	Description
1	EPERR	SW: RW, HW: W EndpointErrorInterrupt. EP DMA transfer aborted on error (read 0x0810 to find out which EP client). Writing a 1 clears the interrupt.
0	ETDERR	SW: RW, HW: W ETDErrorInterrupt. ETD DMA transfer aborted on error (read 0x080c to find out which ETD client). Writing a 1 clears the interrupt.

0x0808 USB_DMA_INT_ENA

Type: Read/Write
Clock: MCLK

Enable or mask DMA Interrupts. The DmaInterruptEnable Register is used to enable the corresponding status bits in rIntStat to assert the MAIP Interrupt.

Bits	Name	Description
31:2	RESERVED_BITS31_2	
1	EPERRINTEN	SW: RW, HW: R EndpointErrorInterruptEnable. 1 = enable the interrupt, 0 = mask the interrupt
0	ETDERRINTEN	SW: RW, HW: R ETDErrorInterruptEnable. 1 = enable the interrupt, 0 = mask the interrupt

0x080c USB_ETD_DMA_ERROR_STATUS

Type: Read/Write
Clock: MCLK

The EtdDmaErrorStatus Register identifies which ETD client DMA transfer was aborted on an Error Response.

Bits	Name	Description
31:0	ETDDMAERST	SW: RW, HW: W ETDDMAErrorStatus. Indicates the Error status from each client. Writing 1 clears the corresponding status bit

0x0810 USB_EP_DMA_ERROR_STATUS

Type: Read/Write
Clock: MCLK

The EpDmaErrorStatus Register identifies which EP client DMA transfer was aborted on an Error Response.

Bits	Name	Description
31:0	EPDMAERRST	SW: RW, HW: W EndpointDMAErrorStatus. Reports the Error status from each client. Writing 1 clears the corresponding status bit

0x0820 USB_ETD_DMA_ENA

Type: Read/Write
Clock: MCLK

The EtdDmaEnable Register enables the DMA transfer for the corresponding ETD client.

Bits	Name	Description
31:0	ETDDMAENB	SW: RW, HW: R ETDDMAEnable. Setting the ETDDmaEnable bit will enable DMA transfer on the corresponding ETD. Will be auto-disabled at the end of the transfer.

0x0824 USB_EP_DMA_ENA

Type: Read/Write
Clock: MCLK

The EpDmaEnable Register enables the DMA transfer for the corresponding EP client.

Bits	Name	Description
31:0	EPDMAENB	SW: RW, HW: R EndpointDMAEnable. Setting the EndpointDMAEnble bit will enable DMA transfer on the corresponding EP. Will be auto-disabled at the end of the transfer.

0x0828 USB_ETD_DMA_ENA_X_TRIGER_REQ

Type: Read/Write
Clock: MCLK

Writing to the EtdDmaEnableXTrigerRequest Register results in two things: it sets the ETD DMA enable bit AND causes an XTrigger Request from the ETD client.

Bits	Name	Description
31:0	ETDDMAXTEN	SW: RW, HW: R ETDDMAXTriggerEnable. Writing a 1 sets the corresponding ETDDMAEnable bit and causes an external XTrigger for the client

0x082c USB_EP_DMA_ENA_X_TRIGER_REQ

Type: Read/Write
Clock: MCLK

Writing to the EpDmaEnableXTrigerRequest Register address results in two things: it sets the EP DMA enable bit AND fakes an XTrigger Request from the EP client.

Bits	Name	Description
31:0	EPDMAXTEN	SW: RW, HW: R EndpointDMAXTriggerEnable. Writing a 1 sets the corresponding EndpointDMAEnable bit and causes an external XTrigger for the client

0x0830 USB_ETD_DMA_ENB_XY_TRIG_REQ

Type: Read/Write
Clock: MCLK

The EtdDmaEnableXYTrigerRequest Register functionality is the same as the EtdDmaEnableXTrigerRequest Register, except that it causes both XTrigger and YTrigger requests from the ETD client.

Bits	Name	Description
31:0	ETDDMAENXYT	SW: RW, HW: R ETDDMAXAndYTriggerEnable. Writing 1 sets the corresponding ETDDMAEnable and causes both XTrigger and YTrigger for the client

0x0834 USB_EP_DMA_ENA_XY_TRIGER_REQ

Type: Read/Write
Clock:

The EpDmaEnableXYTrigerRequest Register functionality is the same as the EpDmaEnableXTrigerRequest Register, except that it causes both XTrigger and YTrigger Requests from the EP client.

Bits	Name	Description
31:0	EPDMAENXYT	SW: RW, HW: R EndpointDMAXAndYTriggerEnable. Writing a 1 sets the corresponding EndpointDMAEnable bit and causes both XTrigger and YTrigger for the client

0x0838 USB_ETD_DMA_BURST4_ENA

Type: Read/Write
Clock: MCLK

The EtdDmaBurst4Enable Register is set by default for all clients. The DMA will transfer words on the AHB using the INCR4 transfer size (burst of 4 DWORDS). If for some reason a source/destination of the DMA transfer cannot handle bursts of 4 DWORDS, the corresponding bit should be cleared to transfer one word at a time.

Bits	Name	Description
31:0	ETDDMABST4EN	SW: RW, HW: R ETDDMABurst4Enable. Writing a 1 sets the corresponding ETDDMAEnable and causes both XTrigger and YTrigger for the client

0x083c USB_EP_DMA_BURST4_ENA

Type: Read/Write
Clock: MCLK

The EpDmaBurst4Enable Register is set by default for all clients. The DMA will transfer words on the AHB using the INCR4 transfer size (burst of 4 DWORDS). If for some reason a source/destination of the DMA transfer cannot handle bursts of 4 DWORDS, the corresponding bit should be cleared to transfer one word at a time.

Bits	Name	Description
31:0	ETDDMABST4EN	SW: RW, HW: R ETDDMABurst4Enable. Writing a 1 sets the corresponding EndpointDMAEnable bit and causes both XTrigger and YTrigger for the client

0x0840 USB_MISC_CTL

Type: Read/Write
Clock: MCLK

The MiscControl register enables certain modes that may make development easier. Then enabling of the modes is application dependent.

Bits	Name	Description
3	ISOPREVFRM	SW: RW, HW: R ISO OUT Previous Frame Mode. If set, the DMA will only service an ISO OUT ETD one frame before the transfer will start. The frame number where the transfer will occur can be found in the StartingFrame field of the Host Isochronous ETD descriptor format.
2	SKPRTRY	SW: RW, HW: R SkipOnRetry Mode. If set, will skip current DMA transfer if an AHB RETRY response is encountered, and will start next pending transfer. The transfer will be retried after the rest of the pending transfers have been completed.
1	ARBMODE	SW: RW, HW: R Arbiter Mode Select puts the DMA arbiter in the following modes: 1: Round Robin mode (The arbitration for the DMA servicing will proceed in a round robin fashion.) 0: Priority Access mode (The DMA enabled transfers are serviced in a prioritized fashion in order of decreasing ETD/EP number.Zero gets highest priority.)
0	FILTCC	SW: RW, HW: R Filter On Completion Code. If set, the DMA controller will check the completion code of the transfer. The data will only be transferred if the completion code shows no error.

0x848 USB_ETD_DMA_CHANNEL_CLEAR

Type: Read/Write
Clock: MCLK

This register will clear the DMA channel corresponding to whichever bit is set and also disable the DMA transfer. It can be used to abort a DMA transfer.

Bits	Name	Description
31:0	ETDDMACHANLCLR	SW: W, HW: R EtdDmaChannelClear. Will clear the DMA channel corresponding to the bit that is set. Will also clear the DMA enable bit for the ETD. This register is a Write- Clear register. Software always reads 0 on this register.

0x84C USB_EP_DMA_CHANNEL_CLEAR

Type: Read/Write
Clock: MCLK

This register will clear the DMA channel corresponding to whichever bit is set and also disable the DMA transfer. It can be used to abort a DMA transfer.

Bits	Name	Description
31:0	EPDMACHANLCLR	SW: W, HW: R EpDmaChannelClear. Will clear the DMA channel corresponding to the bit that is set. Will also clear the DMA enable bit for the EP. This register is a Write- Clear register. Software always reads 0 on this register.

0x900+4n USB_ETDn_SYS_MEM_START_ADDR, n=[0..31]

Type: Read/Write
Clock: MCLK

Starting system memory address location where data will be put/fetched for ETD<n>. Before Enabling DMA for a client ETD, the corresponding register must be loaded with the ETDSystemMemoryStartAddress where the data will be stored. The ETDSystemMemoryStartAddress are byte addresses that must be DWORD aligned.

Bits	Name	Description
31:0	ETDSMSA	SW: RW, HW: R ETDSystemMemoryStartAddress. Starting address in system memory where DMA will put/fetch data to for ETD<n>.

0x980+4n USB_EPn_SYS_MEM_START_ADDR, n=[0..31]

Type: Read/Write
Clock: MCLK

Before enabling the DMA for a client EP, the corresponding register must be loaded with the system memory address where the data will be stored. The EndpointSystemMemoryStartAddress are byte addresses that must be DWORD aligned.

Bits	Name	Description
31	EPSMSA	SW: RW, HW: R EndpointSystemMemoryStartAddress. Starting address in system memory where DMA will put/fetch data to for EP n.

0x0a00+4n USB_ETDn_DMA_BUFFER_XFER_PTR, n=[0..31]

Type: Read/Write
Clock: MCLK

If the DMA is paused due to the encounter of an AHB RETRY response and the SKIPONRETRY feature is enabled, the current index into the buffer is stored in these registers so that the DMA knows where to re-start the transfer. The buffer transfer pointers are accessible only for debug purposes.

Bits	Name	Description
31:0	ETDDMABUFPTR	SW: RW, HW: R ETDDMABufferTransferPointer. Stores the current index of the buffer when AHB RETRY response is asserted.

0x0a80+4n USB_EPn_DMA_BUFFER_XFER_PTR, n=[0..31]

Type: Read/Write
Clock: MCLK

If the DMA is paused due to the encounter of an AHB RETRY response and the SKIPONRETRY feature is enabled, the current index into the buffer is stored in these registers so that the DMA knows where to re-start the transfer. The buffer transfer pointers are accessible only for debug purposes.

Bits	Name	Description
31:0	EPDMABUFPTR	SW: RW, HW: R EPDMABufferTransferPointer. Stores the current index of the buffer when AHB RETRY response is asserted.

20 GSM Core

20.1 Overview

This chapter describes the registers in the GSM core (GSM_CORE).

20.2 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: GSM
- Number of word addresses: 64
- Byte Address: 0x7100 - 0x71FC
- MODULE OFFSET = CHIP_BASE+0x7100
- MAX = CHIP_BASE+0x71FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

20.2.1 GSM clock registers

The GSM registers in this section are clocked by GSM_CLK.

0x0004 MICRO_GSM_TIME_RD

Type: Read
Clock: GSM_CLK
Reset State: 0x0

The MICRO_GSM_TIME_RD register gives the microprocessor the ability to read system time.

Bits	Name	Description
18	ODD_FRAME	This bit indicates to the microprocessor whether the current frame is odd or even. 0 : even frame 1 : odd frame
17:6	SYMBOL_TIME	This field provides the symbol time to the microprocessor. This field is updated every symbol.
5:4	Q_SYMBOL_TIME	This field provides the quarter symbol time to the microprocessor. This field is updated every quarter symbol.
3:0	PRESCALE_TIME	This field provides the prescale_time to the microprocessor. This field is updated every gsm_clk pulse.

0x0008 GO_TO_SLEEP_CMD

Type: Write
Clock: GSM_CLK
Reset State: 0x 0

The GO_TO_SLEEP_CMD register is the sleep command register.

Bits	Name	Description
0	GEN_GO_TO_SLEEP	When this bit is set (1), the go_to_sleep signal is pulsed to the sleep controller (aligned with the next pp_frame). This bit is reset by pp_frame.

0x000C MICRO_IRQ_CONFIG_RW

Type: Read/Write
Clock: GSM_CLK
Reset State: 0x X

The MICRO_IRQ_CONFIG_RW register gives the microprocessor the ability to program time-based interrupts.

Bits	Name	Description
23:20	MICRO_IRQ_MSB_IGNORE	The interrupt request signal occurs (if enabled) when gsm_time[17:0] is equal to micro_irq0_time[17:0]. However, this field can be used to force the most significant bits of the comparison to be ignored with the result that the interrupt occurs more frequently. Field **Value** **Effect of the comparison** 0x0 Ignore no bits (use all bits). 0x1 Ignore MSB (bit 17) 0x2 Ignore the 2 MSBs ... 0xF Ignore the 15 MSBs
19	MIRCO_IRQ_ENA	When this bit is set (1), the interrupt request signal is enabled.
18:0	MICRO_IRQ_TIME	This field contains the value to be compared with the gsm_time. If the comparison is true, irq_value is set (1). micro_irq_time[18] : odd_frame(gsm_time[18]) micro_irq_time[17:6] : symbol_time (gsm_time[17:6]) micro_irq_time[5:4] : q_symbol_time (gsm_time[5:4]) micro_irq_time[3:0] : prescale_time (gsm_time[3:0])

0x0010 VFR_IRQ_ALIGN_CMD

Type: Write
Clock: GSM_CLK
Reset State: 0x 0

The VFR_IRQ_ALIGN_CMD register is the alignment command register for the vocoder frame reference (VFR) interface. The GSTMR contains an independent counter to generate the 20 ms vfr_irq. After the gsm_time counter has been adjusted, the vfr_irq counter must be realigned with the gsm_time_counter.

Bits	Name	Description
0	VRF_IRQ_FRAME_ALIGN	The microprocessor commands the realignment by setting (1) this field. At the next frame boundary, a pulse is forced on the vfr_irq signal and the vfr_irq counter is reset to 0. Only one realignment occurs per command. Note that 20 ms vocoder frame boundaries align with 4.615 ms TDMA frame boundaries only after frames 12 and 25 of the traffic superframe. Consequently, the microprocessor must perform the command during one of these two frames.

0x0014 GSM_DSP_WR_CLK_CTL

Type: Read/Write
Clock: GSM_CLK
Reset State: 0x 0

The GSM_DSP_WR_CLK_CTL register can override the write clock used by GSM-clocked registers.

Bits	Name	Description
0	GSM_DSP_WR_CLK_OVERRIDE	When this bit is set (1), gsm_dsp_wr_clk is overridden with the non-gated gsm_clk.

0x0018 GSM_TIME_TC_ADJ_CMD

Type: Write
Clock: GSM_CLK
Reset State: 0x 0

The GSM_TIME_TC_ADJ_CMD register gives the microprocessor the ability to override the terminal count of system time for one TDMA frame. This register must be written to every frame the override is desired.

Bits	Name	Description
19	ORIDE_TC	When set (1), gsm_time_tc_value is used as terminal count by the gsm_time counter. Once the terminal count has been reached, this bit is reset to '0' and the default value will be used for the terminal count.
18	UNUSED_BIT18	
17:0	GSM_TIME_TC_VALUE	This field supplies the terminal count for gsm_time if oride_tc is set (1).

0x001C GSM_CORE_MICRO_RESET_REG

Type: Read/Write
Clock: GSM_CLK
Reset State: 0x 7

GSM_CORE_MICRO_RESET_REG is the microprocessor reset read/write register. The microprocessor asserts resets by setting (1) the appropriate bit. Once set, the reset stays asserted until it is de-asserted by the microprocessor. The microprocessor de-asserts the reset by clearing (0) the appropriate bit.

Bits	Name	Description
3	RESET_GSACCLK	Reset for hardware that is operating on the GSAC clock.
2	RESET_VDSPCLK	Reset for hardware that is operating on the VDSP clock.
1	RESET_MDSPCLK	Reset for hardware that is operating on the MDSP clock.
0	RESET_GSMCLK	Reset for hardware that is operating on the GSM clock.

0x0020 GSM_TIME_LOAD_CTL_CMD

Type: Write
Clock: GSM_CLK
Reset State: 0x 0

The GSM_TIME_LOAD_CTL_CMD register gives the microprocessor the ability to override the initial count of system time for a particular TDMA frame. This register must be written during the preceding frame for every new initial count.

Bits	Name	Description
19	GSM_TIME_CNT_SEL	Setting (1) this bit causes a one-time transfer of gsm_time_load_value to the gsm_time counter at the next frame boundary.
18:0	GSM_TIME_LOAD_VALUE	This bit field is the value to be loaded into gsm_time counter. See bit 19 above.

0x0024 ENABLE_GSTMR_DURING_SLEEP

Type: Read/Write
Clock: GSM_CLK
Reset State: 0x 0

The ENABLE_GSTMR_DURING_SLEEP register is used as an override of the sleep controller.

Bits	Name	Description
0	GSTMR_DURING_SLEEP_ENA	When this bit is set (1), the GSTMR remains operational during the sleep mode. When this bit is cleared (0), the GSTMR is disabled during sleep. So, if this bit is set before the beginning of the sleep period, the GSMTR continues to operate until this bit has been cleared (0) or when the end of the sleep period has occurred, whichever comes first. The latter case automatically resets this bit. This bit can also be set during sleep, thus “waking up” the GSTMR.

20.2.2 GSAC registers

The GSM ARM registers in this section are clocked by MDSP_CLK.

0x034+0x30n AMBA_KEY_STRMn_BLK1_WORD0_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK1_WORD0_RD register provides the first group of 4 octets of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (31:0) of the block 1 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[31:0]	First group of 4 octets of the GPRS key stream

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK1_RD[31:0]	Bits 31 down to 0 of the key stream, belonging to block 1

0x038+0x30n AMBA_KEY_STRMn_BLK1_WORD1_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK1_WORD1_RD register provides the second group of 4 octets of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (63:32) of the block 1 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[63:32]	Second group of 4 octets of the GPRS key stream.

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK1_RD[63:32]	Bits 63 down to 32 of the key stream, belonging to block 1

0x03C+0x30n AMBA_KEY_STRMn_BLK1_WORD2_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK1_WORD2_RD register provides the third group of 4 octet of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (95:64) of the block 1 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[95:64]	Third group of 4 octets of the GPRS key stream.

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK1_RD[95:64]	Bits 95 down to 64 of the key stream, belonging to block 1

0x040+0x30n AMBA_KEY_STRMn_BLK1_WORD3_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK1_WORD3_RD register provides the fourth group of 4 octet of GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (113:96) of the block 1 key stream and status.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[127:96]	Fourth group of 4 octets of the GPRS key stream

Non-GPRS mode

Bits	Name	Description
31	GSAC_ACTIVE	GSAC active status bit
30:18	UNUSED_BITS30_18	
17:0	AMBA_KEY_STREAM_BLOCK1_RD[113:96]	Bits 113 down to 96 of the key stream, belonging to block 1

0x044+0x30n AMBA_KEY_STRMn_BLK2_WORD0_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK2_WORD0_RD register provides the fifth group of 4 octets of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (31:0) of the block 2 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[159:128]	Fifth group of octets of the GPRS key stream.

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK2_RD[31:0]	Bits 31 down to 0 of the key stream, belonging to block 2

0x048+0x30n AMBA_KEY_STRMn_BLK2_WORD1_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK2_WORD1_RD register provides the sixth group of 4 octets of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal).

In the non-GPRS mode, this register provides bits (63:32) of the block 2 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[191:160]	Sixth group of 4 octets of the GPRS key stream

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK2_RD[63:32]	Bits 63 down to 32 of the key stream, belonging to block 2

0x04C+0x30n AMBA_KEY_STRMn_BLK2_WORD2_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK2_WORD2_RD register provides the seventh group of 4 octets of the GPRS key stream for GSAC0. The GSAC0_CONFIG_0 register controls the formatting of these octets (endianess and/or bit-reversal)

In the non-GPRS mode, this register provides bits (95:64) of the block 2 generated key stream.

GPRS mode

Bits	Name	Description
31:0	GPRS_KEY_STREAM_RD[223:192]	Seventh group of 4 octets of the GPRS key stream.

Non-GPRS mode

Bits	Name	Description
31:0	AMBA_KEY_STREAM_BLOCK2_RD[95:64]	Bits 95 down to 64 of the key stream, belonging to block 2

0x050+0x30n AMBA_KEY_STRMn_BLK2_WORD3_RD, n=[0,1]

Type: Read
Clock: MDSP_CLK
Reset State: 0x X

In the GPRS mode, the AMBA_KEY_STRM0_BLK2_WORD3_RD register provides status for GPRS operation.

In the non-GPRS mode, this register provides bits (113:96) of the block 2 generated key stream and status.

GPRS mode

Bits	Name	Description
6:0	GPRS_STATUS	Bit (6) -> capture_word6_pending Bit (5) -> capture_word5_pending Bit (4) -> capture_word4_pending Bit (3) -> capture_word3_pending Bit (2) -> capture_word2_pending Bit (1) -> capture_word1_pending Bit (0) -> capture_word0_pending

Non-GPRS mode

Bits	Name	Description
31	GSAC_ACTIVE	GSAC active status bit
30:18	UNUSED_BITS30_18	
17:0	AMBA_KEY_STREAM_BLOCK2_RD[113:96)	Bits 113 down to 96 of the key stream, belonging to block 2

0x024+0x30n GSACn_CONFIG_0, n=[0,1]

Type: Read/Write
Clock: MDSP_CLK
Reset State: 0x 0

Registers GSAC0_CONFIG_ 0 and GSAC1_CONFIG_ 0 give the ARM the capability to configure their corresponding GSAC. Their bits are identical.

GSACn_CONFIG_0

Bits	Name	Description
13	BIG_ENDIAN	When this bit is set (1), the output words in the GPRS mode are output in a big endian format. When this bit cleared (0), the output words are output in little endian format. This bit affects only how the data is passed to the ARM, not how it is received and processed by the GSAC0
12:8	PREPEND_X_ZEROS	These bits allow the user to specify x number of zeros to prepend to the GPRS keystream word. This allows the software to generate keystreams with alignments that match the messages that they are ciphering.
7	GSAC_INIT_REG0	This field is bit 0 of the GSAC initialization register for GEAx (GPRS) algorithms
6	SHIFT_INTO_MSB_ENA	When this bit is set (1), keystream shifting occurs into the MSB. When this bit is cleared (0), keystream shifting occurs into the LSB.
5	GPRS_KEY_STREAM_GEN_CMD	When this bit is set (1), it generates a pulse that is used to start generation of the next 114 bits of the keystream. Since this is a one-shot pulse, the read back values always return 0.
4	GPRS_AUTOGEN_ENA	When this bit is set (1), a read made to one of the keystream registers by the ARM automatically generates new cipher keystream bits into the specific double word which has been read.
3:2	GSAC_MODE	For each bit, GSAC0 generates a key stream compliant to: 00 : A5/2 01 : A5/1 10 : GEA2 11 : GEA1

GSACn_CONFIG_0 (Continued)

Bits	Name	Description
1	GSAC_ARM	When this bit is set (1), the GSAC starts computing a new key stream upon the next occurrence of pp_frame. This start of computation is only performed at the next pp_frame event. It must be set (no need to clear it to '0' between) before each pp_frame that a key stream is to be calculated.
0	GSAC_GO	When this bit is set (1), the GSAC immediately starts the calculation of a new key stream. This bit does not need to be cleared, as the "start" is generated as a result of setting the bit. There is no need to clear (0) the bit between sets.

21 Peripherals

21.1 ARM registers

The ARM peripherals registers are separated into two sections according to their respective clocks:

- MICRO_HCLK registers (See 21.1.1 MICRO_HCLK registers on page 21-2):
  - Offset = CHIP_BASE+0x0700, MAX=CHIP_BASE+0x07FC
- TCX04_CLK registers (See 21.1.2 TCXO4 clock registers on page 21-21):
  - Offset = CHIP_BASE+0x0800, MAX=CHIP_BASE+0x08FC

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: PERIPH
- Number of word addresses: 192
- Byte Address: 0x0700 - 0x09FC (total area)
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

21.1.1 MICRO_HCLK registers

The registers in this section are clocked by MICRO_HCLK.

21.1.1.1 System registers

0x0004 CHIP_MODE

Type: Read
Clock: HCLK

Bits	Name	Description
2	MODE2_PIN	Returns the current value of MODE2 pin.
1	MODE1_PIN	Returns the current value of MODE1 pin.
0	MODE0_PIN	Returns the current value of MODE0 pin.

0x0008 SYSTEM_MODE

Type: Write
Clock: HCLK
Reset State: 0x0

This register determines the system mode of the MSM.

SYSTEM_MODE

Bits	Name	Description
25	PA_RANGE_SEL	This bit controls the primary selection for the PA_RANGE1 and PA_RANGE0 pins. 0 : CDMA 1 : HDR
24	RESERVED_BIT24	
23	PDM_SEL	This bit controls the selection of PDM data. Note that the PDM function is an alternate choice for these pins: PA_RANGE1, PA_RANGE0, and GPIO92. To select PDM data onto PA_RANGE1:0, see register definition for WEB_MISC[8:7]. 0 : CDMA 1 : HDR
22	TD_SEL	This bit selects the input to the Turbo Decoder. 0 : CDMA 1 : HDR

SYSTEM_MODE (Continued)

Bits	Name	Description
21:20	PA_POWERUP_MODE	These bits control inputs to the PA_ON2 equation. 00 : CDMA 01 : HDR 10 : undefined
19:18	TX_PUNCT_SEL	These bits control inputs to the TX_ON equation. 00 : CDMA 01 : HDR 10 : undefined
17:16	PA_PUNCT_SEL	These bits control inputs to the PA_ON equation. 00 : CDMA 01 : HDR 10 : undefined
15	SLEEP_TXDAC_SEL	This bit selects the input to the sleep_txdac signal, which is sent to the txdac block. 0 : MISC_CLK_CTL[0] 1 : GSM
14:13	TXDAC_SEL	Selects which block feeds txdac_clk, txdac_idin and txdac_qdin to the txdac. 00 : CDMA1X or HDR 01 : undefined 10 : GSM
12:11	UNUSED_BITS_12_11	
10	TX_IQCLK_SEL	This bit was formerly defined as TX_IQCLK_SEL, but is UNUSED.
9:8	TRK_LO_ADJ_SEL	This bit selects the function for the dedicated TRK_LO_ADJ pin. 0 : CDMA1X 1 : HDR 2 : undefined
7:6	TX_AGC_ADJ_SEL	This bit selects the function for the dedicated TX_AGC_ADJ pin. 0 : CDMA1X 1 : HDR 2 : undefined
5	UNUSED_BIT5	This bit was formerly defined as SLEEP_N_SEL, but is UNUSED.
4:3	TX_ON_SEL	Selects CDMA1X/GSM control of dedicated TX_ON pin. 00 : mod_tx_on 01 : grfc10 10 : grfc1_chooses_grfc10

SYSTEM_MODE (Continued)

Bits	Name	Description
2	GSM_MODE	This bit, combined with bit 0, feeds into the EXT_VOC_FR irq to the ADSP. When clear (0), the vocoder frame reference irq comes from either external pin or CDMA. When set (1), GSM_MODE.
1	UNUSED_BIT1	This bit was formerly defined as GPS_MODE, but is UNUSED.
0	UNUSED_BIT0	This bit was formerly defined as CDMA_FM_SEL, but is UNUSED.

21.1.1.2 Peripherals miscellaneous registers

0x0010 WEB_MISC_WR

Type: Write
Clock: HCLK
Reset State: 0xx888

WEB_MISC_WR

Bits	Name	Description
15	TX_AGC_CTL_SEL	When reset to 0, tx_agc is controlled by chain 0 rx_agc (default). When set to 1, tx_agc is controlled by chain 1 rx_agc.
14	RXF0_ANT_SEL	When reset to 0, rx_front0 uses chain0 antenna inputs (default). When set to 1, rx_front0 uses chain1 antenna inputs The purpose of providing this choice is to allow receive chain1 to be calibrated, using rx_front0 path.
13	DEBUG_BUS_EBI2_EN	When this bit is clear (0), dsp_dbg_bus is not selected onto ebi2 (default). When this bit is set (1), enable dsp_dbg_bus onto ebi2, as determined by WEB_MISC_WR/debug_bus_sel.
12	BT_SLEEP_STOP_CMD	Cannot be reset. Setting this bit (1) stops BlueTooth from sleeping so that some operations don't need to wait until BlueTooth wakeup.
11	UART_NAND_SEL	(Only writable) This bit selects between the UART and the boot sequencer as the driver of the output pin, gpio95. When set (1), this bit allows the boot sequencer to report the boot-from-NAND-flash status to outside of MSM through the gpio95 pin. When cleared (0), this bit allows the UART to use the gpio95 pin as the dp_tx_data data output, although the GPIO_CFG register will still have to be programmed. Note: This bit only has an effect when the pin boot_mode = 1 (boot-up from NAND flash). When boot_mode = 0, the gpio95 pin can be used by the UART as the dp_tx_data data output, regardless of the value of this bit. The power-on value of this bit is 1.

WEB_MISC_WR (Continued)

Bits	Name	Description
10:9	ISDS_RTL_SEL	Controls jtag pin function during ISDS/RTL mode: ‘00’: MSM Jtag ‘01’: MDSP RTL ‘10’: ADSP RTL The ‘11’ encoding is reserved.
8	PDM_TO_PA_RANGE1	Set this bit to 1 to drive PDM onto PA_RANGE1 pad. Default is 0.
7	PDM_TO_PA_RANGE0	Set this bit to 1 to drive PDM onto PA_RANGE0 pad. Default is 0
6	CHIPID_PULL_ALWAYS_ON	When this bit is set, the chipid_pad pulls are active, regardless of chip reset state or padsig connection.
5	RAM_REDUNDANCY_TEST	
4	TXDAC_TEST_SEL	TX DAC test select: 0: from internal logic 1: from pins
3	PA_ON_GUARD	PA_ON protection
2:0	DEBUG_BUS_SEL	The purpose of these bits is to control the muxing of ADSP debug, MDSP debug, and general debug onto the EBI2 bus. Value Function ’000’ : mdsp_dbg_bus0 ’001’ : mdsp_dbg_bus0, with shadow of chip_dbg_bus on bits 15:0 ’010’ : mdsp_dbg_bus1 ’011’ : mdsp_dbg_bus1, with shadow of chip_dbg_bus on bits15:0 ’100’ : adsp_dbg_bus0 ’101’ : adsp_dbg_bus0, with shadow of chip_dbg_bus on bits15:0 ’110’ : adsp_dbg_bus1 ’111’ : adsp_dbg_bus1, with shadow of chip_dbg_bus on bits15:0

0x0014 WEB_MISC2_WR

Type: Write
Clock: HCLK
Reset State: 0x10000000

WEB_MISC2_WR

Bits	Name	Description
31	TX_ON_ALT_SEL	When set (1), GRFC14 is driven out from pin TX_ON. Default: 0
30	PACTL_BG2_EN	When set (1), the second bandgap reference is enabled. Default: 0
29	PACTL_OUTNODE_ENA	Add/remove pull down at the output of PADAC. Default: 0

WEB_MISC2_WR (Continued)

Bits	Name	Description
28	PACTL_GRFC20_ENA	Selects to control for the following PACTL DAC enables: PACTL_BG_ENA PACTL_BG2_ENA PACTL_DECOPA_ENA PACTL_OUTOPA_ENA PACTL_DAC_ENA PACTL_DAC2_ENA When clear (0), the respective bits of the MISC_CTL register control the PACTL_DAC enables. When set (1), PACTL DAC enables are controlled by GRFC20. When GRFC20 is ‘0’, the enables are ‘1’, and when GRFC20 is ‘1’, the enables are ‘0’.) Default : 1
27	PACTL_BG_ENA	PACTL DAC Bandgap Enable. Default: 0
26	PACTL_DECOPA_ENA	PACTL DAC Decoupling Amp Enable. Default: 0
25	PACTL_DAC_ENA	PACTL DAC Enable. Default: 0
24	PACTL_OUTOPA_ENA	PACTL DAC Output Amp Enable. Default: 0
23	PACTL_BGOUT_SEL	PACTL DAC Connects Bandgap to output. Default: 0
22	PACTL_DCOPAOUT_SEL	PACTL DAC Decoupling Amp to output. Default: 0
21	PACTL_DACBOUT_SEL	PACTL DAC Connects Vdacb to output. Default: 0
20	PACTL_DACOUT_SEL	PACTL DAC connects Vdac to output. Default: 0
19	PACTL_REFIN_SEL	PACTL DAC Bandgap Ref. Select 1: Testing pin (HKADC6) is used to supply bandgap reference. 0: PACTL_DAC’s internal bandgap circuit is used to supply bandgap reference. Default: 0
18	PACTL_REFDECIN_SEL	PACTL DAC Connects to Ref Resistor Input. Default: 0
17	PACTL_DACIN_SEL	PACTL DAC Connects Vdac to Input. Default: 0
16	PACTL_ENABLEFB_SEL	Reserved. Default: 0.
15	PACTL_HGAIN_SEL	PACTL_DAC Select High Gain. Default: 0
14:12	PACTL_DECOPA_SEL	PACTL DAC Decoupling Op Amp Bias Control (2:0). Default: 00
11:9	PACTL_OUTOPA_CTL	PACTL DAC output Op Amp Bias Control (2:0). Default: 00
8	ON_LINE_START_SEL	NOT USED

WEB_MISC2_WR (Continued)

Bits	Name	Description
7:6	GSM_SBI_SEL	Controls SBI Muxing: '00': 1X controls both the external and internal SBI bus pins '01': 1X controls the external pins gsm_core controls the internal pins '10': 1X controls the internal pins gsm_core controls the external pins '11': GSM controls both the external and internal SBI bus pins
5	UNUSED_BIT5	
4	PACTL_DAC2_ENA	
3	PACTL_BIREFIN_SEL	
2	PACTL_ONEGAIN_SEL	
1	PACTL_TESTMODE_SEL	
0	PACTL_DAC1_SEL	

0x0018 WEB_MISC_RD

Type: Read
Clock: HCLK

Bits	Name	Description
0	ONES_DETECT	This bit returns the status of ONES_DETECT for AUX_CODEC.

21.1.1.3 RF control registers

TX has a separate control pin: TX_ON (in addition to PA_ON). PA_WARMUP controls the PA_ON; TX_WARMUP controls TX_ON.

0x001C PA_ON_CTL

Type: Write
Clock: HCLK
Reset State: 0x00

PA_ON_CTL

Bits	Name	Description
7	BYPASS_TX_SYNTH_LOCK	TX_ON has an "AND synth_lock" term, and by setting this bit to "1", we ignore the "AND synth_lock". This is to support TX chain, which has a synthesizer inside.
6	BYPASS_TX_PUNCT	Same function as BYPASS_PA_PUNCT for TX_PUNCT.
5	TX_ON_EN	Enables TX_ON function.

PA_ON_CTL (Continued)

Bits	Name	Description
4	BYPASS_PA_SYNTH_LOCK_RAW	BYPASS_PA_SYNTH_LOCK = BYPASS_PA_SYNTH_LOCK_RAW AND pa_on_guard PA_ON has an “OR synth_lock” term, and by setting BYPASS_PA_SYNTH_LOCK to “1”, we ignore the “OR synth_lock”.
3	BYPASS_PA_PUNCT_RAW	BYPASS_PA_PUNCT = BYPASS_PA_PUNCT_RAW AND pa_on_guard Override the PA_PUNCT signal, so that we can turn on PA independent of PA_PUNCT. It still follows the PA_ON_EN, SYNTH_LOCK, IDLE_N states. Note: In modulator, we have an override bit for PA_PUNCT and TX_PUNCT at the same time.
2	UNUSED_BIT2	Reserved for future use.
1	BAND_SEL	‘1’ will select PA_ON2 ‘0’ will select PA_ON
0	PA_ON_EN	Enables the PA_ON function.

0x0020 PA_ON_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
11	HDR_PA_PUNCT	The PA puncture control signal from HDR modulator
10	HDR_TX_PUNCT	The TX puncture control signal from HDR modulator
9	TX_ON	The state of TX_ON output signal
8	PA_PUNCT	The PA puncture signal from CDMA modulator
7	PA_ON2	The state of PA_ON2 output signal
6	DEBUG_STATE	DBGACK signal from ARM, indicating ARM entered debug state
5	PA_ON	The state of PA_ON output signal
4	UNUSED_BIT4	
3	UNUSED_BIT3	Formerly defined as DFM_PAON_TXCTRL; now UNUSED.
2	UNUSED_BIT2	Formerly defined as CDMA_FM_N; now UNUSED.
1	TX_PUNCT	The TX puncture control signal from CDMA modulator
0	SYNTH_LOCK	The state of SYNTH_LOCK pin

21.1.1.4 Codec read registers

0x0024 **Reserved.**

0x0028 **Reserved.**

21.1.1.5 PAD registers

0x002C **PAD_PUPD_EN**

Type: Write
Clock: HCLK
Reset State: 0xFFFF

This register is programmable to either value 0 or 1. At the power on condition, all the bits are set to (1). PAD_PUPD_N/PAD_PUPD_EN Select: 11 : pull up, 01 : pull down, others : no pull up/pull down

PAD_PUPD_EN

Bits	Name	Description (pins controlled by each bit)
17	AD_Y2Q_PUPD_CH1_EN	AD_Y2Q_CH1
16	AD_Y2I_PUPD_CH1_EN	AD_Y2I_CH1
15	AD_Y1Q_PUPD_CH1_EN	AD_Y2Q_CH1
14	AD_Y1I_PUPD_CH1_EN	AD_Y1I_CH1
13	TX_AGC_ADJ_PUPD_EN	TX_AGC_ADJ
12	TRK_LO_ADJ_PUPD_EN	TRK_LO_ADJ
11	UNUSED_BIT11	
10	SBCK_PUPD_EN	SBCK
9	UNUSED_BIT9	
8	PA_R1_PUPD_EN	PA_R1
7	PA_R0_PUPD_EN	PA_R0
6	UNUSED_BIT6	
5	UNUSED_BIT5	
4	AD_Y2Q_PUPD_CH0_EN	AD_Y2Q_CH0
3	AD_Y2I_PUPD_CH0_EN	AD_Y2I_CH0
2	AD_Y1Q_PUPD_CH0_EN	AD_Y2Q_CH0

PAD_PUPD_EN (Continued)

Bits	Name	Description (pins controlled by each bit)
1	AD_Y1I_PUPD_CH0_EN	AD_Y1I_CH0
0	AD_CODEC_DO_PUPD_EN	AD_CODEC_DO

0x0030 PAD_PUPD_N

Type: Write
Clock: HCLK

Reset State: 0x0000

Bits	Name	Description (controlled pin by each pin)
17	AD_Y2Q_PUPD_CH1_N	AD_Y2Q_CH1
16	AD_Y2I_PUPD_CH1_N	AD_Y2I_CH1
15	AD_Y1Q_PUPD_CH1_N	AD_Y2Q_CH1
14	AD_Y1I_PUPD_CH1_N	AD_Y1I_CH1
13	TX_AGC_ADJ_PUPD_N	TX_AGC_ADJ
12	TRK_LO_ADJ_PUPD_N	TRK_LO_ADJ
11	UNUSED_BIT11	
10	SBCK_PUPD_N	SBCK
9	UNUSED_BIT9	
8	PA_R1_PUPD_N	PA_R1
7	PA_R0_PUPD_N	PA_R0
6	UNUSED_BIT6	
5	UNUSED_BIT5	
4	AD_Y2Q_PUPD_CH0_N	AD_Y2Q_CH0
3	AD_Y2I_PUPD_CH0_N	AD_Y2I_CH0
2	AD_Y1Q_PUPD_CH0_N	AD_Y2Q_CH0
1	AD_Y1I_PUPD_CH0_N	AD_Y1I_CH0
0	AD_CODEC_DO_PUPD_N	AD_CODEC_DO

0x0034 PAD_HDRIVE_SEL_0

Type: Write
Clock: HCLK
Reset State: 0x00000000

Bits	Name	Description
31:28	UNUSED_BITS31_28	RESERVED
27	HDRIVE_MMC	GPIO32:30
26:18	HDRIVE_GPIO_29_TO_21	Set (1) bit *n* to select the high drive of the corresponding GPIO (*n*+3). To clear (0) the high drive, set bit *n* to ‘0’.
17:0	HDRIVE_GPIO_17_TO_0	Set (1) bit *n* to select the high drive of the corresponding GPIO *n*. To clear (0) the high drive, set bit *n* to ‘0’.

Set (1) to select high-drive for the corresponding pin(s); clear (0) to de-select high-drive

0x0038 PAD_HDRIVE_SEL_1

Type: Write
Clock: HCLK
Reset State: 0x00000

Bits	Name	Description
19	HDRIVE_ROM1CLK	Set bit to ‘1’ to select the high drive. To clear the high drive, set bit to ‘0’.
18:11	HDRIVE_GPIO_106_TO_99	Set bit *n* to ‘1’ to select the high drive of the corresponding GPIO (*n*+99). To clear(0) the high drive, set bit *n* to ‘0’.
10:8	HDRIVE_GPIO_20_TO_18	Set (1) bit *n* to select the high drive of the corresponding GPIO (*n*+10). To clear (0) the high drive, set bit *n* to ‘0’.
7	HDRIVE_GPIO33	GPIO33
6	HDRIVE_AUX_PCM	GPIO83:80
5	HDRIVE_UART1	GPIO98:95
4	HDRIVE_UART2	GPIO91:88
3	HDRIVE_UART3	GPIO87:84
2	HDRIVE_GPIO94	GPIO94
1	HDRIVE_GPIO93	GPIO93
0	HDRIVE_USB	USB_DAT_VP USB_SE0_VM USB_TP_OE_N

Set (1) to select high-drive for the corresponding pin(s); clear (0) to de-select high-drive

0x003C PAD_HDRIVE_SEL_2

Type: Write
Clock: HCLK
Reset State: 0x01800

Set to ‘1’ to select high-drive, set to ‘0’ to de-select high-drive. All bits are set to 0, except bit11 (hdrive_ebi1) and bit12 (hdrive_ebi1_cntl).

PAD_HDRIVE_SEL_2

Bits	Name	Description (Pins controlled by each bit)
17	UNUSED_BIT17	Reserved.
16	HDRIVE_ETM	GPIO66:39
15	HDRIVE_RF_INTF	PA_ON[0] PA_RANGE[1:0] TRK_LO_ADJ TX_AGC_ADJ TX_ON Note that PA_ON[1] is behind gpio2, and its hdrive can be controlled using PAD_HDRIVE_SEL_0[2]
14	HDRIVE_EBI2_CNTL2	LB2_N UB2_N OE2_N WE2_N XMEM2_CS_N1 XMEM2_CS_N0 LCD_CS_N (gpio38) LCD_EN (gpio37) XMEM2_CS_N3 (gpio36) XMEM2_CS_N2 (gpio35)
13	HDRIVE_EBI2	A2[19:1] D2[15:0] A2[20] (gpio34)

PAD_HDRIVE_SEL_2 (Continued)

Bits	Name	Description (Pins controlled by each bit)
12	HDRIVE_EBI1_CNTL	LB1_N UB1_N WE1_N OE1_N XMEM1_CS_N[2] XMEM1_CS_N[0] SDRAM1_CLK_EN ROM1_ADV_N ROM1_WAIT_N HLB_N (gpio73) HUB_N(gpio74) HROM1_WAIT_N XMEM1_CS_N[3] (gpio77) XMEM1_CS_N[1] (gpio76) SDRAM1_DQM_3 (gpio75)
11	HDRIVE_EBI1	A1[22:1] D1[15:0] A1_25 (gpio75) A1_24 (gpio79) A1_23 (gpio78) SDRAM1_D[23:16] (gpio74:67)
10	HDRIVE_JTAG	TCK TDI TDO TMS TRST_N RTCK
9	HDRIVE_EXTN_SBI	SBCK SBDT SBST
8	UNUSED_BIT8	Reserved.
7		Reserved.
6		Reserved.
5		Reserved.
4		Reserved.
3		Reserved.

PAD_HDRIVE_SEL_2 (Continued)

Bits	Name	Description (Pins controlled by each bit)
2		Reserved.
1		Reserved.
0		Reserved.

21.1.1.6 Analog-die control register

0x0040 SLEEP_N_ADIE

Type: Write
Clock: HCLK

This register controls the ad_sleep_n die-to-die pad, at power-on, it clears to '0,'

Bits	Name	Description
0	AD_SLEEP_N	This bit controls the analog die sleep_n die-to-die pad.

0x0044 Reserved.

0x0048 Reserved.

0x004C Reserved.

21.1.1.7 AUX/CODEC interface registers

0x0050 CODEC_CTL

Type: Write
Clock: HCLK

The CODEC_CTL register is used to configure the hardware-level behavior of the PCM Codec interface. All bits are cleared on assertion of RESOUT.

CODEC_CTL

Bits	Name	Description
15:14	LOOP_SEL	Codec data path loop selection for testing ‘00’: Normal operation ‘01’: loop+ for testing ‘10’: loop- for testing ‘11’: over sampled data input for testing
13	I2S_EN	‘0’: I^2S is disabled (default) ‘1’: I^2S enable (PCM, AUX_PCM and SDAC are all disabled). For this case, if SDAC_en (bit 1) =0, the stereo clock used for the MSM internal I^2S interface circuit will be in the same phase as the clock used outside the chip; If SDAC_en (bit 1) =1, then the two clocks are in reverse phase.
12	ADSP_CODEC_CTL_EN	‘0’: This register is programmed by MSM (default) ‘1’: This register is programmed by ADSP
11	AUX_CLK_EN	When cleared(0), it allows PCM_CLK and PCM_SYNC signals to be driven from AUX_CLK_VAL and AUX_SYNC_VAL for testing purposes.
8	CLK_SYNC_LOOP	When set(1), it allows PCM_CLK and PCM_SYNC signals to loop back into AUX_PCM_CLK and AUX_PCM_SYNC pins for testing purposes.
7	BCLK_DIR	For stereo application. ‘0’: Stereo BCLK is output (default); I^2S and SDAC bit clock will be generated from our MSM clock block. ‘1’: Stereo BCLK is Input; I^2S and SDAC bit clock will be generated from an outside device.
6	PCM_CLK_SENSE	The polarity of PCM_CLK and PCM_SYNC can be inverted by this bit.
5	MCLK_DIR	Stereo Master Clock select. ‘0’ (default): Stereo MCLK is output (generated from our MSM clock block) for both I^2S and SDAC cases. ‘1’: Stereo MCLK is Input, from an outside device to our MSM clock block; for both I^2S and SDAC cases.

CODEC_CTL (Continued)

Bits	Name	Description
4	AUX_CODEC_SEL	When set (1), the auxiliary codec is selected. When cleared (0), the primary PCM codec is selected.
3	AUX_INVERT	When set (1), this bit is used to invert the polarity of AUX_PCM_CLK, AUX_PCM_SYNC, AUX_PCM_DIN, and AUX_PCM_DOUT.
2	PCM_CLK_DIR	'0' : PCM_clk/AUX_PCM_clk is output, so is PCM_sync/AUX_sync. '1': PCM_clk/AUX_PCM_clk is input, so is PCM_sync/AUX_PCM_sync.
1	SDAC_EN	Default is (0); When set (1) will enable SDAC (disable PCM/AUX_PCM). Bit 13 set '1' will overwrite this bit.
0	ONES_POLARITY	ONE detect (ADSP not able to program this bit).

0x0054 PCM_PATH_CTL

Type: Write
Clock: HCLK
Reset State: 0x0000

This register controls the PCM path among MDSP, VDSP, and external devices.

Bits	Name	Description
3	VDSP_CTL_EN	'0': PCM path is controlled by ARM (default) '1': PCM path is controlled by VDSP
2	VDSP_RPCM_SEL	'0': VDSP RPCM data from AUX_CODEC (default) '1': VDSP RPCM data from MDSP
1	MDSP_RPCM_SEL	'0': MDSP RPCM data from VDSP (default) '1': MDSP RPCM data from AUX_CODEC
0	EXT_RPCM_SEL	'0': EXT RPCM data from VDSP (default) '1': EXT RPCM data from MDSP

21.1.1.8 I²C registers

This section describes the I²C registers.

0x0060 I2C_WRITE_DATA

Type: Write
Clock: HCLK

This register contains the information the controller uses to organize byte transfers over the I²C bus.

Bits	Name	Description
9	LAST_BYTE	When transmitting data, setting this bit to 1 while writing to this register informs the controller that the accompanying data_byte will be the final byte transmitted. When receiving data, if addr_byte is not set, setting this bit will inform the controller that the data byte presently being received will be the final one. The controller will then NACK the packet, and issue a STOP command on the bus. If addr_byte is set, this bit is associated with the accompanying data_byte.
8	ADDR_BYTE	Setting this bit to 1 while writing to this register informs the controller that the accompanying data_byte is an address byte and that a subsequent START condition must precede it on the bus.
7:0	DATA_BYTE	Byte (addr +rd/wr_n or data) written to this register is destined for the slave device. These 8 bits are ignored if receiving data and addr_byte not set.

Table 21-1 I²C operations

LAST_BYTE	ADDR_BYTE	DATA_BYTE	Tx/Rx	Description of Operation
0	0	data	Tx	Data transmission
0	1	addr	Tx	START -> addr transmission
1	0	data	Tx	data transmission -> STOP
1	1	addr	Tx	START -> addr transmission -> STOP
0	0	ignored	Rx	Data reception (ACK) + invalid_write bit set in STATUS
1	0	ignored	Rx	Data reception (NACK) -> STOP
x	1	addr	Rx	Data reception (NACK) -> START -> addr transmission. If last_byte = 1, then STOP follows addr transmission.

0x0064 I2C_CLK_CTL

Type: Read/Write
Clock: HCLK

This is a read/write register that controls clock divider values.

Bits	Name	Description
10:8	HS_DIVIDER_VALUE	The value in this register represents the clock period multiplier in high-speed (HS) mode. The minimum value should be 3hex (4 i2c_clk clocks per period) to ensure proper sampling of the bus. For a maximum bit rate of 3.4Mbits/sec, this would require a minimum 40.8 MHz i2c_clk clock. The maximum i2c_clk is 81.6 MHz. This register is reset to a maximum value of 7hex.
7:0	FS_DIVIDER_VALUE	The value in this register represents the clock period multiplier in Fast/Standard (FS) mode. The minimum value should be 3hex to ensure proper sampling of the bus. For a maximum bit rate of 400kbits/sec (Fast mode), this would require a 4.8 MHz i2c_clk clock. For a maximum bit rate of 100kbits/sec (standard mode), this would require a 1.2 MHz i2c_clk clock. Maximum i2c_clk for fast and standard modes are 206.4MHz and 51.6 MHz respectively. This register is reset to a maximum value of 256hex.

Table 21-2 I^2C clock frequencies

i2c_clk frequency	HS (hex)	Fast (hex)	Standard (hex)	CLK_CONTROL fast/standard (hex)	Resultant clock frequencies hs / fast / standard (MHz)
HCLK = 75 MHz	7	5B	N/A	0x75B / N/A	3.125 /.399 / N/A
50 MHz	4	3C	F7	0x43C / 0x4F7	3.333 /.397 /.100
HCLK/2 = 37.5 MHz	3	2C	B9	0x32C / 0x2B9	3.125 /.399 /.0997
25 MHz	3	1D	7A	0x31D / 0x37A	2.083 /.390 /.100
HCLK/4 = 18.75 MHz	3	17	5B	0x317 / 0x35B	1.563 /.391 /.0997
HCLK/8 = 9.38 MHz	3	9	2C	0x309 / 0x32C	.781 /.391 /.0997
HCLK/16 = 4.69 MHz	N/A	3	15	0x30c / 0x315	N/A /.390 /.0977
HCLK/32 = 2.34 MHz	N/A	3	9	0x303 / 0x309	N/A /.195 /.0977

To ensure that all three possible operating modes are available at any given time, the i2c_clk frequency should be within 51.6 MHz and 18.75 MHz. However, if the high_speed mode (HS) is not going to be used, then the i2c_clk frequency can be as low as 1.2 MHz.

0x0068 I2C_STATUS

Type: Read
Clock: HCLK

This is the read-only status register.

I2C_STATUS

Bits	Name	Description
9	BUS_MASTER	This bit is set (1) when the I^2C controller is the present bus master.
8	BUS_ACTIVE	This bit is set(1) when the bus is in use by this, or any other controller.
7:6	FAILED	This 2-bit field contains the failure information of the present I^2C transfers. If transmitting, failed[1] contains the information regarding the byte that has been transmitted. Failed[0] contains the information of the byte awaiting transmission if there is a byte pipelined. If no byte is pipelined, ignore this bit. If receiving, failed[1] contains the information of the byte received and stored in the buffer, and failed[0] should be ignored. For example: ‘00’: Byte n transmitted successfully, byte n+1 to begin transmission. ‘01’: Byte n transmitted successfully, byte n+1 errored: type of error indicated in other STATUS bits (queued invalid write, for example). ‘10’: Byte n errored: type of error indicated in other STATUS bits, byte n+1 to begin transmission (byte n+1 would have to be a valid address byte for this condition to occur) ‘11’: Byte n errored: type of error indicated in other STATUS bits, byte n+1 discarded as well (if the byte was a data byte destined for a NACKed address, data is useless, therefore discarded)
5	INVALID_WRITE	This bit is set (1) when software writes data to the I2C_DATA register that should be flagged as an address but is not.
4	ARB_LOST	This bit is set (1) when the controller loses arbitration for the bus. If this bit gets set during the transmission, all data has been lost, and uP must re-request its transmission.
3	PACKET_NACKED	This bit is set(1) high when a NACK is received from a slave.
2	BUS_ERROR	This bit is set (1) when an unexpected START or STOP condition is detected. This returns the controller to its reset state.
1	RD_BUFFER_FULL	Data in read buffer is available to the uP when this bit is set to 1.
0	WR_BUFFER_FULL	Data can be written to the write buffer when this bit is set to 0.

0x006C I2C_READ_DATA

Type: Read
Clock: HCLK

This register contains the last_byte bit, start_flag bit as well as the next byte to be transmitted (addr or data).

Bits	Name	Description
7:0	DATA_BYTE	Received data is loaded to this register, and an interrupt is generated to inform the uP to read its contents.

0x0070 USB_PIN_CONFIG

Type: Write
Clock: HCLK

This register allows software to control the USB pins directly when the USB_PIN_SEL register is programmed for direct microprocessor control.

Bits	Name	Description
5	REG_OE_TP_OE	This bit controls the output enable of the USB_OE_TP_N line. Set (1) to enable the output.
4	REG_TP_OUT_N	This bit controls the value of the USB_OE_TP_N line.
3	REG_DAT_OE	This bit controls the output enable of the USB_DAT_VP line. Set (1) to enable the output.
2	REG_DAT_OUT	This bit controls the value of the USB_DAT_VP line.
1	REG_SE0_OE	This bit controls the output enable of the USB_SE0_VM line. Set (1) to enable the output.
0	REG_SE0_OUT	This bit controls the value of the USB_SE0_VM line.

0x0074 USB_PIN_SEL

Type: Write
Clock: HCLK

This register is used to control the muxing of functions on the USB pins. The USB pins can be switched between the following interfaces: USB, UART, and microprocessor control.

USB_PIN_SEL

Bits	Name	Description
2	USB_UART_SEL	This bit is used to enable the UART1 interface using the USB pins. Refer to Table 21-3 for programming information.

USB_PIN_SEL (Continued)

Bits	Name	Description
1	USB_REG_SEL	This bit is used to enable the microprocessor to control the USB pins through software. See below for programming information.
0		Reserved.

Table 21-3 USB_UART_SEL and USB_I2C_SEL modes

USB_UART_SEL	USB_REG_SEL		Mode
X	0		USB
0	1		Register
1	1		UART

0x0078 Reserved.

21.1.2 TCXO4 clock registers

The peripherals registers in this section are clocked by TCX04_CLK.

21.1.2.1 PDM0, PDM1, and PDM2 registers

0x0000 TCXO_PDM_CTL

Type: Write
Clock: TCXO4_CLK

The TCXO_PDM_CTL register is used to enable/disable the PDM0, PDM1 and PDM2 pins, and to configure the mode of operation for the PDM output.

.

TCXO_PDM_CTL

Bits	Name	Description
5	PDM2_POLARITY	This bit defines the polarity of PDM2. Set (1) this bit to invert the sense of the PDM output.
4	PDM1_POLARITY	This bit defines the polarity of PDM1. Set (1) this bit to invert the sense of the PDM output.
3	PDM0_POLARITY	This bit defines the polarity of PDM0. Set (1) this bit to invert the sense of the PDM output.

TCXO_PDM_CTL (Continued)

Bits	Name	Description
2	PDM2_EN	■ Setting (1) this bit enables the PDM2 pin. ■ Clearing (0) this bit disables PDM2 and places it in a high impedance state. This bit is cleared on RESOUT.
1	PDM1_EN	■ Setting (1) this bit enables the PDM1 pin. ■ Clearing (0) this bit disables PDM1 and places it in a high impedance state. This bit is cleared on RESOUT.
0	PDM0_EN	■ Setting (1) this bit enables the PDM0 pin. ■ Clearing (0) this bit disables PDM0 and places it in a high impedance state. This bit is cleared on RESOUT.

0x0004 PDM0_CTL

Type: Write
Clock: TCXO4_CLK

Bits	Name	Description
15:0	DATA	The PDM0_CTL register is the pulse density modulation register number 0. This register determines the density of pulses on the PDM0 pin. When the PDM0_POLARITY is set, the output sense is inverted and the clock source is GENERAL_CLK/4.

Table 21-6 lists the signal formats of the PDM0 for various PDM0_CTL register values, when PDM0_POLARITY = 0.

Table 21-4 PDM0 signal formats for various PDM0_CTL register values

Register value	Signal format
0x8000	Low for 65536/65536 TCXO4 pulses
0x0000	Low for 32768/65536 TCXO4 pulses
0x7FFF	Low for 1/65536 TCXO4 pulses

0x0008 PDM1_CTL

Type: Write
Clock: TCXO4_CLK

Bits	Name	Description
7:0	DATA	The PDM1_CTL register is the pulse density modulation register number 1. This register determines the density of pulses on the PDM1 pin. When the PDM1_POLARITY is set, the output sense is inverted and the clock source is GENERAL_CLK/4.

Table 21-5 lists the signal formats of the PDM1 for various PDM1_CTL register values, when PDM1_POLARITY = 0.

Table 21-5 PDM1 signal formats for various PDM1_CTL register values

Register value	Signal format
0x80	Low for 256/256 TCXO4 pulses
0x00	Low for 128/256 TCXO4 pulses
0x7F	Low for 1/256 TCXO4 pulses

0x000C PDM2_CTL

Type: Write
Clock: TCXO4_CLK

Bits	Name	Description
7:0	DATA	The PDM2_CTL register is the pulse density modulation register number 2.It determines the density of pulses on the PDM2 pin. When the PDM2_POLARITY is set (1) , the output sense is inverted and the clock source is GENERAL_CLK/4.

Table 21-6 lists the signal formats of the PDM2 for various PDM2_CTL register values, when PDM2_POLARITY = 0.

Table 21-6 PDM2 signal formats for various PDM2_CTL register values

Register value	Signal format
0x80	Low for 256/256 TCXO4 pulses
0x00	Low for 128/256 TCXO4 pulses
0x7F	Low for 1/256 TCXO4 pulses

Reserved.

Reserved.

Reserved.

21.1.2.2 Ringer registers

0x001C **Reserved.**

0x0020 **Reserved.**

0x0024 **RINGER_MN_A_MDIV**

Type: Write
Clock: TCXO4_CLK

The RINGER_MN_A_MDIV register is the A counter M register for the ringer M/N. This register is written with the A counter M value and with the enable control for the A counter.

Bits	Name	Description
6	RINGER_MN_A_EN	This bit is the enable bit for the A counter. ■ When this bit is cleared (0), the counter is disabled (CNTR_A_M_EN has an edge detect for a 0 to 1 transition). ■ When this bit is set (1), it resets both counters (CNTR_A and CNTR_B) to sync to the same time base (they start counting from the same clock). This bit is clear (0) on RESOUT.
5:0	RINGER_MN_A_MDIV	This bitfield programs the A counter M value.

0x0030 RINGER_MN_B_MDIV

Type: Write
Clock: TCXO4_CLK

The RINGER_MN_B_MDIV register is the B counter M register for the ringer M/N. This register is written with the B counter M value and with the enable control for the B counter.

Bits	Name	Description
6	RINGER_MN_B_EN	This bit is the enable bit for the B counter. ■ When this bit is cleared (0), the counter is disabled (CNTR_B_M_EN has an edge detect for a 0 to 1 transition). ■ When this bit is set (1), it resets both counters (CNTR_A and CNTR_B) to sync to the same time base (they start counting from the same clock). This bit is clear (0) on RESOUT.
5:0	RINGER_MN_B_MDIV	This bitfield programs the B counter M value.

21.1.2.3 Timetick registers

0x0034 TIME_TICK_CTL

Type: Write
Clock: TCXO4_CLK

The TIME_TICK_CTL register is the time tick control register. This register is used to define the behavior of the TIME_TICK_INT.

Bits	Name	Description
2	SYNC	This bit allows the software to sync the rate generation to a specific time boundary. This bit is positive-edge detected. So, when the TCXO clock detects a zero-to-one transition, it generates a reset to the counter. Within approximately 2 TCXO clocks, the counter is cleared. Once the counter is cleared, it begins counting again (about 400 nS later). ■ Set (1) this bit whenever TIME_TICK_INT is to be synchronized. ■ Under normal operating conditions, clear (0) this bit. The software must clear (0) SYNC whenever an interrupt rate is defined.
1:0	RATE_SEL	This bit determines the interrupt rate of the TIME_TICK_INT. 00 : 1.25 ms 01 : 2.5 ms 10 : 5 ms 11 : 10 ms

0x0038 TIME_TICK_INT_MSB

Type: Read
Clock: TCXO4_CLK

Bits	Name	Description
7:0	DATA	The TIME_TICK_INT_MSB register is the time tick interrupt register. This register provides the instantaneous value of the TIME_TICK_INT counter. The counter is 8 bits wide, with a clock of TXCO/4. It takes 52 s to change one bit in this register

22 SBI

22.1 SBI overview

The MSM6550 device contains two SBI controllers, SBI0 and SBI1. SBI0 is to connect to receive-chain one, and SBI1 is to connect to receive-chain two

Each SBI controller contains its own duplicate set of ARM registers. Rather than repeat identical descriptions for each entry, the following list uses notation (“c”) that defines the register as belonging to either SBI Rx chain 0 or SBI Rx chain 1. The “c,” representing 0 or 1, appears immediately after the word “SBI” that prepends each register name (for example, SBI0_* or SBI1_*).

The two sets of registers are separated by 64 bytes of address space, or 0x40 hexadecimal. (Each Rx front chain has been assigned 16 Word addresses, or 16 possible registers each. Thus the offset from SBI0 to SBI1 should be 16 words, or 64 bytes.) Each address is notated using the offset plus 0x40*c, where c is the 0 or 1 chain feeding into the SBI0 or SBI1.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block names: SBI0, SBI1
- Number of word addresses: 64 for each block
- Byte Address:
  - SBI0 = 0x5F00 - 0x5F3C
  - SBI1 = 0x5F40 - 0x5FFC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

22.2 ARM registers

This section contains descriptions of SBI read and write registers.

22.2.1 Write registers

0x000+0x40c SBIc_CLK_CTL, c=[0,1]

Type: Write
Clock: SBI_CLK

The SBIc_CLK_CTL register is the SBI clock control register.

Bits	Name	Description
7	MICRO_RESET	This bit resets the microprocessor. ■ Set (1) this bit to reset the clock divider block and state machine controller. ■ Clear (0) this bit to refresh the clock divider block; i.e., reload the counter with the value programmed in CLK_CTL part of this register.
6	CNT_EN	When set (1), this bit enables the clock sub-divider counter. This bit defaults to ‘0’ after initial chip reset.
5:0	CLKCTL	These bits determine the modulo divide ratio (M) to generate SBCK. For example, if the desired serial bus rate is 164 kHz and the SBI controller clock is the TCXO/4 clock, then the required ratio is M = 4.92MHz/164kHz = 30 (decimal). Therefore, the value written into this field is 0x1E. If these bits are clear (0), the result is a zero clock frequency on the SBCK line, and high for all time. These bits are cleared (0) on RESOUT_N (initial reset).

0x004+0x40c SBIc_CTL, c=[0,1]

Type: Write
Clock: SBI_TCX04_CLK
Reset State: 0x0

The SBIc_CTL register is the SBI mode control register. All bits are clear (0) upon power-up reset. Hardware request 3 (HW_REQ_3) ports are added for “interrupt transfer mode” only. This port is to be used for ZIF gain stepping.

SBIc_CTL

Bits	Name	Description
21:20	HW_REQ3_PRIORITY	There are four hardware request modules. All of them have higher priority than the microprocessor. These bits help program the relative priority of the four hardware requests. 00: 1st priority 01: 2nd priority 10: 3rd priority 11: 4th priority
19	HW_REQ3_EN	Enables hw_req3 port. (only for “interrupt transfer mode”)
18:17	HW_REQ2_PRIORITY	There are four hardware request modules. All of them have higher priority than the microprocessor. These bits help the relative priority of the four hardware requests. 00: 1st priority 01: 2nd priority 10: 3rd priority 11: 4th priority
16	HW_REQ2_EN	This bit determines whether the hw_req2 feature is enabled or not. When this bit is set (1), the hw_req2 feature is enabled and the microprocessor will give the priority to the hw_req2 request if there is one. If the microprocessor is doing a one-word transaction, the SBI controller will wait until the current transaction is finished and then turn over the control from the microprocessor to the HW request. If the microprocessor is doing a multiple-word transaction, the SBI controller will wait until the current word is done and then turn over the control to the HW request. Once the HW request is done, the microprocessor will resume control of the SBI controller and start a new access that completes the rest of the words. When this bit is set (1), the hw_req2 feature is disabled and the microprocessor/hw_req1 or hw_req3 may take control of the SBI controller. This bit is clear (0) after the initial reset.

SBIc_CTL (Continued)

Bits	Name	Description
15:14	HW_REQ1_PRIORITY	There are four hardware request modules. All of them have higher priority than the microprocessor. These bits help program the relative priority of the four hardware requests. 00: 1st priority 01: 2nd priority 10: 3rd priority 11: 4th priority
13	HW_REQ1_EN	This bit determines whether the hw_req1 feature is enabled or not. When this bit is set (1), the hw_req1 feature is enabled and the microprocessor gives priority to the hw_req1 request, if there is one. If the microprocessor is doing a one-word transaction, the SBI controller waits until the current transaction is finished and then turn over control from the microprocessor to the HW request; If the microprocessor is doing a multiple-word transaction, the SBI controller will wait until the current word is done and then turn over control to the HW request. Once the HW request is done, the microprocessor will resume control of the SBI controller and start a new access that completes the rest of the words. When this bit is clear (0), the hw_req1 feature is disabled, and the microprocessor/hw_req2 or hw_req3 may take the control of the SBI controller. This bit is clear (0) after the initial reset.
12	I2C_I3Q_N	This bit determines the type of SBI device the MSM is going to access. 1: I^2C device 0: I3Q device. When the MSM is interfacing with an I^2C device, the sbdt_po is configured as an open-drain structure. When the MSM is interfacing with an I3Q device, it is configured as a tri-state structure. This bit is clear (0) after the initial reset.
11	MSM_SBI_EN	Set (1) this bit to enable the “forward compatibility mode” and allow the SBI controller to gain control of the ADC pins (namely, ADC_EN, ADC_DATA, and ADC_CLK). This bit is clear (0) after initial reset.
8	LAST_WORD	This bit is the last word flag. When this bit is set (1), the SBI controller terminates transactions after shifting the word associated with this flag. In the case where the flag is cleared (0) by the processor, the controller monitors the write buffer and terminates transactions when the buffer is empty. This bit is clear (0) after initial reset.

SBIc_CTL (Continued)

Bits	Name	Description
7:6	I3Q_MODE	I3Q mode control for a microprocessor-triggered SBI transaction or these bits are the protocol mode selector bits. 00: interrupt service 01: fast transfer mode 10: bulk transfer mode 11: undefined Clear (00 binary) this bitfield after initial reset.
5:0	SLAVE_ID	This 6-bit field is the unique slave SBI ID code. Clear (00 binary) this bitfield after initial reset.

0x008+0x40c SBIc_BYPASS_WR, c=[0,1]

Type: Write
Clock: SBI_CLK

The SBIc_BYPASS_WR register is used to override various SBI functions.

Bits	Name	Description
2	OVR_SBCK	Override mode serial bus clock. In the override mode (SBIc_CTL register, OVR_MODE bit = 1), this bit drives the SBI SBCK output.
1	OVR_SBST	Override mode serial start/stop. In the override mode (SBIc_CTL register, OVR_MODE bit = 1), this bit drives the SBI SBST output.
0	OVR_SBDT	Override mode serial output data. In the override mode (SBIc_CTL register, OVR_MODE bit = 1), this bit drives the SBI SBDT output.

0x00C+0x40c SBIc_WR, c=[0,1]

Type: Write
Clock: SBI_CLK

This is the read/write address and data.

Bits	Name	Description
15	RD_WR_N	This bit selects SBI read or write serial operation. ■ When set (1), the SBI performs a “read data from register address” operation. ■ When clear (0), the SBI performed a “write data to register address” operation.
14:8	REG_ADDR	This 7-bit field is the SBI slave device register address. The data in the REG_DATA field is either written to or read from the addressed register in the slave device.
7:0	REG_DATA	In the case of a write operation, this register contains data to be serially transferred to the slave device identified by SLV_ID and destined to the slave register identified by REG_ADD. In the case of a read operation, this register contains data serially transferred from the slave device identified by SLV_ID and from the slave register identified by REG_ADD.

0x010+0x40c SBIc_START_CTL, c=[0,1]

Type: Write
Clock: SBI_CLK

Bits	Name	Description
0	START_FLAG	The SBIc_START_CTL register is the SBI start flag for I3Q transaction. The SBI controller samples this flag in order to start the serialization of words through the SBI interface. The SBI controller begins operation after sensing a transition from 0 to 1. The controller expects the flag to return to 0 before starting a new transaction. After power-up reset the flag is clear (0).

0x014+0x40c SBIc_SECOND_IDCODE, c=[0,1]

Type: Write
Clock: SBI_CLK

Bits	Name	Description
7:0	SBI_SECOND_IDCODE	The SBIc_SECOND_IDCODE register is the IDCODE register for SBI devices inside the MSM chip. There are 2 MSBits for the mode and 6 LSBits to identify the address.

22.2.2 Read registers

0x018+0x40c SBIc_STATUS, c=[0,1]

Type: Read
Clock: SBI_CLK

The SBIc_STATUS register returns the status of the SBI.

Bits	Name	Description
15:14	I3Q_MODE_RD	These bits are the transfer mode bits. 00: interrupt service 01: fast transfer mode 10: bulk transfer mode 11 : Undefined_Bit
13:8	SLAVE_ID_RD	These bits are the slave ID code, as read from the slave device. For example, the code for IFR3000 is 011111 and the code for IFT3000 is 101010.
7	OVR_MODE_RD	When set (1), this bit indicates override mode. If MSM_SBI_EN is set (1), then the processor can implement software I^2C or I3Q by writing and reading the BYPASS register.
6	DIAG_MODE_RD	When this bit is set (1), the diagnostic mode (loop-back mode) is enabled.
5	MSM_SBI_EN_RD	■ When this bit is set (1), the forward compatible mode is enabled. ■ When this bit is clear (0), the backward compatible mode is enabled.
4:3	MCHN_STATE	These two bits indicate the current phase of the state machine. 00: reset state 01: CHIP_ID state 10: register address state 11: data processing state
2	MCHN_BUSY	This bit is the state machine busy flag. ■ When this bit is set (1), the state machine is busy and the SBI is not available to start a new transfer. ■ When this bit is clear (0), a new transaction can be started.
1	RDBUF_FULL	This bit is the read buffer flag. When this bit is set (1), data has been received from a slave and is waiting to be read by the processor.
0	WRBUF_FULL	This bit is the write buffer full flag. When this bit is clear (0), the write buffer is empty, and it can accept new address and data fields for transmission.

0x01C+0x40c SBIc_BYPASS_RD, c=[0,1]

Type: Read
Clock: SBI_CLK

Bits	Name	Description
4	SBCK_PIN_STATUS	This bit is the input for the override mode serial bus clock. This bit reflects the state of the SBCK input.
3	OVR_SBCK_RD	This bit is the output for the override mode serial bus clock. This bit reflects the state of the OVR_SBCK bit in the SBIc_BYPASS write register.
2	OVR_SBST_RD	This bit is the start/stop selector for the override mode serial bus. It reflects the state of the OVERRIDE_SBST bit in the SBIc_BYPASS write register.
1	OVR_SBDT_RD	This bit is the output of the override mode serial bus data. It reflects the state of the OVERRIDE_SBDT bit in the SBIc_BYPASS write register.
0	SBDT_PIN_STATUS	This bit is the input of the override mode serial bus data. This bit reflects the state of the SBDT input.

0x020+0x40c SBIc_RD, c=[0,1]

Type: Read
Clock: SBI_TCX04_CLK

The SBIc_RD register is the address and data of the last SBI read result.

Bits	Name	Description
15	RD_WR_N	This bit indicates if the SBI was performing a read (‘1’) or write (‘0’) operation. The bit value is the same as what was written into the SBIc_WR register.
14:8	REG_ADDR	This bitfield is the SBI slave device register address. The data in the REG_DATA field is either written to or read from the addressed register in the slave device.
7:0	REG_DATA	Data transferred to the MSM via the SBI from the slave device. The REG_ADDR bits indicate the register in the slave device.

0x024+0x40c SBIc_RD1, c=[0,1]

Type: Read
Clock: SBI_CLK

The SBIc_RD1 register is the address and data of the second-to-last SBI read result.

Bits	Name	Description
15	RD_WR_N	This bit indicates if the SBI was performing a read (‘1’) or write (‘0’) operation. The bit value is the same as what was written into the SBIc_WR register.
14:8	REG_ADDR1	This bitfield is the SBI slave device register address. The data in the REG_DATA1 field is either written to or read from the addressed register in the slave device.
7:0	REG_DATA1	Data transferred to the MSM via the SBI from the slave device. The REG_ADDR1 bits indicate the register in the slave device.

0x028+0x40c SBIc_RD2, c=[0,1]

Type: Read
Clock: SBI_CLK

The SBIc_RD2 register is the address and data of the third-to-last SBI read result.

Bits	Name	Description
15	RD_WR_N	This bit indicates if the SBI was performing a read (‘1’) or write (‘0’) operation. The bit value is the same as what was written into the SBIc_WR register.
14:8	REG_ADDR2	This bitfield is the SBI slave device register address. The data in the REG_DATA2 field is either written to or read from the addressed register in the slave device.
7:0	REG_DATA2	Data transferred to the MSM via the SBI from the slave device. The REG_ADDR1 bits indicate the register in the slave device.

0x02C+0x40c SBIc_INT_STATUS, c=[0,1]

Type: Read
Clock: SBI_CLK

The SBIc_INT_STATUS register is the interrupt status register for the SBI.

Bits	Name	Description
0	DATA	Microprocessor writes a ‘0’ to reset this bit. This register bit is set (1) if any microprocessor directed access is successful finished.

23 UART

23.1 Overview

MSM6550/6150 has three UARTs, each with its own set of registers. The register sets are virtually identical, except for the following differences, which stem from the added SIM/UIM functionality in UART2 and UART3 :

1. The UART_MR1 contains bits 10:8, which are not present in the equivalent UART2[3] registers.
2. The UART2[3]_SIM_CFG register is not present in UART1.
3. UART2[3]_IMR, UART2[3]_ISR, and UART2[3]_MISR registers contain bit 7, which the equivalent UART1 registers do not.
4. The UART2[3]_CR register includes channel command 8 for the RESET_TX_ERROR function, which is not available in UART1.

23.2 ARM registers

UART registers are accessible only by the ARM.

- Block name: UART1/2/3
  - UART1:
    - Offset = CHIP_BASE + 0x4E00; MAX = CHIP_BASE + 0x4EFC
  - UART2:
    - Offset = CHIP_BASE + 0x4F00; MAX = CHIP_BASE + 0x4FFC
  - UART3:
    - Offset = CHIP_BASE + 0x5000; MAX = CHIP_BASE + 0x50FC
- Number of word addresses: 192
- Byte Address: 0x4E00 - 0x50FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

23.2.1 UART1 registers

This section contains the read and write registers for UART1.

23.2.1.1 Write and read/write registers

0x0000 UART_MR1

Type: Read/Write
Clock: UART_CLK

The UART_MR1 register is the UART mode register #1. It is used, along with UART_MR2, to configure the operational mode of the UART.

Bits	Name	Description
10:8	AUTO_RFR_LEVEL1	These bits are used, along with bits 5:0 (AUTO_RFR_LEVEL0) to program the level in the receive FIFO at which the RFR_N signal is deasserted, if programmed to do so. This value is programmed from 1 to 511. The RFR_N signal is deasserted when the RX FIFO level (the number of characters remaining in the RX FIFO) is greater than the level that is programmed into this register.
7	RX_RDY_CTL	Setting (1) this bit enables the automatic ready-for-receiving (RFR_N) control for the receiver. RFR_N is turned off, or set (1), when a valid start bit is received and the channel FIFO is at the level programmed in bits 4 through 0 of this mode register. When the FIFO level falls below the programmed level, the RFR_N signal is turned on, or clear (0). When this feature is off, the RFR_N signal can be used by normal signal port bit manipulation.
6	CTS_CTL	When this bit is set(1), the transmitter checks the CTS_N input to determine whether to begin transmission of a new character. If CTS_N is low, the character is sent. Otherwise the transmitter continues marking until CTS_N goes low, then the next character is transmitted. A change on CTS_N during the transmission of a character has no effect on that character. When this bit is clear(0), the CTS_N input for the channel has no effect on the transmitter.
5:0	AUTO_RFR_LEVEL0	See the description of bits 10:8 (AUTO_RFR_LEVEL1).

0x0004 UART_MR2

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
6	ERROR_MODE	This bit controls the operation of the two FIFO status bits for the channel (parity or framing error and received break). ■ When clear (0), the UART operates in character mode and the status bits apply only to the character at the top of the FIFO. ■ When set (1), the UART operates in block mode and both bits are the “OR” of the status for all previously received characters arriving after the last ‘reset error status’ command was issued.
5:4	BITS_PER_CHAR	These bits determine how many bits are transmitted or received per character, not including the start, stop, and parity bits. 00 : 5 bits 01 : 6 bits 10 : 7 bits 11 : 8 bits
3:2	STOP_BIT_LEN	This field programs the duration of the stop bit that is appended to each transmitted character. The stop bit duration can be 9/16, 1, 1 9/16, and 2 bit times. The receiver only samples the stop bit once, regardless of the programmed stop bit length. 00 : 0.563 (9/16) bit times 01 : 1.000 bit time 10 : 1.563 (1 9/16) bit times 11 : 2.000 bit times
1:0	PARITY_MODE	These bits determine which parity mode is used. The user can select between odd, even, space, or no parity. 00 : no parity 01 : odd parity 10 : even parity 11 : space parity

0x0008 UART_CSR

Type: Write
Clock: UART_CLK

The UART_CSR register is the UART clock selection register. This register is used in conjunction with the UART M/N counter registers to determine the bit rate for the transmitter and receiver. The transmitter and receiver can be clocked at different rates. When MISC_CLK_SEL1: UART1_CLK_SRC_SEL is 00 (binary), the bit rates are the values shown below in the “bit rate” row multiplied by 4.

Bits	Name	Description
7:4	UART_RX_CLK_SEL	Use the CLK SEL values in Table 23-1 to select the appropriate receive and transmit bit rates.
3:0	UART_TX_CLK_SEL	

Table 23-1 lists the hexadecimal values for the clock select field and the corresponding data rates.

Table 23-1 CLK SEL values (hex) and corresponding bit rates for TCXO/4

CLK SEL value	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Bit rate (b/sec)	75	150	300	600	1200	2400	3600	4800	7200	9600	14.4k	19.2k	28.8k	38.4k	57.6k	115.2k

0x000C UART_TF

Type: Write
Clock: UART_CLK

Bits	Name	Description
7:0	UART_TF	The UART_TF register is the UART transmit FIFO register. This register is used to access the channel transmit FIFO. The character written is not sent until the transmitter loads it into the transmit shift register and sends it out. If the FIFO is full at the time of writing, the character is lost. This register is updated asynchronously.

0x0010 UART_CR

Type: Write
Clock: UART_CLK

The UART_CR register is the UART command register. This register is used to issue specific commands to the UART subsystem. This register is updated asynchronously.

CAUTION Do not reset the transmitter and disable it at the same time. Do not reset the receiver and disable it at the same time.

Bits	Name	Description
7:4	CHANNEL_COMMAND	Writing the appropriate value to these bits executes the commands that are listed in Table 23-2.
3	UART_TX_DISABLE	This command terminates the operation of the transmitter after any character in the transmit shift register is sent.
2	UART_TX_EN	This command enables the operation of the transmitter.
1	UART_RX_DISABLE	This command immediately terminates the operation of the channel receiver and any incoming characters are lost. None of the receiver status bits are affected by this command and characters that are already in the receive FIFO remain there.
0	UART_RX_EN	This command enables the channel receiver.

Table 23-2 CHANNEL_COMMAND bit value descriptions

Value	Description	Result
0	Null command	Does nothing.
1	Reset receiver	Resets the receiver as if a hardware reset were issued. The receiver is disabled and the FIFO is flushed.
2	Reset transmitter	Resets the transmitter as if a hardware reset were issued. The transmitter signal goes high (marking) and the FIFO is flushed.
3	Reset error status	Clears the overrun error and hunt char received status bits in both the character and block error modes. In the block error mode, it clears the error status and received break.
4	Reset break change interrupt	Clears the break change interrupt status bit.
5	Start break	Forces the transmitter signal low. The transmitter must be enabled. If the transmitter is busy, the break is started when all characters in the transmit FIFO and the transmit shift register have been completely sent.
6	Stop break	If executed while channel is breaking, this command causes the transmitter signal to go high. The signal remains high for at least one bit time before sending out a new character.
7	Reset CTS_N	Clears ISR bit 5.
8	Undefined	

Table 23-2 CHANNEL_COMMAND bit value descriptions (Continued)

Value	Description	Result
9	Packet mode	Turns on the sample data mode, which causes the receiver to sample the receive data stream at 16 times the programmed baud rate. The data is sampled with the start of the start bit, or the first data bit, and continued until the marking state. To exit this state, write 1100 in the command field.
C	Mode reset	Turns off the sample data mode.
D	Set RFR_N	Asserts the ready for receiving signal (active low).
E	Reset RFR_ND	Deasserts the ready for receiving signal.

0x0014 UART_IMR

Type: Write
Clock: UART_CLK

The UART_IMR register is the UART interrupt mask register. This register is used to enable the corresponding functions in the UART_ISR register. Setting (1) a bit in the UART_IMR register causes an interrupt to be generated, if the corresponding bit in the UART_ISR register is set. Clearing (0) a bit in the UART_IMR register causes the setting of the corresponding bit in the UART_ISR register to have no effect on the interrupt signal.

Bit 6 of the UART_IMR register, CURRENT_CTS, is slightly different. If the current value of the CTS is one (1), no interrupt is generated. However, the user can use bit 6 in this register to mask the CTS value when reading the UART_MISR register or as a general-purpose bit.

UART_IMR

Bits	Name	Description
6	CURRENT_CTS	This bit indicates the current state of the CTS input. It never generates an interrupt.
5	DELTA_CTS	This bit, when set (1), indicates that the CTS input has changed states. To clear the detection logic associated with this function, write CR[7:4]=0x7.
4	RXLEV	This bit is set when a character is loaded into the receive FIFO that brings the total number of characters in the FIFO up to the programmed watermark level in the FIFO watermark register (RFWR). This bit is cleared after enough characters have been read to bring the level equal to or below the programmed watermark level.

UART_IMR (Continued)

Bits	Name	Description
3	RXSTALE	This bit shows that there are one or more characters in the Rx FIFO (but less than the level defined in RFWR) and they have been there longer than the time-out value specified by the UART_IPR register. After this bit is set (1), an appropriate time-out value is then written to the STALE_TIMEOUT_LSB bits of the UART_IPR register. The STALE_TIME_OUT value specifies how many character times must elapse before a stale character time-out is generated. In this instance, a character time is defined as 10 times the bit rate. The RXSTALE interrupt status bit clears each time a character is read from the Rx FIFO. The stale character interrupt duration is measured from either the first or the last arriving character in the Rx FIFO as programmed by the RXSTALE_LAST bit (bit 5) in the UART_IPR register. When the RXSTALE_LAST bit is clear (0), the value that is programmed into the STALE_TIMEOUT bits is usually greater than the value programmed in RFWR. If the value is not greater, then STALE_TIMEOUT expires before the RFWR setpoint.
2	RXBREAK	This bit is set (1) if the start or end of a break condition is detected. The logic associated with this condition is cleared (0) by writing CR[7:4]=0x4. A detected break is defined as an all-zeros character with an invalid stop bit AND LOW PARITY. The end of a break is defined as a mark condition (1) at least 1/2 bit width. A detected break occupies one position in the Rx FIFO.
1	RXHUNT	This bit is set (1) when an incoming character matches the value programmed into the UART_HCR register. This condition is cleared (0) by issuing a reset error status command (CR[7:4]=0x3).
0	TXLEV	This bit is set (1) when a character which is transferred from the transmit FIFO to the transmit shift register brings the total number of characters in the transmit FIFO equal to or below the programmed watermark level in the UART_TFWR register. This bit is cleared (0) after enough characters have been written to the transmit FIFO to bring the level above the programmed watermark level.

0x0018 UART_IPR

Type: Read/Write
Clock: UART_CLK

The UART_IPR register is the UART interrupt programming register.

Bits	Name	Description
10:7	STALE_TIMEOUT_MSB	These bits are the STALE_TIMEOUT bitfield. The stale character time-out duration field contains a number from 1 to 511. This number determines how many character times must elapse before a stale character time-out is generated. This character time is defined as 10 times the bit rate. Do not clear (0) this register if the stale character time-out interrupt is enabled. Normally, the stale character time-out is greater than the value in RFWR. When RXSTALE_LAST is cleared, the value programmed into STALE_TIMEOUT is usually greater than the value programmed in RFWR. Otherwise the STALE_TIMEOUT always expires prior to the RFWR setpoint. The STALE_TIMEOUT field is 6 bits long. Note the discontinuity in the bit assignments.
6	SAMPLE_DATA	Setting (1) this bit enables the new sample data mode, which means that the start bit is sampled as well as the rest, when in sample data mode. See the UART_CR register, CHANNEL_COMMAND bit for more information.
5	RXSTALE_LAST	This bit controls the method for calculating the RX_STALE character time-out interrupt. ■ Set (1) this bit to measure the RX stale character time-out duration from the end of the last character arriving in the RX FIFO. ■ Clear (0) this bit to measure the RX stale character time-out duration from the end of the first character arriving in the RX FIFO. This bit is valid for all devices which return a value greater than or equal to 0x11.
4:0	STALE_TIMEOUT_LSB	This bitfield is the LSbits of the STALE_TIMEOUT bitfield.

0x001C UART_TFWR

Type: Read/Write
Clock: UART_CLK

The UART_TFWR register is the UART transmit FIFO watermark register.

Bits	Name	Description
8:0	TFW	These bits contain a number, between 0 and 511, that determines the level of the transmit FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of TX characters in the FIFO) is less than or equal to the value in TFWR. For example: If this register is programmed with the value 5, a TXLEV interrupt is issued when five characters or less are left in the TX FIFO.

0x0020 UART_RFWR

Type: Read/Write
Clock: UART_CLK

The UART_RFWR register is the UART receive FIFO watermark register.

Bits	Name	Description
8:0	RFW	These bits contain a number, between 0 and 511, that determines the level of the receive FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of characters in the RX FIFO) is greater than the value in RFWR. For example: If this register is programmed with the value 5 then an RXLEV interrupt is issued when six characters are received.

0x0024 UART_HCR

Type: Read/Write
Clock: UART_CLK

.

Bits	Name	Description
7:0	DATA	The UART_HCR register is the UART hunt character register, which contains the character for which the receiver looks to assert a hunt character received interrupt.

0x0028 UART_MREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART_MREG register is the M value register, which contains the 8 most significant bits of the M value for the system clock that generates the M/N counter. The optimal M value for the present clock scheme is 12.The recommended programming value is 0x6 when the clk source is at TCXO =19.2MHz. The recommended programming value is 0x18 when the clock source is at TCXO/4 = 4.8MHz.See UART_CSR on page 23-4.

0x002C UART_NREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	UART_NREG	The UART_NREG register is the N value register, which contains the 8 most significant bits of the 1's complement of N–M for the system clock that generates the M over N counter. The optimal N value for the present clock scheme is 125. The recommended programming value is 0xF1 when the clk source is at TCXO = 19.2MHz. The recommended programming value is 0xF6 when the clk source is at TCXO/4 = 4.8MHz.

0x0030 UART_DREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART_DREG register is the D value register, which contains the 8 most significant bits of the D value for the system clock that generates the M/N counter. The optimal D value for the present clock scheme is 62. The recommended programming value is 0xF when the clk source is at TCXO=19.2MHz as well as TCXO/4 = 4.8MHz.

0x0034 UART_MNDREG

Type: Read/Write
Clock: UART_CLK

The UART_MNDREG register is the M, N, and D extra value register. This register contains the least significant bits for the M, N, and D registers of the M/N counter. The recommended programming value is 0x1A when the clk source is at TCXO = 19.2MHz. The recommended programming value is 0xA when the clk source is at TCXO/4 = 4.8MHz.

Bits	Name	Description
5	MREG_LSB	See UART_MREG on page 23-10.
4:2	NREG_LSB	See UART_NREG on page 23-10.
1:0	DREG_LSB	See UART_DREG on page 23-10

0x0038 UART_IRDA

Type: Write
Clock: UART_CLK

The UART_IRDA register enables the IRDA function, selects a loopback, and allows inversion of RX and TX data. This register also controls the IRDA transceiver that optionally interfaces between the UART and the DP_RX_DATA and DP_TX_DATA pins.

The UART proper is a standard RX/TX configuration that converts the serial pin input and output lines to 8-bit data for the microprocessor. The configuration contains receive and transmit blocks, a control block, a bit rate generator for RX and TX, and FIFO interfaces with the microprocessor data bus. The IRDA register provides the means of switching IRDA circuits into the serial I/O lines and the IRDA function is normally switched out. The IRDA modulator/demodulator may be enabled or disabled under microprocessor control. When enabled, the RX and TX data paths have individual inversion select bits, so that external true or inverted drivers may be used in the signal path.

UART_IRDA

Bits	Name	Description
3	IRDA_LOOPBACK	■ This bit loops the IRDA modulated TX line back into the IRDA RX line. The normal UART loopback mode must be off in order to see this effect.
2	INVERT_IRDA_TX	This bit inverts the polarity of the IRDA modulated signal on the DP_TX_DATA pin. ■ Set (1) this bit for an inverted polarity. ■ Clear (0) this bit for a non-inverted polarity.

UART_IRDA (Continued)

1	INVERT_IRDA_RX	This bit inverts the polarity of the IRDA received signal on the DP_RX_DATA pin. ■ Set (1) this bit for inverted the polarity. ■ Clear (0) this bit for non-inverted polarity.
0	IRDA_EN	■ Set (1) this bit to enable the IRDA transceiver. ■ Clear (0) this bit to bypass the IRDA transceiver and ignore the other bits of this register.

23.2.1.2 Read registers

NOTE The addresses of the read-only registers are mapped into the same addresses of the 4 write-only registers in the section above this one. They are: UART_CSR, UART_TF, UART_CR, and UART_IMR, respectively.

0x0008 UART_SR

Type: Read
Clock: UART_CLK

The UART_SR register is the UART status register. This register is used to obtain the current state of the UART subsystem. This register is updated asynchronously.

UART_SR

Bits	Name	Description
7	HUNT_CHAR	When set (1), this bit indicates that the programmed character in the UART_HCR register was received. If programmed, this condition causes the RXHUNT bit to be set (1) in the UART_ISR register. This bit is cleared by issuing a reset error status command.
6	RX_BREAK	When set (1), this bit indicates that an all-zero character of the programmed length was received without a stop bit. If a break begins in the middle of a character, it is detected if the condition remains until the end of the next character time. Only one all zero character is written into the FIFO upon receiving the break. After the RX input returns high for at least half a bit time, then more characters are accepted by the receiver. Whenever the break condition starts or stops as defined above, the RXBREAK bit is set (1) in the UART_ISR register. This bit is appended to the character in the receive FIFO.
5	PAR_FRAME_ERR	A parity error occurs if parity is programmed to be checked and an incoming character is received with incorrect parity. A framing error occurs when the RX input is low while the stop bit is being sampled. This bit is sampled in the middle of the first stop bit position, regardless of the programmed length of the stop bit. This error bit is appended to the character in the receive FIFO.

UART_SR (Continued)

Bits	Name	Description
4	UART_OVERRUN	An overrun error occurs if one or more incoming characters are lost. This happens when the receive FIFO is full, a full character has been received and is waiting to be written to the FIFO, and a new start bit is detected. The character in the receive shift register is lost. Upon overrun, this status bit remains high until cleared by a reset error status command.
3	TXEMT	This bit is set (1) when the transmitter under-runs. It is set after the transmission of the stop bit at the end of a character, if there are no characters waiting to be sent. It is reset when a character is written to the transmit FIFO.
2	TXRDY	This bit indicates that the transmit FIFO has space in it. It is reset when the FIFO becomes full and set when space becomes available again. Any characters that are written to the FIFO while this bit is low is lost.
1	RXFULL	This bit is set (1) when the receive FIFO becomes full. It is reset when the CPU reads a character, if there is not a character waiting in the receive shift register that takes the read character's place.
0	RXRDY	This bit is set (1) when there is a character in the receive FIFO waiting to be read. It is reset when the last character currently stored in the FIFO is read.

0x000C UART_RF

Type: Read
Clock: UART_CLK

The UART_RF register is the UART receive FIFO register, which is used to access the channel receive FIFO.

Bits	Name	Description
9	RXBREAK	This bit is set (1) when a break condition is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 6, for more information on break conditions.
8	PAR_FRAME_ERR	This bit is set (1) when a parity or frame error is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 5, for more information on parity and frame errors.
7:0	UART_RF	This register returns the next value in the receive FIFO. After the read is completed, the FIFO read pointer is updated to make the next character available. This register is updated asynchronously.

0x0010 UART_MISR

Type: Read
Clock: UART_CLK

Bits	Name	Description
6:0	UART_MISR	The UART_MISR register is the masked interrupt status register. A read from this register returns the “AND” of the UART_ISR and the UART_IMR registers. This register is updated asynchronously.This bit field is misr (6:0) <= imr(6:0) AND isr (6:0).

0x0014 UART_ISR

Type: Read
Clock: UART_CLK

The UART_ISR register is the UART interrupt status register. This register provides the current status of the possible interrupt conditions. If one of these bits is set (1), and the corresponding bit in the UART_IMR register is set (1), an interrupt is generated (with the exception of bit 6, CURRENT_CTS – see the description of the UART_IMR register). If the corresponding bit in the UART_IMR register is clear (0), setting one of these bits has no effect on the UART interrupt request signal. This register is updated asynchronously.

Bits	Name	Description
6	CURRENT_CTS	See UART_IMR on page 23-6, for descriptions of the UART_ISR bits.
5	DELTA_CTS	
4	RXLEV	
3	RXSTALE	
2	RXBREAK	
1	RXHUNT	
0	TXLEV	

23.2.2 UART2 registers

This section contains the read and write registers for UART2.

23.2.2.1 Write and read/write registers

0x0000 UART2_MR1

Type: Read/Write
Clock: UART_CLK

The UART2_MR1 register is the UART2 mode register #1. This register is used, along with UART2_MR2, to configure the operational mode of the UART2.

Bits	Name	Description
7	RX_RDY_CTL	Setting (1) this bit enables the automatic ready-for-receiving (RFR_N) control for the receiver. RFR_N is turned off, or set (1), when a valid start bit is received and the channel FIFO is at the level programmed in bits 4 through 0 of this mode register. When the FIFO level falls below the programmed level, the RFR_N signal is turned on, or clear (0). When this feature is off, the RFR_N signal can be used by normal signal port bit manipulation.
6	CTS_CTL	When clear (0), this feature is disabled and the CTS_N input for the channel has no effect on the transmitter. When set (1), it is enabled, and the transmitter checks the CTS_N input to determine whether to begin transmission of a new character. If CTS_N is low, the character is sent. Otherwise the transmitter continues marking until CTS_N goes low, then the next character is transmitted. A change on CTS_N during the transmission of a character has no effect on that character.
5:0	AUTO_RFR_LEVEL	These bits are used to program the level in the receive FIFO at which the RFR_N signal is deasserted, if programmed to do so. The level can be programmed from 1 to 63. RFR_N is de-asserted when the Rx FIFO level (the number of characters remaining in the Rx FIFO) is greater than the level programmed into this register.

0x0004 UART2_MR2

Type: Read/Write
Clock: UART_CLK

The UART2_MR2 register is the UART2 mode register #2. This register is used, along with UART2_MR1, to configure the operational mode of the UART2

Bits	Name	Description
6	ERROR_MODE	This bit controls the operation of the two FIFO status bits for the channel (parity or framing error and received break). ■ When clear (0), the UART2 operates in character mode and the status bits apply only to the character at the top of the FIFO. ■ When set (1), the UART2 operates in block mode and both bits are the “OR” of the status for all previously received characters arriving after the last ‘reset error status’ command was issued.
5:4	BITS_PER_CHAR	These bits determine how many bits are transmitted or received per character, not including the start, stop, and parity bits. 00 : 5 bits 01 : 6 bits 10 : 7 bits 11 : 8 bits
3:2	STOP_BIT_LEN	This field programs the duration of the stop bit that is appended to each transmitted character. The stop bit duration can be 9/16, 1, 1 9/16, and 2 bit times. The receiver only samples the stop bit once, regardless of the programmed stop bit length. 00 : 0.563 (9/16) bit times 01 : 1.000 bit time 10 : 1.563 (1 9/16) bit times 11 : 2.000 bit times
1:0	PARITY_MODE	These bits determine which parity mode is used. The user can select between odd, even, space, or no parity. 00 : no parity 01 : odd parity 10 : even parity 11 : space parity

0x0008 UART2_CSR

Type: Write
Clock: UART_CLK

The UART2_CSR register is the UART2 clock selection register. This register is used in conjunction with the UART2 M/N counter registers to determine the bit rate for the transmitter and receiver. The transmitter and receiver can be clocked at different rates. When MISC_CLK_SEL1: UART2_CLK_SRC_SEL is 00 (binary), the bit rates are the values shown in the table below in the "bit rate" row multiplied by 4.

Bits	Name	Description
7:4	UART2_RX_CLK_SEL	Use the CLK SEL values in Table 23-3 to select the appropriate receive and transmit bit rates.
3:0	UART2_TX_CLK_SEL	

Table 23-3 lists the hexadecimal values for the clock select field and the corresponding data rates.

Table 23-3 CLK SEL values (hex) and corresponding bit rates

CLK SEL value	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Bit rate (b/sec)	75	150	300	600	1200	2400	3600	4800	7200	9600	14.4k	19.2k	28.8k	38.4k	57.6k	115.2k

0x000C UART2_TF

Type: Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART2_TF register is the UART2 transmit FIFO register, which is used to access the channel transmit FIFO. The character written is not sent until the transmitter loads it into the transmit shift register and sends it out. If the FIFO is full at the time of writing, the character is lost. This register is updated asynchronously.

0x0010 UART2_CR

Type: Write
Clock: UART_CLK

The UART2_CR register is the UART2 command register, which is used to issue specific commands to the UART2 subsystem. This register is updated asynchronously.

CAUTION Do not reset the transmitter and disable it at the same time. Do not reset the receiver and disable it at the same time.

UART2_CR

Bits	Name	Description
8:4	CHANNEL_COMMAND	Writing the appropriate value to these bits executes the commands that are listed in Table 23-4.
3	UART2_TX_DISABLE	This bit terminates the operation of the transmitter after any character in the transmit shift register is sent.
2	UART2_TX_EN	This bit enables the operation of the transmitter.
1	UART2_RX_DISABLE	This bit immediately terminates the operation of the channel receiver and any incoming characters are lost. None of the receiver status bits are affected by this command and characters that are already in the receive FIFO remain there.
0	UART2_RX_EN	This bit enables the channel receiver.

Table 23-4 CHANNEL_COMMAND bit value descriptions

Value	Description	Result
00	Null command	Does nothing.
01	Reset receiver	Resets the receiver as if a hardware reset were issued. The receiver is disabled and the FIFO is flushed.
02	Reset transmitter	Resets the transmitter as if a hardware reset were issued. The transmitter signal goes high (marking) and the FIFO is flushed.
03	Reset error status	Clears the overrun error and hunt char received status bits in both the character and block error modes. In the block error mode, it clears the error status and received break.
04	Reset break change interrupt	Clears the break change interrupt status bit.
05	Start break	Forces the transmitter signal low. The transmitter must be enabled. If the transmitter is busy, the break is started when all characters in the transmit FIFO and the transmit shift register have been completely sent.
06	Stop break	If executed while channel is breaking, this command causes the transmitter signal to go high. The signal remains high for at least one bit time before sending out a new character.
07	Reset CTS_N	Clears ISR bit 5.
08	Reset TX_ERROR	Clears TX_ERROR
09	Packet mode	Turns on the sample data mode, which causes the receiver to sample the receive data stream at 16 times the programmed baud rate. The data is sampled with the start of the start bit, or the first data bit, and continued until the marking state. To exit this state, write 1100 in the command field.
0C	Mode reset	Turns off the sample data mode.
0D	Set RFR_N	Asserts the ready for receiving signal (active low).
0E	Reset RFR_ND	Deasserts the ready for receiving signal.
10	Clear TX_DONE	Clears the TX_DONE interrupt (ISR bit 8)

0x0014 UART2_IMR

Type: Write
Clock: UART_CLK

The UART2_IMR register is the UART2 interrupt mask register. This register is used to enable the corresponding functions in the UART2_ISR register.

- Setting (1) bit 8:7 and bit 5:0 in this register causes an interrupt to be generated, if the corresponding bit in the UART2_ISR register is set.
- Clearing (0) bit 8:7 and bit 5:0 in this register causes the setting of the corresponding bit in the UART2_ISR register to have no effect on the interrupt signal.

Bit 6 of this register, CURRENT_CTS, is slightly different. If the current value of the CTS is one (1), no interrupt is generated. However, the user can use bit 6 in this register to mask the CTS value when reading the UART2_MISR register or as a general-purpose bit.

UART2_IMR

Bits	Name	Description
8	TX_DONE	This bit is used only in UIM/SIM_IF mode. Set (1) this bit when the last byte in the transmitter has been sent.
7	TX_ERROR	Only used in UIM_IF mode. Set (1) this bit to indicate that there has been parity error during transmission. To clear the detection logic associated with this function, write CR[7:4]=0x8.
6	CURRENT_CTS	This bit indicates the current state of the CTS input. It never generates an interrupt.
5	DELTA_CTS	Set (1),this bit to indicate that the CTS input has changed states. To clear the detection logic associated with this function, write CR[7:4]=0x7.
4	RXLEV	■ Set (1) this bit when a character is loaded into the receive FIFO that brings the total number of characters in the FIFO up to the programmed watermark level in the FIFO watermark register (RFWR). ■ Clear (0) this bit after enough characters have been read to bring the level equal to or below the programmed watermark level.

UART2_IMR (Continued)

Bits	Name	Description
3	RXSTALE	This bit indicates that there are one or more characters in the Rx FIFO (but less than the level defined in RFWR) and they have been there longer than the time-out value that is specified by the UART2_IPR register. After this bit is set (1), an appropriate time-out value is then written to the STALE_TIMEOUT_LSB bits of the UART2_IPR register. The STALE_TIME_OUT value specifies how many character times must elapse before a stale character time-out is generated. In this instance, a character time is defined as 10 times the bit rate. The RXSTALE interrupt status bit clears each time a character is read from the Rx FIFO. The stale character interrupt duration is measured from either the first or the last arriving character in the Rx FIFO as programmed by the RXSTALE_LAST bit (bit 5) in the UART2_IPR register. When the RXSTALE_LAST bit is clear (0), the value that is programmed into the STALE_TIMEOUT bits is usually greater than the value programmed in RFWR. If the value is not greater, then STALE_TIMEOUT expires before the RFWR setpoint.
2	RXBREAK	This bit is set (1) if the start or end of a break condition is detected. The logic associated with this condition is cleared (0) by writing CR[7:4]=0x4. A detected break is defined as an all-zeros character with an invalid stop bit AND LOW PARITY. The end of a break is defined as a mark condition (1) at least 1/2 bit width. A detected break occupies one position in the Rx FIFO.
1	RXHUNT	This bit is set (1) when an incoming character matches the value programmed into the UART2_HCR register. This condition is cleared (0) by issuing a reset error status command (CR[7:4]=0x3).
0	TXLEV	■ Set (1) this bit when a character which is transferred from the transmit FIFO to the transmit shift register brings the total number of characters in the transmit FIFO equal to or below the programmed watermark level in the UART2_TFWR register. ■ Clear (0) this bit after enough characters have been written to the transmit FIFO to bring the level above the programmed watermark level.

0x0018 UART2_IPR

Type: Read/Write
Clock: UART_CLK

The UART2_IPR register is the UART2 interrupt programming register.

Bits	Name	Description
7	STALE_TIMEOUT_MSB	This bit is the MSbit of the STALE_TIMEOUT bitfield. The stale character time-out duration field contains a number from 1 to 63. This number determines how many character times must elapse before a stale character time-out is generated. This character time is defined as 10 times the bit rate. Do not clear (0) this register if the stale character time-out interrupt is enabled. Normally, the stale character time-out is greater than the value in RFWR. When RXSTALE_LAST is cleared, the value programmed into STALE_TIMEOUT is usually greater than the value programmed in RFWR. Otherwise the STALE_TIMEOUT always expires prior to the RFWR setpoint. The STALE_TIMEOUT field is 6 bits long. Note the discontinuity in the bit assignments.
6	SAMPLE_DATA	Setting (1) this bit enables the new sample data mode, which means that the start bit is sampled as well as the rest, when in sample data mode. See the UART2_CR register, CHANNEL_COMMAND bit for more information.
5	RXSTALE_LAST	This bit controls the method for calculating the RX_STALE character time-out interrupt. ■ Set (1) this bit to measure the RX stale character time-out duration from the end of the last character arriving in the RX FIFO. ■ Clear (0) this bit to measure the RX stale character time-out duration from the end of the first character arriving in the RX FIFO. This bit is valid for all devices which return a value greater than or equal to 0x11.
4:0	STALE_TIMEOUT_LSB	This bitfield is the LSbits of the STALE_TIMEOUT bitfield.

0x001C UART2_TFWR

Type: Read/Write
Clock: UART_CLK

The UART2_TFWR register is the UART2 transmit FIFO watermark register.

Bits	Name	Description
5:0	TFW	These bits contain a number, between 0 and 63, that determines the level of the transmit FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of TX characters in the FIFO) is less than or equal to the value in TFWR. For example: If this register is programmed with the value 5, a TXLEV interrupt is issued when five characters or less are left in the TX FIFO.

0x0020 UART2_RFWR

Type: Read/Write
Clock: UART_CLK

The UART2_RFWR register is the UART2 receive FIFO watermark register.

Bits	Name	Description
5:0	RFW	These bits contain a number, between 0 and 63, that determines the level of the receive FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of characters in the RX FIFO) is greater than the value in RFWR. For example: If this register is programmed with the value 5 then an RXLEV interrupt is issued when six characters are received.

0x0024 UART2_HCR

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART2_HCR register is the UART2 hunt character register. This register contains the character for which the receiver looks to assert a hunt character received interrupt.

0x0028 UART2_MREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART2_MREG register is the M value register, which contains the 8 most significant bits of the M value for the system clock that generates the M/N counter. The optimal M value for the present clock scheme is 12. The recommended programming value is 0x6 when the clk source is at TCXO =19.2MHz. The recommended programming value is 0x18 when the clock source is at TCXO/4 = 4.8MHz.See UART2_CSR on page 23-17.

0x002C UART2_NREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART2_NREG register is the N value register. This register contains the 8 most significant bits of the 1's complement of N–M for the system clock that generates the M over N counter. The optimal N value for the present clock scheme is 125. The recommended programming value is 0xF1 when the clk source is at TCXO=19.2MHz. The recommended programming value is 0xF6 when the clk source is at TCXO/4 = 4.8MHz.

0x0030 UART2_DREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART2_DREG register is the D value register. This register contains the 8 most significant bits of the D value for the system clock that generates the M/N counter. The optimal D value for the present clock scheme is 62. The recommended programming value is 0xF when the clk source is at TCXO=19.2MHz as well as TCXO/4 = 4.8MHz.

0x0034 UART2_MNDREG

Type: Read/Write
Clock: UART_CLK

The UART2_MNDREG register is the M, N, and D extra value register. This register contains the least significant bits for the M, N, and D registers of the M/N counter. The recommended programming value is 0x1A when the clk source is at TCXO = 19.2MHz. The recommended programming value is 0xA when the clk source is at TCXO/4 = 4.8MHz.

Bits	Name	Description
5	MREG_LSB	See the description for the UART2_MREG register.
4:2	NREG_LSB	See the description for the UART2_NREG register.
1:0	DREG_LSB	See the description for the UART2_DREG register.

0x0038 UART2_IRDA

Type: Write
Clock: UART_CLK

The UART2_IRDA register enables the IRDA function, selects a loopback, and allows inversion of RX and TX data. This register also controls the IRDA transceiver that optionally interfaces between the UART2 and the DP_RX_DATA and DP_TX_DATA pins.

The UART2 proper is a standard RX/TX configuration that converts the serial pin input and output lines to 8-bit data for the microprocessor. The configuration contains receive and transmit blocks, a control block, a bit rate generator for RX and TX, and FIFO interfaces with the microprocessor data bus. The IRDA register provides the means of switching IRDA circuits into the serial I/O lines and the IRDA function is normally switched out. The IRDA modulator/demodulator may be enabled or disabled under microprocessor control. When enabled, the RX and TX data paths have individual inversion select bits, so that external true or inverted drivers may be used in the signal path.

UART2_IRDA

Bits	Name	Description
3	IRDA_LOOPBACK	■ This bit loops the IRDA modulated TX line back into the IRDA RX line. The normal UART2 loopback mode must be off in order to see this effect.
2	INVERT_IRDA_TX	This bit inverts the polarity of the IRDA modulated signal on the DP_TX_DATA pin. ■ Set (1) this bit for an inverted polarity. ■ Clear (0) this bit for a non-inverted polarity.

UART2_IRDA (Continued)

1	INVERT_IRDA_RX	This bit inverts the polarity of the IRDA received signal on the DP_RX_DATA pin. ■ Set (1) this bit for inverted the polarity. ■ Clear (0) this bit for non-inverted polarity.
0	IRDA_EN	■ Set (1) this bit to enable the IRDA transceiver. ■ Clear (0) this bit to bypass the IRDA transceiver and ignore the other bits of this register.

0x003C UART2_SIM_CFG

Type: Write
Clock: UART_CLK

The UART2_SIM_CFG register is used to configure the SIM interface for the UART2.

UART2_SIM_CFG

Bits	Name	Description
17	UIM_TX_MODE	When this bit is set (1), the transmission portion of the SIM interface operates in block mode (T=1). When clear (0), the transmission portion of the SIM interface operates in asynchronous transfer mode (T=0). Note that the operation of this function is conditional upon UART2_SIM_CFG[0] being set (1) to designate the UIM_IF mode of operation.
16	UIM_RX_MODE	When this bit is set (1), the receive portion of the SIM interface operates in block mode (T=1). When clear (0), the receive portion of the SIM interface operates in asynchronous transfer mode (T=0). Note that the operation of this function is conditional upon UART2_SIM_CFG[0] being set (1) to designate the UIM_IF mode of operation.
15:8	SIM_STOP_BIT_LEN	In UIM_IF mode, this field programs the duration of the stop bit that is appended to each transmitted character. The stop bit duration can be 1 to 254 bit times. The receiver only samples the stop bit once, regardless of the programmed stop bit length. 00000001 : 1 bit times 00000010 : 2 bit times ..... 11 111110: 254 bit times
7	SIM_CLK_ON	When this bit is set(1), the UIM_CLK is turned on at either the TD8 or TD4 frequency, as indicated by bit6(UIM_CLK_TD8_SEL). When this bit is cleared, the UIM_CLK is stopped either LOW or HIGH, as indicated by bit 5(UIM_CLK_STOP_HIGH).
6	SIM_CLK_TD8_SEL	This bit selects the UIM_CLK frequency at which the UIM_CLK is turned on. 1 : TD8 (the UIM_CLK runs at the TD8 frequency) 0 : TD4 (the UIM_CLK runs at the TD4 frequency)

UART2_SIM_CFG (Continued)

Bits	Name	Description
5	SIM_CLK_STOP_HIGH	This bit indicates at what point - high or low - the UIM_CLK will stop. 1 : high 0 : low
4	SIM_CLK_SEL	unused
3	MASK_RX	Setting (1) this bit masks off any 0 on the RX line, which prevents the receiver from seeing the START bit. Set (1) this bit if you do not want to receive the data you transmit.
2	SWAP_D	Setting (1) this bit causes the UIM_IF to swap the 8 data bits (making MSB into LSB, bit 6 into bit1, and so on) before it is read by the microprocessor.
1	INV_D	Set (1) this bit to cause the UIM_IF to invert the 8 data bit before it is read by the microprocessor.
0	SIM_SEL	Set (1) this bit to designate the UIM_IF mode of operation.

23.2.2.2 Read registers

NOTE The addresses of the read-only registers are mapped into the same addresses of the 4 write-only registers in the section above this one. They are: UART2_CSR, UART2_TF, UART2_CR, and UART2_IMR, respectively.

0x0008 UART2_SR

Type: Read
Clock: UART_CLK

The UART2_SR register is the UART2 status register. This register is used to obtain the current state of the UART2 subsystem. This register is updated asynchronously.

UART2_SR

Bits	Name	Description
7	HUNT_CHAR	When set (1), this bit indicates that the programmed character in the UART2_HCR register was received. If programmed, this condition causes the RXHUNT bit to be set (1) in the UART2_ISR register. This bit is cleared by issuing a reset error status command.
6	RX_BREAK	When set (1), this bit indicates that an all-zero character of the programmed length was received without a stop bit. If a break begins in the middle of a character, it is detected if the condition remains until the end of the next character time. Only one all zero character is written into the FIFO upon receiving the break. After the RX input returns high for at least half a bit time, then more characters are accepted by the receiver. Whenever the break condition starts or stops as defined above, the RXBREAK bit is set (1) in the UART2_ISR register. This bit is appended to the character in the receive FIFO.
5	PAR_FRAME_ERR	A parity error occurs if parity is programmed to be checked and an incoming character is received with incorrect parity. A framing error occurs when the RX input is low while the stop bit is being sampled. This bit is sampled in the middle of the first stop bit position, regardless of the programmed length of the stop bit. This error bit is appended to the character in the receive FIFO.
4	UART2_OVERRUN	An overrun error occurs if one or more incoming characters are lost. This happens when the receive FIFO is full, a full character has been received and is waiting to be written to the FIFO, and a new start bit is detected. The character in the receive shift register is lost. Upon overrun, this status bit remains high until cleared by a reset error status command.
3	TXEMT	This bit is set when the transmitter under-runs. It is set after the transmission of the stop bit at the end of a character, if there are no characters waiting to be sent. It is reset when a character is written to the transmit FIFO.
2	TXRDY	This bit indicates that the transmit FIFO has space in it. It is reset when the FIFO becomes full and set when space becomes available again. Any characters that are written to the FIFO while this bit is low is lost.
1	RXFULL	This bit is set when the receive FIFO becomes full. It is reset when the CPU reads a character, if there is not a character waiting in the receive shift register that takes the read character's place.
0	RXRDY	This bit is set when there is a character in the receive FIFO waiting to be read. It is reset when the last character currently stored in the FIFO is read.

0x000C UART2_RF

Type: Read
Clock: UART_CLK

Bits	Name	Description
9	RXBREAK	This bit is set (1) when a break condition is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 6, for more information on break conditions.
8	PAR_FRAME_ERR	This bit is set (1) when a parity or frame error is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 5, for more information on parity and frame errors.
7:0	DATA	The UART2_RF register is the receive FIFO register. This register returns the next value in the receive FIFO. After the read is completed, the FIFO read pointer is updated to make the next character available. This register is updated asynchronously.

0x0010 UART2_MISR

Type: Read
Clock: UART_CLK

Bits	Name	Description
8:0	UART2_MISR	The UART_MISR register is the masked interrupt status register. A read from this register returns the “AND” of the UART_ISR and the UART_IMR registers. This register is updated asynchronously.This bit field is misr (7:0) <= imr(7:0) AND isr (7:0).

0x0014 UART2_ISR

Type: Read
Clock: UART_CLK

The UART2_ISR register is the UART2 interrupt status register. This register provides the current status of the possible interrupt conditions.

- If one of these bits is set (1), and the corresponding bit in the UART2_IMR register is set (1), an interrupt is generated (with the exception of bit 6, CURRENT_CTS – see the description of the UART2_IMR register).

- If the corresponding bit in the UART2_IMR register is clear (0), setting one of these bits has no effect on the UART2 interrupt request signal. This register is updated asynchronously.

UART2_ISR

Bits	Name	Description
8	TX_DONE	See UART2_IMR on page 23-19 for descriptions of the UART2_ISR bits.
7	TX_ERROR	
6	CURRENT_CTS	
5	DELTA_CTS	
4	RXLEV	
3	RXSTALE	
2	RXBREAK	
1	RXHUNT	
0	TXLEV	

23.2.3 UART3 registers

This section contains the read and write registers for UART3.

23.2.3.1 Write and read/write registers

0x0000 UART3_MR1

Type: Read/Write
Clock: UART_CLK

The UART3_MR1 register is the UART3 mode register #1. This register is used, along with UART3_MR2, to configure the operational mode of the UART3.

Bits	Name	Description
7	RX_RDY_CTL	Setting (1) this bit enables the automatic ready-for-receiving (RFR_N) control for the receiver. RFR_N is turned off, or set (1), when a valid start bit is received and the channel FIFO is at the level programmed in bits 4 through 0 of this mode register. When the FIFO level falls below the programmed level, the RFR_N signal is turned on, or clear (0). When this feature is off, the RFR_N signal can be used by normal signal port bit manipulation.
6	CTS_CTL	When clear (0), this feature is disabled and the CTS_N input for the channel has no effect on the transmitter. When set (1), it is enabled, and the transmitter checks the CTS_N input to determine whether to begin transmission of a new character. If CTS_N is low, the character is sent. Otherwise the transmitter continues marking until CTS_N goes low, then the next character is transmitted. A change on CTS_N during the transmission of a character has no effect on that character.
5:0	AUTO_RFR_LEVEL	These bits are used to program the level in the receive FIFO at which the RFR_N signal is deasserted, if programmed to do so. The level can be programmed from 1 to 63. RFR_N is de-asserted when the Rx FIFO level (the number of characters remaining in the Rx FIFO) is greater than the level programmed into this register.

0x0004 UART3_MR2

Type: Read/Write
Clock: UART_CLK

The UART3_MR2 register is the UART3 mode register #2. This register is used, along with UART3_MR1, to configure the operational mode of the UART3

Bits	Name	Description
6	ERROR_MODE	This bit controls the operation of the two FIFO status bits for the channel (parity or framing error and received break). ■ When clear (0), the UART3 operates in character mode and the status bits apply only to the character at the top of the FIFO. ■ When set (1), the UART3 operates in block mode and both bits are the “OR” of the status for all previously received characters arriving after the last ‘reset error status’ command was issued.
5:4	BITS_PER_CHAR	These bits determine how many bits are transmitted or received per character, not including the start, stop, and parity bits. 00 : 5 bits 01 : 6 bits 10 : 7 bits 11 : 8 bits
3:2	STOP_BIT_LEN	This field programs the duration of the stop bit that is appended to each transmitted character. The stop bit duration can be 9/16, 1, 1 9/16, and 2 bit times. The receiver only samples the stop bit once, regardless of the programmed stop bit length. 00 : 0.563 (9/16) bit times 01 : 1.000 bit time 10 : 1.563 (1 9/16) bit times 11 : 2.000 bit times
1:0	PARITY_MODE	These bits determine which parity mode is used. The user can select between odd, even, space, or no parity. 00 : no parity 01 : odd parity 10 : even parity 11 : space parity

0x0008 UART3_CSR

Type: Write
Clock: UART_CLK

The UART3_CSR register is the UART3 clock selection register. This register is used in conjunction with the UART3 M/N counter registers to determine the bit rate for the transmitter and receiver. The transmitter and receiver can be clocked at different rates. When MISC_CLK_SEL1: UART3_CLK_SRC_SEL is 00(binary), the bit rates are the values shown in the table below in the "bit rate" row multiplied by 4.

Bits	Name	Description
7:4	UART3_RX_CLK_SEL	Use the CLK SEL values in Table 23-3 to select the appropriate receive and transmit bit rates.
3:0	UART3_TX_CLK_SEL	

Table 23-3 lists the hexadecimal values for the clock select field and the corresponding data rates.

Table 23-5 CLK SEL values (hex) and corresponding bit rates

CLK SEL value	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Bit rate (b/sec)	75	150	300	600	1200	2400	3600	4800	7200	9600	14.4k	19.2k	28.8k	38.4k	57.6k	115.2k

0x000C UART3_TF

Type: Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART3_TF register is the UART3 transmit FIFO register, which is used to access the channel transmit FIFO. The character written is not sent until the transmitter loads it into the transmit shift register and sends it out. If the FIFO is full at the time of writing, the character is lost. This register is updated asynchronously.

0x0010 UART3_CR

Type: Write
Clock: UART_CLK

The UART3_CR register is the UART3 command register, which is used to issue specific commands to the UART3 subsystem. This register is updated asynchronously.

CAUTION Do not reset the transmitter and disable it at the same time. Do not reset the receiver and disable it at the same time.

Bits	Name	Description
8:4	CHANNEL_COMMAND	Writing the appropriate value to these bits executes the commands that are listed in Table 23-4.
3	UART3_TX_DISABLE	This bit terminates the operation of the transmitter after any character in the transmit shift register is sent.
2	UART3_TX_EN	This bit enables the operation of the transmitter.
1	UART3_RX_DISABLE	This bit immediately terminates the operation of the channel receiver and any incoming characters are lost. None of the receiver status bits are affected by this command and characters that are already in the receive FIFO remain there.
0	UART3_RX_EN	This bit enables the channel receiver.

Table 23-6 CHANNEL_COMMAND bit value descriptions

Value	Description	Result
00	Null command	Does nothing.
01	Reset receiver	Resets the receiver as if a hardware reset were issued. The receiver is disabled and the FIFO is flushed.
02	Reset transmitter	Resets the transmitter as if a hardware reset were issued. The transmitter signal goes high (marking) and the FIFO is flushed.
03	Reset error status	Clears the overrun error and hunt char received status bits in both the character and block error modes. In the block error mode, it clears the error status and received break.
04	Reset break change interrupt	Clears the break change interrupt status bit.
05	Start break	Forces the transmitter signal low. The transmitter must be enabled. If the transmitter is busy, the break is started when all characters in the transmit FIFO and the transmit shift register have been completely sent.
06	Stop break	If executed while channel is breaking, this command causes the transmitter signal to go high. The signal remains high for at least one bit time before sending out a new character.
07	Reset CTS_N	Clears ISR bit 5.
08	Reset TX_ERROR	Clears TX_ERROR

Table 23-6 CHANNEL_COMMAND bit value descriptions (Continued)

Value	Description	Result
09	Packet mode	Turns on the sample data mode, which causes the receiver to sample the receive data stream at 16 times the programmed baud rate. The data is sampled with the start of the start bit, or the first data bit, and continued until the marking state. To exit this state, write 1100 in the command field.
0C	Mode reset	Turns off the sample data mode.
0D	Set RFR_N	Asserts the ready for receiving signal (active low).
0E	Reset RFR_ND	Deasserts the ready for receiving signal.
10	Clear TX_DONE	Clears the TX_DONE interrupt (ISR bit 8)

0x0014 UART3_IMR

Type: Write
Clock: UART_CLK

The UART3_IMR register is the UART3 interrupt mask register. This register is used to enable the corresponding functions in the UART3_ISR register.

- Setting (1) bit 8:7 and bit 5:0 in this register causes an interrupt to be generated, if the corresponding bit in the UART3_ISR register is set.
- Clearing (0) bit 8:7 and bit 5:0 in this register causes the setting of the corresponding bit in the UART3_ISR register to have no effect on the interrupt signal.

Bit 6 of this register, CURRENT_CTS, is slightly different. If the current value of the CTS is one (1), no interrupt is generated. However, the user can use bit 6 in this register to mask the CTS value when reading the UART3_MISR register or as a general-purpose bit.

UART3_IMR

Bits	Name	Description
8	TX_DONE	This bit is used only in UIM/SIM_IF mode. Set (1) this bit when the last byte in the transmitter has been sent.
7	TX_ERROR	Only used in UIM_IF mode. Set (1) this bit to indicate that there has been parity error during transmission. To clear the detection logic associated with this function, write CR[7:4]=0x8.
6	CURRENT_CTS	This bit indicates the current state of the CTS input. It never generates an interrupt.
5	DELTA_CTS	Set (1),this bit to indicate that the CTS input has changed states. To clear the detection logic associated with this function, write CR[7:4]=0x7.
4	RXLEV	■ Set (1) this bit when a character is loaded into the receive FIFO that brings the total number of characters in the FIFO up to the programmed watermark level in the FIFO watermark register (RFWR). ■ Clear (0) this bit after enough characters have been read to bring the level equal to or below the programmed watermark level.

UART3_IMR (Continued)

Bits	Name	Description
3	RXSTALE	This bit indicates that there are one or more characters in the Rx FIFO (but less than the level defined in RFWR) and they have been there longer than the time-out value that is specified by the UART3_IPR register. After this bit is set (1), an appropriate time-out value is then written to the STALE_TIMEOUT_LSB bits of the UART3_IPR register. The STALE_TIME_OUT value specifies how many character times must elapse before a stale character time-out is generated. In this instance, a character time is defined as 10 times the bit rate. The RXSTALE interrupt status bit clears each time a character is read from the Rx FIFO. The stale character interrupt duration is measured from either the first or the last arriving character in the Rx FIFO as programmed by the RXSTALE_LAST bit (bit 5) in the UART3_IPR register. When the RXSTALE_LAST bit is clear (0), the value that is programmed into the STALE_TIMEOUT bits is usually greater than the value programmed in RFWR. If the value is not greater, then STALE_TIMEOUT expires before the RFWR setpoint.
2	RXBREAK	This bit is set (1) if the start or end of a break condition is detected. The logic associated with this condition is cleared (0) by writing CR[7:4]=0x4. A detected break is defined as an all-zeros character with an invalid stop bit AND LOW PARITY. The end of a break is defined as a mark condition (1) at least 1/2 bit width. A detected break occupies one position in the Rx FIFO.
1	RXHUNT	This bit is set (1) when an incoming character matches the value programmed into the UART3_HCR register. This condition is cleared (0) by issuing a reset error status command (CR[7:4]=0x3).
0	TXLEV	■ Set (1) this bit when a character which is transferred from the transmit FIFO to the transmit shift register brings the total number of characters in the transmit FIFO equal to or below the programmed watermark level in the UART3_TFWR register. ■ Clear (0) this bit after enough characters have been written to the transmit FIFO to bring the level above the programmed watermark level.

0x0018 UART3_IPR

Type: Read/Write
Clock: UART_CLK

The UART3_IPR register is the UART3 interrupt programming register.

Bits	Name	Description
7	STALE_TIMEOUT_MSB	This bit is the MSbit of the STALE_TIMEOUT bitfield. The stale character time-out duration field contains a number from 1 to 63. This number determines how many character times must elapse before a stale character time-out is generated. This character time is defined as 10 times the bit rate. Do not clear (0) this register if the stale character time-out interrupt is enabled. Normally, the stale character time-out is greater than the value in RFWR. When RXSTALE_LAST is cleared, the value programmed into STALE_TIMEOUT is usually greater than the value programmed in RFWR. Otherwise the STALE_TIMEOUT always expires prior to the RFWR setpoint. The STALE_TIMEOUT field is 6 bits long. Note the discontinuity in the bit assignments.
6	SAMPLE_DATA	Setting (1) this bit enables the new sample data mode, which means that the start bit is sampled as well as the rest, when in sample data mode. See the UART3_CR register, CHANNEL_COMMAND bit for more information.
5	RXSTALE_LAST	This bit controls the method for calculating the RX_STALE character time-out interrupt. ■ Set (1) this bit to measure the RX stale character time-out duration from the end of the last character arriving in the RX FIFO. ■ Clear (0) this bit to measure the RX stale character time-out duration from the end of the first character arriving in the RX FIFO. This bit is valid for all devices which return a value greater than or equal to 0x11.
4:0	STALE_TIMEOUT_LSB	This bitfield is the LSbits of the STALE_TIMEOUT bitfield.

0x001C UART3_TFWR

Type: Read/Write
Clock: UART_CLK

The UART3_TFWR register is the UART3 transmit FIFO watermark register.

Bits	Name	Description
5:0	TFW	These bits contain a number, between 0 and 63, that determines the level of the transmit FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of TX characters in the FIFO) is less than or equal to the value in TFWR. For example: If this register is programmed with the value 5, a TXLEV interrupt is issued when five characters or less are left in the TX FIFO.

0x0020 UART3_RFWR

Type: Read/Write
Clock: UART_CLK

The UART3_RFWR register is the UART3 receive FIFO watermark register.

Bits	Name	Description
5:0	RFW	These bits contain a number, between 0 and 63, that determines the level of the receive FIFO at which an interrupt is generated. The interrupt is generated when the FIFO level (number of characters in the RX FIFO) is greater than the value in RFWR. For example: If this register is programmed with the value 5 then an RXLEV interrupt is issued when six characters are received.

0x0024 UART3_HCR

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART3_HCR register is the UART3 hunt character register. This register contains the character for which the receiver looks to assert a hunt character received interrupt.

0x0028 UART3_MREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART3_MREG register is the M value register, which contains the 8 most significant bits of the M value for the system clock that generates the M/N counter. The optimal M value for the present clock scheme is 12.The recommended programming value is 0x6 when the clk source is at TCXO =19.2MHz. The recommended programming value is 0x18 when the clock source is at TCXO/4 = 4.8MHz. See UART2_CSR on page 23-17.

0x002C UART3_NREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART3_NREG register is the N value register. This register contains the 8 most significant bits of the 1's complement of N–M for the system clock that generates the M over N counter. The optimal N value for the present clock scheme is 125. The recommended programming value is 0xF1 when the clk source is at tcxo=19.2MHz. The recommended programming value is 0xF6 when the clk source is at TCXO/4 = 4.8MHz.

0x0030 UART3_DREG

Type: Read/Write
Clock: UART_CLK

Bits	Name	Description
7:0	DATA	The UART3_DREG register is the D value register. This register contains the 8 most significant bits of the D value for the system clock that generates the M/N counter. The optimal D value for the present clock scheme is 513. The recommended programming value is 0x80, when the TCXO is 4.92 MHz.

0x0034 UART3_MNDREG

Type: Read/Write
Clock: UART_CLK

The UART3_MNDREG register is the M, N, and D extra value register. This register contains the least significant bits for the M, N, and D registers of the M/N counter. The recommended programming value is 0x1A when the clk source is at TCXO = 19.2MHz. The recommended programming value is 0xA when the clk source is at TCXO/4 = 4.8MHz.

Bits	Name	Description
5	MREG_LSB	See the description for the UART3_MREG register.
4:2	NREG_LSB	See the description for the UART3_NREG register.
1:0	DREG_LSB	See the description for the UART3_DREG register.

0x0038 UART3_IRDA

Type: Write
Clock: UART_CLK

The UART3_IRDA register enables the IRDA function, selects a loopback, and allows inversion of RX and TX data. This register also The UART3_IRDA register controls the IRDA transceiver that optionally interfaces between the UART3 and the DP_RX_DATA and DP_TX_DATA pins.

The UART3 proper is a standard RX/TX configuration that converts the serial pin input and output lines to 8-bit data for the microprocessor. The configuration contains receive and transmit blocks, a control block, a bit rate generator for RX and TX, and FIFO interfaces with the microprocessor data bus. The IRDA register provides the means of switching IRDA circuits into the serial I/O lines and the IRDA function is normally switched out. The IRDA modulator/demodulator may be enabled or disabled under microprocessor control. When enabled, the RX and TX data paths have individual inversion select bits, so that external true or inverted drivers may be used in the signal path.

UART3_IRDA

Bits	Name	Description
3	IRDA_LOOPBACK	■ This bit loops the IRDA modulated TX line back into the IRDA RX line. The normal UART3 loopback mode must be off in order to see this effect.
2	INVERT_IRDA_TX	This bit inverts the polarity of the IRDA modulated signal on the DP_TX_DATA pin. ■ Set (1) this bit for an inverted polarity. ■ Clear (0) this bit for a non-inverted polarity.

UART3_IRDA (Continued)

1	INVERT_IRDA_RX	This bit inverts the polarity of the IRDA received signal on the DP_RX_DATA pin. ■ Set (1) this bit for inverted the polarity. ■ Clear (0) this bit for non-inverted polarity.
0	IRDA_EN	■ Set (1) this bit to enable the IRDA transceiver. ■ Clear (0) this bit to bypass the IRDA transceiver and ignore the other bits of this register.

0x003C UART3_SIM_CFG

Type: Write
Clock: UART_CLK

The UART3_SIM_CFG register is used to configure the SIM interface for the UART3.

UART3_SIM_CFG

Bits	Name	Description
17	UIM_TX_MODE	When this bit is set (1), the transmission portion of the SIM interface operates in block mode (T=1). When clear (0), the transmission portion of the SIM interface operates in asynchronous transfer mode (T=0). Note that the operation of this function is conditional upon UART2_SIM_CFG[0] being set (1) to designate the UIM_IF mode of operation.
16	UIM_RX_MODE	When this bit is set (1), the receive portion of the SIM interface operates in block mode (T=1). When clear (0), the receive portion of the SIM interface operates in asynchronous transfer mode (T=0). Note that the operation of this function is conditional upon UART2_SIM_CFG[0] being set (1) to designate the UIM_IF mode of operation.
15:8	SIM_STOP_BIT_LEN	In UIM_IF mode, this field programs the duration of the stop bit that is appended to each transmitted character. The stop bit duration can be 1 to 254 bit times. The receiver only samples the stop bit once, regardless of the programmed stop bit length. 00000001 : 1 bit times 00000010 : 2 bit times ..... 11 111110: 254 bit times
7	SIM_CLK_ON	When this bit is set(1), the UIM_CLK is turned on at either the TD8 or TD4 frequency, as indicated by bit6(UIM_CLK_TD8_SEL). When this bit is cleared, the UIM_CLK is stopped either LOW or HIGH, as indicated by bit 5(UIM_CLK_STOP_HIGH).
6	SIM_CLK_TD8_SEL	This bit selects the UIM_CLK frequency at which the UIM_CLK is turned on. 1 : TD8 (the UIM_CLK runs at the TD8 frequency) 0 : TD4 (the UIM_CLK runs at the TD4 frequency)

UART3_SIM_CFG (Continued)

Bits	Name	Description
5	SIM_CLK_STOP_HIGH	This bit indicates at what point - high or low - the UIM_CLK will stop. 1 : high 0 : low
4	SIM_CLK_SEL	Unused.
3	MASK_RX	Setting (1) this bit masks off any 0 on the RX line, which prevents the receiver from seeing the START bit. Set (1) this bit if you do not want to receive the data you transmit.
2	SWAP_D	Setting (1) this bit causes the UIM_IF to swap the 8 data bits (making MSB into LSB, bit 6 into bit1, and so on) before it is read by the microprocessor.
1	INV_D	Set (1) this bit to cause the UIM_IF to invert the 8 data bit before it is read by the microprocessor.
0	SIM_SEL	Set (1) this bit to designate the UIM_IF mode of operation.

23.2.3.2 Read registers

NOTE The addresses of the read-only registers are mapped into the same addresses of the 4 write-only registers in the section above this one. They are: UART3_CSR, UART3_TF, UART3_CR, and UART3_IMR, respectively.

0x0008 UART3_SR

Type: Read
Clock: UART_CLK

The UART3_SR register is the UART3 status register. This register is used to obtain the current state of the UART3 subsystem. This register is updated asynchronously.

UART3_SR

Bits	Name	Description
7	HUNT_CHAR	When set (1), this bit indicates that the programmed character in the UART3_HCR register was received. If programmed, this condition causes the RXHUNT bit to be set (1) in the UART3_ISR register. This bit is cleared by issuing a reset error status command.
6	RX_BREAK	When set (1), this bit indicates that an all-zero character of the programmed length was received without a stop bit. If a break begins in the middle of a character, it is detected if the condition remains until the end of the next character time. Only one all zero character is written into the FIFO upon receiving the break. After the RX input returns high for at least half a bit time, then more characters are accepted by the receiver. Whenever the break condition starts or stops as defined above, the RXBREAK bit is set (1) in the UART3_ISR register. This bit is appended to the character in the receive FIFO.
5	PAR_FRAME_ERR	A parity error occurs if parity is programmed to be checked and an incoming character is received with incorrect parity. A framing error occurs when the RX input is low while the stop bit is being sampled. This bit is sampled in the middle of the first stop bit position, regardless of the programmed length of the stop bit. This error bit is appended to the character in the receive FIFO.
4	UART3_OVERRUN	An overrun error occurs if one or more incoming characters are lost. This happens when the receive FIFO is full, a full character has been received and is waiting to be written to the FIFO, and a new start bit is detected. The character in the receive shift register is lost. Upon overrun, this status bit remains high until cleared by a reset error status command.
3	TXEMT	This bit is set when the transmitter under-runs. It is set after the transmission of the stop bit at the end of a character, if there are no characters waiting to be sent. It is reset when a character is written to the transmit FIFO.
2	TXRDY	This bit indicates that the transmit FIFO has space in it. It is reset when the FIFO becomes full and set when space becomes available again. Any characters that are written to the FIFO while this bit is low is lost.
1	RXFULL	This bit is set when the receive FIFO becomes full. It is reset when the CPU reads a character, if there is not a character waiting in the receive shift register that takes the read character's place.
0	RXRDY	This bit is set when there is a character in the receive FIFO waiting to be read. It is reset when the last character currently stored in the FIFO is read.

0x000C UART3_RF

Type: Read
Clock: UART_CLK

Bits	Name	Description
9	RXBREAK	This bit is set (1) when a break condition is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 6, for more information on break conditions.
8	PAR_FRAME_ERR	This bit is set (1) when a parity or frame error is detected in bits 7:0 of this register. Refer to the UART_SR register, bit 5, for more information on parity and frame errors.
7:0	DATA	The UART3_RF register is the receive FIFO register. This register returns the next value in the receive FIFO. After the read is completed, the FIFO read pointer is updated to make the next character available. This register is updated asynchronously.

0x0010 UART3_MISR

Type: Read
Clock: UART_CLK

.

Bits	Name	Description
8:0	UART3_MISR	The UART_MISR register is the masked interrupt status register. A read from this register returns the “AND” of the UART_ISR and the UART_IMR registers. This register is updated asynchronously.This bit field is misr (8:0) <= imr(8:0) AND isr (8:0).

0x0014 UART3_ISR

Type: Read
Clock: UART_CLK

The UART3_ISR register is the UART3 interrupt status register. This register provides the current status of the possible interrupt conditions.

- If one of these bits is set (1), and the corresponding bit in the UART3_IMR register is set (1), an interrupt is generated (with the exception of bit 6, CURRENT_CTS – see the description of the UART3_IMR register).
- If the corresponding bit in the UART3_IMR register is clear (0), setting one of these bits has no effect on the UART3 interrupt request signal. This register is updated asynchronously.

Bits	Name	Description
8	TX_DONE	See UART2_IMR on page 23-19 for descriptions of the UART3_ISR bits.
7	TX_ERROR	
6	CURRENT_CTS	
5	DELTA_CTS	
4	RXLEV	
3	RXSTALE	
2	RXBREAK	
1	RXHUNT	
0	TXLEV	

24 RLP DMA

24.1 ARM registers

This chapter covers the RLP DMA registers. These registers are accessible only by the ARM.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: RLP_DMA
- RLP_DMA memory space: 0x80000F00 - 0x80000F7C
- Offset = RLPDMA_BASE+0x0
- RLPDMA_BASE = 0x80000F00
- chipaddr range = 6:2

24.1.1 RLP DMA configuration registers

The configuration registers should be written only when the command queue is empty.

0x0004 USB_RLPDMA_CFG

Type: Read/Write
Clock: HCLK

Bits	Name	Description
3	RLPDMA_RESET	RLP DMA reset bit.Reset the RLP DMA block only. This bit is not self-clearing.
2	BURST_TYPE	Burst type bit (burst mode used only when reading command items or reading data. Data writes are always done in NONSEQ mode) When clear (0), HBURST is either incremented by 4 or single. Any transfers less than 4 beats are treated as a SINGLE transfer. When set (1), HBURST is tied up to INCR. Burst length is undefined.
1	ENABLE	Enable bit (only works when CQ_DISABLE bit is set to 1) 0 : data transfer disabled 1 : data transfer enabled (or started) Enable bit is a pulse signal and is not readable.
0	CQ_DISABLE	Command Queue disable. When disabled, software programs the command items directly. When enabled, RLP DMA hardware loads the command items from memory. 0 : command queue enabled (normal operation) 1 : command queue disabled

0x0010 USB_RLPDMA_QUEUE_BASE_ADDR

Type: Read/Write
Clock: HCLK

Bits	Name	Description
31:2	BASE[29:0]	Base address in 32-bit words
1:0	RESERVED_BITS1_0[1:0]	Reserved.

0x0014 USB_RLPDMA_QUEUE_CFG

Type: Read/Write
Clock: HCLK

Bits	Name	Description
25:16	CQ_WATER_MARK[9:0]	Command Queue Water Mark Range from 0 to 1023 (Recommended values are >= 2)
15:10	RESERVED_BITS15_10[5:0]	Reserved
9:0	CQ_SIZE[9:0]	Command Queue Size, in terms of command items. Range from 0 to 1023. Sets the maximum value of GRP_WR_PT and GRP_RD_PT.

0x0018 USB_RLPDMA_QUEUE_WR_PT

Type: Read/Write
Clock: HCLK

Bits	Name	Description
9:0	WPTR[9:0]	Write pointer, in terms of command items. Programmed to value less than CQ_SIZE.

0x001C USB_RLPDMA_QUEUE_RD_PT

Type: Read
Clock: HCLK

Bits	Name	Description
9:0	RPTR[9:0]	Read pointer, in terms of command items.

24.1.2 RLP DMA command item registers

The registers below form a Command Item (CI). They are read only in normal operation, where the RLP DMA block fetches the command item registers from memory (fetching address determined by USB_RLPDMA_QUEUE_BASE_ADDR register).

0x0020 USB_RLPDMA_CI_CTL

Type: Read
Clock: HCLK

Bits	Name	Description
25:16	LENGTH[9:0]	Command Item transfer length in bytes
15:10	RESERVED_BITS15_10[5:0]	Reserved
9	INT_EN	Termination Interrupt 0: Disable interrupt (at completion of command item) 1: Enable interrupt (at completion of command item)
8	DNI	Destination Address Non-increment 0 : increment (destination address increment) 1 : non-increment (destination address non-increment)
7:1	RESERVED_BITS7_1[6:0]	Reserved
0	ROP_COPY_EN	Raster Operation. Setting this bit starts the copy operation 0: No Copy 1: Copy (from Source to Destination)

0x0024 USB_RLPDMA_CI_SRC_ADDR

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	SRC_ADDR[31:0]	Byte-addressable starting address of the source region

0x0028 USB_RLPDMA_CI_DST_ADDR

Type: Read
Clock: HCLK

Bits	Name	Description
31:0	DST_ADDR[31:0]	Byte-addressable starting address of the destination region

24.1.3 USB control registers

The following registers control USB logic that lies outside of RLPDMA block and OTG core.

0x0040 USB_FIFO_CFG

Type: Read/Write
Clock: HCLK

Bits	Name	Description
0	RLPFIFO_RESET	Reset the rlpfifo block. The bit is not self-clearing.

0x0048 USB_HNP_ASSIST_CTL

Type: Read/Write
Clock: HCLK

Bits	Name	Description
0	HNP_HW_ASSIST_EN	Enable HNP mode for transferring to host mode. Valid only in usb DAT_SE0 mode.

0x004C USB_FIFO_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
29:24	ISO_DSP2USB_STATUS[5:0]	Provides the current status of isochronous transfer from DSP to USB Bits 5:4 provide the current DSP pointer location in buffer. Bits 3:0 provide the valid status of bytes in buffer.
23:22	RESERVED_BITS23_22[1:0]	Reserved
21:16	ISO_USB2DSP_STATUS[5:0]	Provides the current status of isochronous transfer from USB to DSP. Bits 5:4 provide the current DSP pointer location in buffer. Bits 3:0 provide the valid status of bytes in buffer.
15:14	RESERVED_BITS15_14[1:0]	Reserved
13:0	RLP_STATUS[13:0]	Provides the current status of transfer from RLP DMA to USB Bits 13:11 provide the current OTG core pointer location in buffer. Bits 10:8 provide the current DSP ointer location in buffer. Bits 7:0 provide the valid status of bytes in buffer.

0x0060 USB_INT_MASK

Type: Read/Write
Clock: HCLK

Bits	Name	Description
9	OE_TP_N_MASK	USB xcvr signal usb_oe_tp_n enable
8	RLPDMA_CIT_MASK	RLPDMA Command Item Termination interrupt enable
7	RLPDMA_CQA_MASK	RLPDMA Command Queue Available interrupt enable
6	RLPDMA_MBERROR_MASK	RLPDMA AHB Master bus error interrupt enable
5	HOST_WAKEUP_MASK	Host wakeup enable
4	FUNC_WAKEUP_MASK	Function wakeup enable
3	OTG_WAKEUP_MASK	OTG wakeup enable
2:0	RESERVED_BITS2_0[2:0]	Reserved

0x0064 USB_INT_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
9	OE_TP_N_STATUS	Status of USB xcvr signal usb_oe_tp_n before mask
8	RLPDMA_CIT_STATUS	Status of Command Item Termination interrupt before mask: set when the current command item has been finished
7	RLPDMA_CQA_STATUS	Status of Command Queue Available interrupt before mask: set when the number of emptry entries in the command queue reaches the value indicated by CQ_QUEUE_WATERMARK
6	RLPDMA_MBERROR_STATUS	Status of AHB Master bus error interrupt before mask: set when the AHB master encounters a bus error response from a slave
5	HOST_WAKEUP_STATUS	Status of Host wakeup before mask
4	FUNC_WAKEUP_STATUS	Status of Function wakeup before mask
3	OTG_WAKEUP_STATUS	Status of OTG wakeup before mask
2	CORE_DMA_INTR_STATUS	Status of TDI Core DMA interrupt before mask
1	HOST_INTR_STATUS	Status of Host interrupt before mask
0	FUNC_INTR_STATUS	Status of Function interrupt before mask

0x0068 USB_INT_CLEAR

Type: Write
Clock: HCLK

Bits	Name	Description
9	RESERVED_BITS9	Reserved (interrupt bit 9 cleared in usb xcvr)
8	RLPDMA_CIT_CLEAR	Clear RLPDMA Command Item Termination interrupt
7	RLPDMA_CQA_CLEAR	Clear RLPDMA Command Queue Available interrupt
6	RLPDMA_MBERROR_CLEAR	Clear RLPDMA AHB Master bus error interrupt
2:0	RESERVED_BITS5_0[2:0]	Reserved (interrupt bits 5:0 cleared in OTG core)

0x0070 Reserved.

25 Secure Digital Card Controller(SDCC)

25.1 ARM registers

Addresses are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: SDCC
- Number of word addresses: 64 (not all used)
- Byte address range: 0x0E00 - 0x0EFC
- OFFSET = CHIP_BASE+0x00E00
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x000 MCI_POWER

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00

(ARM name: MCIPower)

Bits	Name	Description
6	OPEN_DRAIN	MCICMD output control
1:0	CONTROL	This register is used to control the operation of the memory controller. 00 : power-off 01 : reserved_encoding 10 : powerup 11 : poweron

0x004 MCI_CLK

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x0000

(ARM name:MCIClock)

Bits	Name	Description
14	INVERT_IN	Clear (0) to latch data and command coming in on the falling edge. Set (1) to latch data and command coming in on the rising edge.
13	INVERT_OUT	Clear (0) to change data and command going out on the falling edge. Set (1)to change data and command going out on the rising edge.
12	FLOW_ENA	Enable flow control: 0 : disable (default) 1 : enable
11	WIDEBUS	Enable widebus mode: 0 : 1 bit mode 1 : 4 bit mode
9	PWRSAVE	Disable PrimeCell MCI clock ouput when bus is idle. 0 : always enabled 1 : clock enabled (when bus is active)
8	ENABLE	Enable PrimeCell MCI bus clock: 0 : clock disabled 1 : clock enabled

0x008 MCI_ARGUMENT

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00000000

(ARM name: MCIArgument)

Bits	Name	Description
31:0	CMD_ARG	Command argument.

0x00C MCI_CMD

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x000

(ARM name: MCICommand)

Bits	Name	Description
11	PROG_ENA	If set (1), PROG_DONE status bit will be asserted when busy is de-asserted. This bit is to be used with a stop or status command after a block write is performed. Default is 0.
10	ENABLE	If set (1), CPSM is enabled.
9	PENDING	If set (1), CPSM waits for CmdPend before it starts sending a command.
8	INTERRUPT	If set (1), CPSM disables command timer and waits for interrupt request.
7	LONGRSP	If set (1), receives a 136-bit long response.
6	RESPONSE	If set (1), CPSM waits for a response.
5:0	CMD_INDEX	Command index.

0x010 MCI_RESP_CMD

Type: Read
Clock: same rate as HCLK
Reset State: 0x00

(ARM name: MCIRespCmd)

Bits	Name	Description
5:0	RESPCMD	Response command index.

0x014+4n MCI_RESPn, n=[0..3]

Type: Read
Clock: same rate as HCLK
Reset State: 0x00000000

(ARM name: MCIResponse0-3). Note that MCIResponse3 has 31 bits; the other three (*RESP0, *RESP1, *RESP2) have 32 bits.

Bits	Name	Description
31:0	STATUS	Card status.

The card status size can be 32 or 127 bits, depending on the response type (see Table 25-1). The most significant bit of the card status is received first. The MCIResponse3 register LSBit is always 0.

Table 25-1 Response register type

Description	Short response	Long response
MCIResponse0	Card Status [31:0]	Card status [127:96]
MCIResponse1	Unused	Card status [95:64]
MCIResponse2	Unused	Card status [63:32]
MCIResponse3	Unused	Card status [31:1]

0x024 MCI_DATA_TIMER

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00000000

(ARM name : MCIDataTimer)

Bits	Name	Description
31:0	DATA_TIME	Data timeout period.

0x028 MCI_DATA_LENGTH

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x000

(ARM name : MCIDataLength)

Bits	Name	Description
15:0	DATALENGTH	Data length value.

0x02C MCI_DATA_CTL

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00

(ARM name : MCIDataCtrl)

MCI_DATA_CTL

Bits	Name	Description
7:4	BLOCKSIZE	Data block length. Block length = 2^n (For example, BLOCKSIZE = 9 -> 2^9 = 512 bytes
3	DMA_ENABLE	Enable DMA: 0 : DMA disabled 1 : DMA enabled
2	MODE	Data transfer mode: 0 : block data transfer 1 : stream data transfer (not supported for DMA transfers)

MCI_DATA_CTL (Continued)

Bits	Name	Description
1	DIRECTION	Data tranfer direction: 0 : controller to card 1: card to controller
0	ENABLE	Data transfer enabled.

0x030 MCI_DATA_COUNT

Type: Read
Clock: same rate as HCLK
Reset State: 0x0000

(ARM name: MCIDataCnt)

NOTE This register should be read only when the data transfer is complete.

Bits	Name	Description
15:0	DATACOUNT	Remaining data.

0x034 MCI_STATUS

Type: Read
Clock: same rate as HCLK
Reset State: 0x000000

(ARM name: MCIStatus)

Static[24:22] and [10:0] These remain asserted until they are cleared by writing to the Clear register (see Clear register, MCI_CLEAR)

Dynamic[12:11] These change state depending on the state of the underlying logic (for example, FIFO full and empty flags are asserted and deasserted as data while written to the FIFO).

MCI_STATUS

Bits	Name	Description
24	DMA_DONE	DMA transfer complete.
23	PROG_DONE	Programming done.
22	SDIO_INTR	SDIO interrupt indicator.
21	RXDATA_AVLBL	Data available in receive FIFO.
20	TXDATA_AVLBL	Data available in transmit FIFO.

MCI_STATUS (Continued)

Bits	Name	Description
19	RXFIFO_EMPTY	Receive FIFO empty.
18	TXFIFO_EMPTY	Transmit FIFO empty.
17	RXFIFO_FULL	Receive FIFO full.
16	TXFIFO_FULL	Transmit FIFO full.
15	RXFIFO_HALF_FULL	Receive FIFO half full.
14	TXFIFO_HALF_FULL	Transmit FIFO half full.
13	RXACTIVE	Data receive in progress.
12	TXACTIVE	Data transmit in progress.
11	CMD_ACTIVE	Command transfer in progress.
10	DATA_BLK_END	Data block sent / received (CRC check passed).
9	START_BIT_ERR	Start bit not detected on all data signals in wide bus mode.
8	DATAEND	Data end (data counter is zero).
7	CMD_SENT	Command sent (no response required).
6	CMD_RESPONSE_END	Command response received (CRC check passed).
5	RX_OVERRUN	Receive FIFO overrun error.
4	TX_UNDERRUN	Transmit FIFO underrun error.
3	DATA_TIMEOUT	Data timeout.
2	CMD_TIMEOUT	Command response timeout.
1	DATA_CRC_FAIL	Data block sent / received (CRC check failed)
0	CMD_CRC_FAIL	Command response received (CRC check failed)

0x038 MCI_CLEAR

Type: Write
Clock: same rate as HCLK
Reset State: —

(ARM name: MCIClear)

Bits	Name	Description
24	DMA_DONE_CLR	Clears DmaDone flag.
23	PROG_DONE_CLR	Clears ProgDone flag.
22	SDIO_INTR_CLR	Clears SDIOInt flag.
10	DATA_BLK_END_CLR	Clears DataBlockEnd flag.
9	START_BIT_ERR_CLR	Clears StartBitErr flag.
8	DATA_END_CLR	Clears DataEnd flag.
7	CMD_SENT_CLR	Clears commandSent flag.
6	CMD_RESP_END_CLT	Clears CmdRespEnd flag.
5	RX_OVERRUN_CLR	Clears RxOverrunClr flag.
4	TX_UNDERRUN_CLR	Clears TxUnderrun flag.
3	DATA_TIMEOUT_CLR	Clears DataTimeOut flag.
2	CMD_TIMOUT_CLR	Clears CmdTimOutflag.
1	DATA_CRC_FAIL_CLR	Clears DataCrcFail flag.
0	CMD_CRC_FAIL_CLR	Clears CmdCrcFail flag.

0x03C+4n MCI_INT_MASKn, n=[0,1]

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x000000

(ARM name: MCIMask0 , MCIMask1)

MCI_INT_MASKn

Bits	Name	Description
24	MASK24	Mask DmaDone flag.
23	MASK23	MASK ProgDone flag.
22	MASK22	MASK SDIOInt flag.
21	MASK21	MASK RxDataAvlbl flag.

MCI_INT_MASKn (Continued)

Bits	Name	Description
20	MASK20	MASK TxDataAvlbl flag.
19	MASK19	MASK RxFifoEmpty flag.
18	MASK18	MASK TxFifoEmpty flag.
17	MASK17	MASK RxFifoFull flag.
16	MASK16	MASK TxFifoFull flag.
15	MASK15	MASK RxFifoHalfFull flag.
14	MASK14	MASK TxFifoHalfFull flag.
13	MASK13	MASK RxActive flag.
12	MASK12	MASK TxActive flag.
11	MASK11	MASK CmdActive flag.
10	MASK10	MASK DataBlockEnd flag.
9	MASK9	MASK StartBitErr flag.
8	MASK8	MASK DataEnd flag.
7	MASK7	MASK CmdSent flag.
6	MASK6	MASK CmdRespEnd flag.
5	MASK5	MASK RxOverrun flag.
4	MASK4	MASK TxOverrun flag.
3	MASK3	MASK DataTimeOut flag.
2	MASK2	MASK CmdTimeOut flag.
1	MASK1	MASK DataCmdCrcFail flag.
0	MASK0	MASK CmdCrcFail flag.

0x044 MCI_FIFO_COUNT

Type: Read
Clock: same rate as HCLK

Bits	Name	Description
14:0	DATA_COUNT	Remaining data.

0x080 MCI_FIFO

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	DATA	FIFO data. This register is aliased to 16 words, 0x080 - 0xBC.

0x0C0 MCI_ABORT

Type: Write
Clock: same rate as HCLK

Bits	Name	Description
0	MCIABORT	Signals the next command will be an abort (stop) command.

0x0C4 MCI_START_ADDR

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	START_ADDR	Start address for DMA transfer.

0x0C8 MCI_DMA_CONFIG

Type: Read/Write
Clock: same rate as HCLK
Reset State: 0x8863

Bits	Name	Description
15:12	RX_THRESH	FIFO level at which to set dmareqset for Rx direction. 0x8 : default
11:8	TX_THRESH	FIFO level at which to set dmareqset for Tx direction. 0x8 : default
7	TEST_ENA	For testing, hold dmareqset high and dmareqclr low and multiplex the address counter into the data path (Rx). 0 : default
6	ADDR_INC_ENA	Enable address counter to increment. 1 : default
5	HLOCK_ENA	Lock the bus. 1 : default
4	ENDIAN_CHANGE	Enable endian byte order change (B3 B2 B1 B0 => B0 B1 B2 B3) 0 : default
3:0	MAX_BEAT	Maximum number of beats in a burst (MAX_BEAT + 1). For example, MAX_BEAT = 0x3 = 4-beat burst maximum. 0x3 : default

0x0E0 Reserved.

0xE4 Reserved.

0x0E8 Reserved.

0x0EC Reserved.

0x0F0 Reserved.

0x0F4 Reserved.

0x0F8 Reserved.

0x0FC Reserved.

26 Multi-Media Card Controller

26.1 ARM registers

This chapter contains definitions of the MultiMediaCard registers. The MultiMedia Card controller is controlled by a set of registers that the application configures before every operation on the MultiMediaCard bus. The following table lists register addresses and reset states.

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: MMC
- Offset = CHIP_BASE+0x4000
- Max = CHIP_BASE+0x48FC
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

0x0000 **Reserved.**

0x0400 **Reserved.**

0x0800 **MMCC_CLK_CTL**

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

This register starts and stops the MultiMediaCard clock. To start the clock, set (1) bit 1 of this register. To stop the clock set (1) bit 0 of this register. The value ‘11’ is prohibited.

Bits	Name	Description
1	MMC_START_CLK	Starts the MultiMediaCard clock (‘1’ = true).
0	MMC_STOP_CLK	Stops the MultiMediaCard clock, (‘1’ = true).

0x0804 MMCC_STATUS

Type: Read
Clock: HCLK
Reset State: 0x0140

This register provides the current status of the MMC controller.

Bits	Name	Description
13	MMC_END_CMD_RESP	End command response. (‘1’ = true).
12	MMC_PRG_DONE	End write and read operation. (‘1’ = true).
11	MMC_DATA_TRAN_DONE	In a read/write operation, this bit indicates that the data transfer is finished. (‘1’ = true).
10:9	RESERVED_BITS	
8	MMC_CLK_OFF	MultiMediaCard clock stopped. (‘1’ = true).
7	MMC_FULL_FIFO	FIFO buffer full. (‘1’ = true).
6	MMC_EMPTY_FIFO	FIFO buffer empty. (‘1’ = true).
5	MMC_RESP_CRC_ERROR	Response CRC error. (‘1’ = true).
4	MMC_SPI_RD_ERR_TOKEN	SPI data error token has been received. (‘1’ = true).
3	MMC_CRC_RD_ERROR	CRC read error. (‘1’ = true).
2	MMC_CRC_WR_ERROR	CRC write error. (‘1’ = true).
1	MMC_TIMEOUT_RESP	Time out response. (‘1’ = true).
0	MMC_TIMEOUT_READ	Time out read data. (‘1’ = true).

0x0808 MMCC_CLKRATE

Type: Read/Write
Clock: HCLK
Reset State: 0x0006

This register is used to set the speed for the MMC bus clock. This register should be written only after software has checked the status register to see if the MultiMediaCard clock has stopped.

This register should never be set to 0x0.

Bits	Name	Description	
2:0	DATA	Content	Frequency
		0x0	Reserved_Bit
		0x1	1/2 Master clock
		0x2	1/4 Master clock
		0x3	1/8 Master clock
		0x4	1/16 Master clock
		0x5	1/32 Master clock
		0x6	1/64 Master clock

NOTE The maximum MultiMediaCard clock frequency is half the clock speed of the ARM, up to 20 MHz. If the ARM runs at 20 MHz, the maximum frequency of the MultiMedia clock is 10 MHz. If the ARM runs at 40 MHz or higher, the maximum frequency for the MultiMedia clock is 20 MHz.

0x080C Reserved.

0x0810 MMCC_CMDDAT_CTL

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Writing this register activates the controller when the clock is started. This needs to be written each time a CMD is to be executed.

Bits	Name	Description
6	MMC_INITIALIZE	This bit enables 80 clocks for initializing the cards (‘1’ = true).
5	MMC_BUSY_BIT	Specifies whether a busy signal is expected after the current command. (‘1’ = true).
4	MMC_STREAM	Specifies whether the data transfer of the current command is in stream or block mode (‘1’ = stream, ‘0’ = block).
3	MMC_WRITE	Specifies whether the data transfer of the current command is a write or read operation (‘1’ = write, ‘0’ = read).
2	MMC_DATA_EN	Specifies whether this command includes a data transfer (‘1’ = true).
1:0	MMC_RESP_FORMAT	Response formats are described in the following table: Bit Value — MultiMediaCard Description 00 — No response 01 — Format R1 10 — Format R2 11 — Format R3

0x0814 MMCC_RESPONSE_TO

Type: Read/Write
Clock: HCLK
Reset State: 0x40

Bits	Name	Description
6:0	DATA	This register specifies the number of MultiMediaCard clocks after the command, before the host turns on the time out error for the received response. The default value of this register is 0x40.

0x0818 MMCC_READ_TO

Type: Read/Write
Clock: HCLK
Reset State: 0xFFFF

Bits	Name	Description
15:0	DATA	This register specifies the number of clocks after the command, before the host turns on the time out error for the received data. The units are [Master clock/256].

0x081C MMCC_BLKLEN

Type: Read/Write
Clock: HCLK
Bits: 9:0
Reset State: 0x0000

Bits	Name	Description
9:0	DATA	his register specifies the block length.

0x0820 MMCC_NOB

Type: Read/Write
Clock: HCLK
Reset State: 0x0001

Bits	Name	Description
15:0	DATA	In block mode, this register specifies the number of blocks. One block is also a possibility.

0x0824 MMCC_INTMASK

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

This register is used to mask different interrupts.

Bits	Name	Description
3	MMC_BUFRDY_MASK	Buffer ready interrupt mask (“1” = Masked).
2	MMC_ENDCMD_MASK	End command response interrupt mask (“1” = Masked).
1	MMC_PRGDONE_MASK	Program done interrupt mask (“1” = Masked).
0	MMC_DATATRAN_MASK	Data transfer done interrupt mask (“1” = Masked).

0x0828 MMCC_CMD

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
7	RESERVED_BIT	Always ‘0’
6:0	MMCC_CMD	This register specifies the command number. If bit 6 is set to ‘0’, then the MMC is in Secure Disk (SD) mode. If bit 6 is set to ‘1’, then the MMCC is in MMC mode.

0x082C MMCC_ARGH

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
15:0	DATA	This register specifies the MSW part of the argument in the current command.

0x0830 MMCC_ARGL

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
15:0	DATA	This register specifies the LSW part of the argument in the current command.

0x0834 MMCC_RESPONSE

Type: Read
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
15:0	DATA	This FIFO register contains the responses (R1/R2/R3) after every command that is sent by the Host. The size of this FIFO register is 8x16-bit.

0x0838 MMCC_FIFO

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
15:0	DATA	This address accesses the FIFO for reading/writing.

0x083C MMCC_BUF_PART_FULL

Type: Read/Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
0	DATA	This register is used when a partial buffer is written in stream mode write.

0x0840 MMCC_DEBUG

Type: Read
Clock: HCLK, AUX_WR_CLK
Reset State: 0x0000

Bits	Name	Description
4:0	DATA	This register holds the state of one of the MMC controller state machines. See the MMCC_DEBUG_CTL register, below.

0x0840 MMCC_DEBUG_CTL

Type: Write
Clock: HCLK, AUX_WR_CLK
Reset State: 0x0000

This register controls the mux to provide visibility to the controller state machines.

Bits	Name	Description
1:0	MMCC_DEBUG_CTL	The following values indicate that MMCC_DEBUG contains the following states: 00: State of CMD fsm 01: State of DAT fsm 10: State of DAT CRC fsm The 11 encoding is reserved.

0x0844 Reserved.

0x0850 MMCC_MUXSEL

Type: Read/Write
Clock: HCLK, AUX_WR_CLK
Reset State: 0x0000

This register controls the interface mux.

Bits	Name	Description
1:0	MMCC_MUXSEL	Selected interface: 00: ARM interface (default) 01: DMA interface 10: MDSP interface

0x0854 MMCC_INTF_ERROR

Type: Read
Clock: HCLK, AUX_RD_CLK
Reset State: 0x0000

This register indicates which interface attempted to illegally access the controller. This register is updated whenever the MMC error interrupt is generated.

Bits	Name	Description
2	MMC_DSP_ACCESS	Set (1) when the MDSP attempted an illegal access -
1	MMC_DMA_ACCESS	Set (1) when the DMA Interface attempted an illegal access -
0	MMC_ARM_ACCESS	Set (1) when the ARM attempted an illegal access.

0x0858 MMCC_WAIT

Type: Read/Write
Clock: HCLK, AUX_WR_CLK
Reset State: 0x0000

This register enables/disables a waitstate for ARM accesses to registers and memory.

Bits	Name	Description
0	WAIT_ENA	0: No waitstates (default) 1: Wait enabled

27 Graphics

27.1 ADSP registers

Defender registers may be accessed by the ADSP through the XMEMC interface or by the ARM through the AHB slave interface.

Addresses in this section are relative to the base address and represent the lower significant bits of the address.

- DSP address space allocated for Defender: 0x0400 to 0x7fc (byte addressing).
- AHB address space allocated for Defender: 0xA8000000 to 0xAFFFFFFC (byte addressing).
- If Xmemc interface is used, bits 13:2 of the address are decoded; for AHB, bits 15:0 are decoded and also hsel_s.
- XMEMC_BASE = 0x0400
- AHB_BASE = 0xA8000000

0x0000 **Reserved.**

0x0004 **Reserved.**

0x0008 **Reserved.**

0x000C **Reserved.**

0x0010 **Reserved.**

0x0018 HSR_INIT_VALUE

Type: Read/Write
Clock: AHB

Bits	Name	Description
15:0	HSR_INIT_VAL	The value used to initialize the Zram. Only the lower 16 bits are used.

0x001C HSR_DISPLAY_SIZE

Type: Read/Write
Clock: AHB

Bits	Name	Description
17:0	HSR_DISPLAY_VAL	The screen display size (e.g., 320x240): 8:0 : Width of display (e.g., 320) 17:9 : Height of display (e.g., 240)

0x0020 BLD_FB_ADDR

Type: Read/Write
Clock: AHB

Bits	Name	Description
31:0	BLD_FB_ADDR	64-byte-aligned address of the upper left corner of the frame buffer in bytes (lower 6 bits are ignored).

0x0024 BLD_FB_LINEOFFSET

Type: Read/Write
Clock: AHB

Bits	Name	Description
11:0	BLD_FB_LINEOFFSET	Frame buffer format register - line offset. This is currently unused, since the frame buffer is expected to be in contiguous memory space.

0x0028 BLD_FB_BPP

Type: Read/Write
Clock: AHB

Bits	Name	Description
1:0	BLD_FB_BPP	Frame buffer color format, bytes per pixel: 00 = 16 bpp RGB (5-6-5) 01 = 16 bpp RGAB (5-5-1-5) 10 = 24 bpp RGB (8-8-8) 11 = 32 bpp ARGB (8-8-8-8, or 24 bpp plus an unused byte: x-8-8-8)

0x002C BLD_AB_ADDR

Type: Read/Write
Clock: AHB

Bits	Name	Description
31:0	BLD_AB_ADDR	16-byte-aligned address of upper left corner of the alpha buffer in bytes (lower 4 bits are ignored). When the frame buffer format has integrated alpha, this address is not used, since the Blending block expects the alpha values to be stored along with the pixel color in the frame buffer.

0x0030 BLD_AB_LINEOFFSET

Type: Read/Write
Clock: AHB

Bits	Name	Description
9:0	BLD_AB_LINEOFFSET	Alpha buffer format register - line offset. This is currently unused, since the alpha buffer is expected to be in contiguous memory space.

0x0034 BLD_AB

Type: Read/Write
Clock: AHB

Bits	Name	Description
0	BLD_AB	0: Separate alpha buffer disable (reset value) 1: Separate alpha buffer enable This setting is valid when the frame buffer format is in RGB, otherwise no effect in frame buffer format with integrated alpha. When disabled, the alpha blending process always uses 1 (the maximum 8 bit numerical representation) as the destination alpha and the alpha values produced are not stored.

0x0038 BLD_CACHE

Type: Read/Write
Clock: AHB

Bits	Name	Description
0	BLD_CACHE	0: Blending block Cache disable 1: Blending block Cache enable (reset value)

0x003C BLD_COLOR0

Type: Read/Write
Clock: AHB

Bits	Name	Description
31:24	ALPHA	Constant color alpha. The constant color values are used for certain rendering functions. In particular, they are optionally used as an input to the texture shading and alpha blending functions.
23:16	RED	Constant color red
15:8	GREEN	Constant color green
7:0	BLUE	Constant color blue

0x0040 BLD_MASK

Type: Read/Write
Clock: AHB

Bits	Name	Description
3	ALPHA	Alpha enable, a value of 1 means alpha produced is written to the buffer. Reset = 1.
2	RED	Red enable, a value of 1 means red color produced is written out to the frame buffer. Reset = 1.
1	GREEN	Green enable, a value of 1 means green color produced is written out to the frame buffer. Reset = 1.
0	BLUE	Blue enable, a value of 1 means blue color produced is written out to the frame buffer. Reset = 1.

0x0044 BIF_CONFIG

Type: Read/Write
Clock: AHB

Bits	Name	Description
3:1	PRIORITY_SEL	Fixed priority for the AHB Read Arbiter 000: CMD then TEX then BLD 001: CMD then BLD then TEX 010: TEX then CMD then BLD 011: TEX then BLD then CMD 100: BLD then CMD then TEX 101: BLD then TEX then CMD
0	LOCK_ARBITER	1: lock the AHB Read Arbiter on the selected priority 0: AHB Read Arbiter performs “Round Robin”

0x004C AHB_M_CONFIG

Type: Read/Write
Clock: AHB

Bits	Name	Description
31:11		don't care
10:8	AHB_IDLE_CYCLES	Programable value of the number of idle hready cycles following each external AHB bus access {1 through 8}. These idle cycles will be inserted between the AHB bus release and the next AHB bus request. 000: 1 idle cycle 001: 2 idle cycles 010: 2idle cycles 011: 3idle cycles 100: 4idle cycles 101: 5idle cycles 110: 6idle cycles 111: 8idle cycles
7:0	DONT_CARE	don't care

0x0050 MICRO1_N

Type: Read/Write
Clock: AHB

Bits	Name	Description
0	MICRO1_N	Interface enabled: 0 : Micro1 Intf ena 1 : AHB Intf ena

0x0070 GO_BIT

Type: Read/Write
Clock: AHB

Bits	Name	Description
0	GO_BIT	0 : Go bit off 1 : Go bit on

0x0080 INTR_STAT

Type: Read/Write
Clock: AHB

Bits	Name	Description
5	AHB_ERROR_INTR	0: interrupt is off 1: interrupt is on
4	BUFFER_A_EMPTY_INTR	0: interrupt is off 1: interrupt is on
3	BUFFER_B_EMPTY_INTR	0: interrupt is off 1: interrupt is on
2	COMMAND_ABORT_INTR	0: interrupt is off 1: interrupt is on
1	FLUSH_DONE_INTR	0: interrupt is off 1: interrupt is on
0	FINISH_DONE_INTR	0: interrupt is off 1: interrupt is on

0x0084 INTR_MASK

Type: Read/Write
Clock: AHB

Bits	Name	Description
5	AHB_ERROR_INTR_MASK	0: interrupt is masked 1: interrupt is enabled
4	BUFFER_A_EMPTY_INTR_MASK	0: interrupt is masked 1: interrupt is enabled
3	BUFFER_B_EMPTY_INTR_MASK	0: interrupt is masked 1: interrupt is enabled
2	COMMAND_ABORT_INTR_MASK	0: interrupt is masked 1: interrupt is enabled
1	FLUSH_DONE_INTR_MASK	0: interrupt is masked 1: interrupt is enabled
0	FINISH_DONE_INTR_MASK	0: interrupt is masked 1: interrupt is enabled

0x0088 INTR_CLEAR

Type: Read/Write
Clock: AHB

Bits	Name	Description
5	AHB_ERROR_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.
4	BUFFER_A_EMPTY_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.
3	BUFFER_B_EMPTY_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.
2	COMMAND_ABORT_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.
1	FLUSH_DONE_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.
0	FINISH_DONE_INTR_CLEAR	Clears the corrosponding bit in INTR_STATUS registers.

0x008C INTR_ACK

Type: Read/Write
Clock: AHB

Bits	Name	Description
0	INTR_ACK	0 : Do not reset (the interrupt request signal) 1 : Reset (the interrupt request signal, and generate a new request for pending interrupts.)

0x00A0 STATUS_CLEAR

Type: Read/Write
Clock: AHB

Bits	Name	Description
6	BIF_STATUS_CLEAR	Clears the BIF_STATUS internal register.
5	BLD_STATUS_CLEAR	Clears the BLD_STATUS internal register.
4	TEX_STATUS_CLEAR	Clears the TEX_STATUS internal register.
3	HSR_STATUS_CLEAR	Clears the HSR_STATUS internal register.
2	SHD_STATUS_CLEAR	Clears the SHD_STATUS internal register.
1	SET_STATUS_CLEAR	Clears the SET_STATUS internal register.
0	CMD_STATUS_CLEAR	Clears the CMD_STATUS internal register.

0x00A4 STATUS

Type: Read
Clock: AHB

(Register bit descriptions on following page.)

Bits	Name	Description
15:0	STATUS	Status selected by STATUS_SEL register. **Command block** 15:14: pp_buffer_status 13:11: fifo_status 10:0: decode_status **Setup block** 15:5: Unused 4 : Illegal triangle coordinate 3 : Illegal triangle size 2 : Illegal state 1 : Zero area triangle 0 : Negative 1/area **Shading block** 15:3: Unused 2 : Perspective correction divided by zero 1 : Illegal state 0 : Bounding box size violation **HSR block** 15: set (1) if hsr_state was HSRWAIT after last clear (sticky) 14: set (1) if hsr_state was HSRZRAM after last clear (sticky) 13: set (1) if hsr_state was HSRINIT after last clear (sticky) 12: set (1) if hsr_state was HSRREADY after last clear (sticky) 11: set (1) if zram_state was an illegal state after last clear (sticky) 10: set (1) if zram_state was ZRAMUPDT after last clear (sticky) 9: set (1) if zram_state was ZRAMCMP after last clear (sticky) 8: set (1) if zram_state was ZRAMWRITE after last clear (sticky) 7: set (1) if zram_state was ZRAMREAD after last clear (sticky) 6:0: Latest snapshot of the statistics counter. Use test mode cfg register to set what statistic is being measured. **Texture block** 14:8: Texture cache statistics 7: Tpixel FIFO full 6: Tpixel FIFO error 5: Blending control FIFO full 4: Blending control FIFO error 3: Attribute FIFO full 2: Attribute FIFO error 1: Weight FIFO full 0: Weight FIFO error **Blending block** 15:9: Percent cache line hit statistic 8:6: Average pixel per cache line access 5: Any memory subsystem FIFO is full 4: Any datapath FIFO is full 3: Memory controller FIFO error, latched 2: Datapath buffer FIFO error, latched 1: Bld_wr_done counter over 2, latched 0: Idle, 1 = no data in the internal pipe **Bus interface block** 5:: Read fifo full 4:: Read fifo empty 3:: Read fifo error 2:: AHB modified regs 1:: AHB lock timeout error 0:: AHB HRESP_M error

0x00A8 STATUS_SEL

Type: Read/Write
Clock: AHB

Bits	Name	Description
6:0	STATUS_SEL	0x1 : route CMD status reg out 0x2 : route SET status reg out 0x4 : route SHD status reg out 0x8 : route HSR status reg out 0x10 : route TEX status reg out 0x20 : route BLD status reg out 0x40 : route BIF status reg out :

0x00AC STATUS_VALID

Type: Read
Clock: AHB

Bits	Name	Description
0	STATUS_VALID	Selected status is: 0 : not valid, cannot be read 1 : valid, can be read

0x00C4 Reserved.

0x00C8 Reserved.

0x00D0 Reserved.

0x00D4 Reserved.

0x00D8 Reserved.

0x00DC Reserved.

0x00E0 Reserved.

0x00E4 **Reserved.**

0x00E8 **Reserved.**

0x00F0 **Reserved.**

0x00F4 **Reserved.**

0x00F8 **Reserved.**

28 Mobile Display Digital Interface

28.1 ARM registers

Addresses in this file are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: MDDI
- Number of word addresses: 34
- Byte Address: 0x5100 - 0x5184
- OFFSET=CHIP_BASE+0x5100
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

28.1.1 MDDI read/write registers

0x0000 MDDI_CMD_REG

Type: Write
Clock: HCLK

31:16	RESERVED_BITS31_16	
15:8	COMMAND	Write values here to send MDDI commands. The values in bits 7:0 are parameter fields associated with the particular command. Valid Commands 0x01 -- Power Down 0x02 -- Power Up 0x03 -- Auto Hibernate: This command has an associated parameter. If this parameter value is zero, auto hibernations will be turned off. If this value is non-zero, the host core will wait that number of empty subframes before putting the MDDI link into hibernation 0x04 -- Reset 0x05 --Display Ignore: This command has an associated parameter. A ‘1’ in bit 0 of the parameter field indicates the host should ignore the display for client based wakeup. A ‘0’ in bit 0 of the parameter field indicates the host should not ignore the display for client based wakeup. 0x06 -- Reverse Encapsulation Request: This command has an associated field. The values passed in the parameters field will be output in the “Reverse Link Flags” field of the reverse encapsulation packet. This is done to request specific reverse packets 0x07 -- Round Trip Delay Measurement Request 0x08 -- Reserved 0x09 -- Link Activate 0x0A -- Periodic Reverse Encapsulation: This command has an associated parameter. The value in the parameter field indicates how many subframes to wait in between automatically sending reverse encapsulation packets. A Value of zero turns off periodic reverse encapsulation. 0x0B -- Reserved 0x0C -- Force New Reverse Pointer: This command is valid only for r2 devices. This command forces a new reverse pointer to be used for the next reverse packet received.
7:0	COMMAND_PARAMS	Values written here are parameters that may be associated with a particular command. The commands “Auto Hibernate”, “Display Ignore”, “Reverse Encapsulation Request”, and “Periodic Reverse Encapsulation” all make use of this parameter field.

0x0004 MDDI_VERSION_REG

Type: Read/Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:16	RESERVED_BITS31_16	Field has no function and should be set to zero for future compatibility
15:0	VERSION	This value is output as the version field of the sub-frame header packet.

0x0008 MDDI_PRI_PTR_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:0	PRI_LIST_ADDR	This address points to the header of the MDDI Primary Linked list. For optimum performance this address should be word aligned.

0x000C MDDI_SEC_PTR_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:0	SEC_LIST_ADDR	Secondary Linked list head pointer. For optimum performance this should be word aligned.

0x0010 MDDI_BPS_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0100

Bits	Name	Description
31:0	BYTES_PER_SF	The number of bytes per subframe.

0x0014 MDDI_SPM_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0001

Bits	Name	Description
31:16	RESERVED_BITS31_16	
15:0	SUBFRAME_PER_MF	This is the number of subframes per media frame.

0x0018 MDDI_INT_REG

Type: Read/ Write
Clock: HCLK

MDDI_INT_REG

Bits	Name	Description
31:19	RESERVED_BITS31_19	Reserved for future use.
18	RTD_FAILURE	RTD Failure, no response from client
17	NO_REQ_PKTS_PENDING	No packets requested through command interface are pending. This includes Timing Packet, Reverse Encapsulation, or Powerdown
16	SEC_LINK_LIST_DONE	Secondary linked list has been transmitted
15	PRI_LINK_LIST_DONE	Primary linked list has been transmitted
14	IN_HIBERNATION	Link is in hIbernation
13	LINK_ACTIVE	Link is active
12	DMA_FAILURE	A DMA failure occurred in the AHB interface
11	REV_OVERWRITE	Reverse pointer was overwritten
10	SEC_OVERWRITE	Secondary linked list pointer was overwritten
9	PRI_OVERWRITE	Primary linked list pointer was overwritten
8	MDDI_IN	MDDI receiver input (this can be used to interrupt the processor to wake up from hibernation by the client)
7	CRC_ERROR	Reverse Link CRC error
6	REV_OVERFLOW	Reverse Link data overflow in the reverse fifo
5	SEC_UNDERFLOW	Secondary data underflow in the forward data fifo
4	PRI_UNDERFLOW	Primary data underflow in the forward data fifo
3	DISP_REQ	Display has held the data line high long enough to indicate it is trying to wake up the host core

MDDI_INT_REG (Continued)

Bits	Name	Description
2	REV_DATA_AVAIL	Reverse packet completed, and new reverse data may be available
1	SEC_PTR_READ	Secondary linked list pointer has been read from the secondary linked list register, and is being processed. A new secondary linked list pointer can now be written to the register.
0	PRI_PTR_READ	Primary linked list pointer has been read from the primary linked list register, and is being processed. A new primary linked list pointer can now be written to the register.

0x001C MDDI_INTEN_REG

Type: Read/Write
Clock: HCLK
Reset State: 0x00000

Bits	Name	Description
31:19	RESERVED_BITS31_19	
18:0	INTERRUPT_EN	Interrupt enable mask for the interrupts listed above. Setting the bit to one will enable that interrupt.

0x0020 MDDI_REV_PTR_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:0	REV_DATA_PTR	Location in memory to put reverse data

0x0024 MDDI_REV_SIZE_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
31:16	RESERVED_BITS31_16	
15:0	REV_SIZE	Reverse Data size. This is the number of bytes available for MDDI host core to write reverse data at the address pointed to in the previous register.

0x0028 MDDI_STAT_REG

Type: Read
Clock: HCLK

Bits	Name	Description
31:15	RESERVED_BITS31_15	
14:0	STATUS	Status bits. Bit 14: REV_OVERFLOW_RESET Bit 13: DMA_ABORT Bit 12: CLIENT_WAKEUP_REQ Bit 11: RTD_MEAS_FAIL Bit 10: DO_HANDSHAKE Bit 9: SEND_POWER_DOWN Bit 8: SEND_REV_ENCAP_WITH_FLAGS Bit 7: SEND_TIMING_PKT Bit 6: SEC_LINK_LIST_DONE Bit 5: PRI_LINK_LIST_DONE Bit 4: IN_HIBERNATION Bit 3: NEW_SEC_PTR Bit 2: NEW_PRI_PTR Bit 1: NEW_REV_PTR Bit 0 : LINK_ACTIVE

0x002C MDDI_REV_RATE_DIV_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	REV_RATE_DIV	MDDI Reverse Rate Divisor to pass to client in reverse encapsulation packets.

0x0030 MDDI_REV_CRC_ERR_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	REV_CRC_ERR	Reverse CRC error count. This value can be reset by writing to the address.

0x0034 MDDI_TA1_LEN_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x05

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	TURN_AROUND_1_LEN	Reverse encapsulation Turn Around 1 length in forward bytes

0x0038 MDDI_TA2_LEN_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x09

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	TURN_AROUND_2_LEN	Reverse encapsulation Turn Around 2 Length in forward bytes

0x003C Reserved.

0x0040 Reserved.

0x0044 MDDI_REV_PKT_CNT_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	REV_PKT_CNT	Reverse link packets received. This counter can be reset by writing to this address.

0x0048 MDDI_DRIVE_HI_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0099

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	DRIVE_HI	Number of IO_CLK cycles to drive the data line high during the handshake processes bringing the link up

0x004C MDDI_DRIVE_LO_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0035

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	DRIVE_LO	Number of IO_CLK cycles to drive the data line low during the handshake processes bringing the link up

0x0050 MDDI_DISP_WAKE_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x003c

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	DISP_WAKE_LEN	Number of IO_CLK cycles to monitor a value of one on the input to determine that the Client is trying to wake up the host.

0x0054 MDDI_REV_ENCAP_SZ_REG

Type: Read/ Write
Clock: HCLKHCLK
Reset State: 0x000E

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	REV_ENCAP_SZ	Size in reverse bytes to allow in each reverse encapsulation packet. This host core automatically scales this to the correct number of forward link bytes based on the reverse rate divisor.

0x0058 MDDI_RTD_VAL_REG

Type: Read
Clock: HCLK
Reset State: 0x01

Bits	Name	Description
31:8	RESERVED_31_8	
7:0	RTD_MEAS_VAL	Measured value for the round trip delay measurement.

0x005C MDDI_MDP_VID_FMT_DES_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	VID_FMT_DES	MDP generated video packet format descriptor field value

0x0060 MDDI_MDP_VID_PIX_ATTR_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	VID_PIX_DATA_ATTR	MDP generated video packet pixel data attribute field

0x0064 MDDI_MDP_VID_CLIENTID_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0000

Bits	Name	Description
31:16	RESERVED_31_16	
15:0	VID_CLIENTID	MDP generated video packet client id field value

0x0068 MDDI_PAD_CTL_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x0000

MDDI_PAD_CTL_REG

Bits	Name	Description
31:23	RESERVED_31_23	Reserved bits
22:20	SPARE_STATUS	Spare Status input signals (not writable)
19	DATA_DRIVER_HW_ENA	Allows hardware to control the driver enable
18	DATA_DRIVER_ENA	Data driver enable
17	STB_DRIVER_HW_ENA	Allows hardware to control the driver enable
16	STB_DRIVER_ENA	Strobe driver enable
15	BANDGAP_ENA	Pad bandgap enable
14:13	DATA_DRV_INPUT_MUX	Mux to select the input for the data driver
12:11	STB_DRV_INPUT_MUX	Mux to select the input for the strobe driver
10	HW_CTL_OFFSET_RCV_ENA	Allow hardware to control the Offset Receiver Enable
9	OFFSET_RCV_ENA	Offset receiver enable signal

MDDI_PAD_CTL_REG (Continued)

Bits	Name	Description
8	HW_CTL_RCV_ENA	Allow Hardware to control the receiver enable.
7	RCV_ENA	Standard receiver enable signal
6:4	MDDI_SPARE_CTL	Spare pad control signals
3:2	MDDI_STB_CTL	Strobe driver output strength control signals. Set to 01. 00 : 200mV 01 : 250mV 10 : 300mV 11 : 350mV
1:0	MDDI_DOUT_CTL	Data driver output strength control signals. Set to 01. 00 : 200mV 01 : 250mV 10 : 300mV 11 : 350mV

0x006c MDDI_DRIVER_START_CNT_REG

Type: Read/ Write
Clock: HCLK
Reset State: 0x00100010

Bits	Name	Description
31:16	DATA_STARTUP_CNT	MDDI Data driver startup enable count
15:0	STB_STARTUP_CNT	MDDI Strobe driver startup enable count.

0x0070 MDDI_NEXT_PRI_PTR_REG

Type: Read
Clock: HCLK
Reset State: 0x0000 0000

Bits	Name	Description
31:0	NEXT_PRIMARY_POINTER	Returns the next primary pointer address to be read during processing of the primary linked list. Allows the processor to see what packet from the linked list is going to be processed next or to debug where an error occurred.

0x0074 MDDI_NEXT_SEC_PTR_REG

Type: Read
Clock: HCLK
Reset State: 0x0000 0000

Bits	Name	Description
31:0	NEXT_SECONDARY_POINTER	Returns the next secondary pointer address to be read during processing of the secondary linked list. Allows the processor to see what packet from the linked list is going to be processed next or to debug where an error occurred

0x0078 Reserved.

0x007c Reserved.

0x0080 Reserved.

0x0084 Reserved.

0x0088 MDDI_CURR_REV_PTR_REG

Type: Read
Clock: HCLK
Reset State: 0x0000 0000

This register is **valid only for revision r2 silicon and later**.

Bits	Name	Description
31:0	CURRENT_REVERSE_POINTER	This register returns the current active reverse pointer. This allows the processor to see where the next reverse data is going to be placed, or to help debug a reverse pointer problem.

0x008C MDDI_CORE_VER_REG

Type: Read
Clock: HCLK
Reset State: 0x0000 0008

This register is **valid only for revision r2 silicon and later**.

Bits	Name	Description
31:16	RESERVED_31_16	These bits are reserved and will return zero.
15:3	MDDI_CORE_VERS	These bits return the current mddi host core version. This will help software determine the MCCI core capabilities. This value is hard-coded at thesis. Since this will be the first chip to support this address, reading from previous versions will return all zero.
2:0	MDDI_SUB_REVISION	This is the sub-revision of the version.

29 MDDI Camera Interface

29.1 ARM registers

Addresses are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: MDDI_CAMIF
- Number of word addresses: 128
- Byte Address: 0x5200 - 0x53FC
- OFFSET= CHIP_BSE + 0x5200
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

29.1.1 Clocking considerations

If any mddi_camif registers are accessed without a PCLK, the register interface will not be able to respond, which will hang the MSM Bus, which will then cause the ARM to crash. This register interface depends on PCLK for clocking. The PCLK can be sourced from mulitple sources, and defaults to HCLK. When this block is being used to interface to an MDDI camera, then the PCLK should be programmed to use the output of an internal PLL. Both of these sources are “safe.”

When this interface is being used to access a camera using the parallel interface, care must be exercised to switch the PCLK source to HCLK in order to ensure that the absence of a clock from the external camera doesn’t cause the ARM to crash. This is workable, as these registers should be accessed only during initialization.

29.1.2 MDDI CAMIF control registers

0x0000 MDDI_CAMIF_CFG

Type: Read/Write
Clock: PCLK
Reset State: 0x00000082

MDDI_CAMIF_CFG

Bits	Name	Description
10	EXT_CAM_HSYNC_EDGE_SEL	When 1, samples HSYNC on rising edge of clock. When 0, samples HSYNC on falling edge of clock.
9	EXT_CAM_VSYNC_EDGE_SEL	When 1, samples VSYNC on rising edge of clock. When 0, samples VSYNC on falling edge of clock.
8	EXT_CAM_DATA_EDGE_SEL	When 1, samples data on rising edge of clock. When 0, samples data on falling edge of clock.
7	MDDI_CLK_CHICKEN_BIT	When set (1), this bit disables client HW control of CSC enable pin. Always disable this when not using the MDDI clock generated in the pad. When clear (0), this bit enables HW control of CSC enable.
6	OVERFLOW_INTR_ENA	Enable interrupt to signify that the circular buffer for FL register data has overflowed.
5	VBUF_ERR_INTR_ENA	Enable interrupt to signify that an underflow error occurred in the ping-pong video buffer and that the pclk rate needs to be increased.
4	WAKE_UP_INTR_ENA	Enable interrupt to signify that the MDDI link has been woken up
3	TX_INTR_ENA	Enable interrupt for RL register encapsulation packets transmitted by the client
2	RX_INTR_ENA	Enable interrupt for FL register encapsulation packets received by the client
1	CAM_SEL	Select MDDI or parallel camera: 0: MDDI Camera 1: Parallel camera
0	MTV_8B_10B_DATA	When set (1), 8 bits of incoming data are mapped to bits 9:2 of video data output to vfe. When clear (0), 10 bits of incoming data are mapped straight to the 10 bits going to the vfe.

0x0004 MDDI_CAMIF_INTR_STATUS

Type: Read
Clock: PCLK
Reset State: 0x00000000

MDDI_CAMIF_INTR_STATUS

Bits	Name	Description
4	OVERFLOW_STATUS	Status: 1 - RX FIFO overflowed; 0 - RX FIFO has not overflowed)
3	VBUF_ERR_INTR_STATUS	Status of interrupt that video ping-pong buffer underflow error has occurred: When 0, no interrupts received since last clear. When 1, at least one underflow error since last clear.
2	WAKE_UP_INTR_STATUS	Status of interrupt that MDDI has woken up: When 0, no interrupts received since last clear. When 1, link woke up at least one time since last clear.
1	TX_INTR_STATUS	Status of interrupt for RL register encapsulation packets sent by the client: When 0, no interrupts received since last clear. When 1, sent at least one packet since last clear.
0	RX_INTR_STATUS	Status of interrupt for FL register encapsulation packets received by the client: When 0, no interrupts sent since last clear. When 1, mddi camif received at least one FL register packet since last interrupt clear.

0x0008 MDDI_CAMIF_INTR_CLEAR

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
3	VBUF_ERR_INTR_CLEAR	Rising edge clears interrupt that indicated video ping-pong buffer underflow error has occurred.
2	WAKE_UP_INTR_CLEAR	Rising edge clears interrupt that indicated MDDI has woken up:
1	TX_INTR_CLEAR	Rising edge clears interrupt for RL register encapsulation packets sent by the client:
0	RX_INTR_CLEAR	Rising edge clears interrupt for FL register encapsulation packets received by the client:

0x000C Reserved.

0x0010 Reserved.

0x0014 **Reserved.**

0x0018 **Reserved.**

0x001C **Reserved.**

0x0020 **Reserved.**

0x0024 **Reserved.**

29.1.3 MDDI client configuration registers

0x0030 MDDI_CLIENT_STATUS

Type: Read
Clock: PCLK

Bits	Name	Description
31:16	FWD_PIXEL_CNT	Number of pixels in current/last packet (line)
15:0	FWD_VIDEO_DATA_FMT_DES	Packet header of current/last video packet

0x0034 MDDI_CLIENT_LINK_ACTIVE

Type: Read
Clock: PCLK

Bits	Name	Description
0	LINK_ACTIVE	Current status of link: 1 : Link is active 0 : Link is inactive

0x0038 MDDI_CLIENT_WAKEUP

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
0	WAKEUP	Wake client up out of hibernation 1: Wake up 0: Hibernate

29.1.4 MDDI client pad configuration registers

0x0040 MDDI_CLIENT_PAD_TESTMODE

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
0	MODE	When set (1), test mode. Pad signals in **MDDI_CLIENT_PAD_CNTL** register under SW control. When clear (0), normal mode. Pad inputs under client HW control.

0x0044 MDDI_CLIENT_PAD_STATUS

Type: Read
Clock: PCLK

Bits	Name	Description
2:0	SPARE_STATUS	Spare status signals to allow pad to send information about its current state to the core logic.

0x0048 Reserved.

0x0060 Reserved.

0x0064 Reserved.

0x0068 Reserved.

0x006C Reserved.

29.1.5 MDDI CAMIF reverse link encapsulation registers

0x0070 MDDI_TX_REG_BYTE_COUNT

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
15:0	BYTE_CNT	Number of bytes of register data to be transmitted back to the host.

0x0074 MDDI_REV_REG_RD_WR_INFO

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
15:0	REV_REG_RD_WR_INFO	Reverse register read/write information field.

0x0078 MDDI_REV_REG_START_ADDR

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	REV_REG_START_ADDR	Reverse register start address

0x007C MDDI_REV_REG_DATA1

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	REV_REG_DATA1	Reverse register data word 1.

0x0080 MDDI_REV_REG_DATA2

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	REV_REG_DATA2	Reverse register data word 2.

0x0084 MDDI_REV_REG_DATA3

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
31:0	REV_REG_DATA3	Reverse register data word 3

0x0088 MDDI_REV_REG_START

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
0	REV_REG_START	Setting (1) this bit sends reverse register packet during next hsync. Hardware clears (0) this bit when packet is sent.

29.1.5.1 Reverse link host register write procedure

There are some requirements for software when using this block. Software must do these steps in the following order :

1. When sending a RL encapsulation packet, check the REV_REG_START bit to make sure it is a ‘0’. This means there is not a packet currently waiting to go out.
2. Write the packet data to the BYTE_CNT, REV_REG_RD_WR_INFO, REV_REG_START_ADDR and REV_REG_DATA1, REV_REG_DATA2 and REV_REG_DATA3 registers.)
3. Write a ‘1’ to the REV_REG_START bit to start the access.
4. The access is complete when the REV_REG_START is cleared by the hardware, or if enabled, the packet sent interrupt is received.

29.1.6 MDDI CAMIF forward link encapsulation registers

0x008C MDDI_FWD_REG_STATUS

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
5:0	LAST_WR	Status end address of last complete packet in buffer.
6	OVERFLOW	Buffer status: 1 : overflowed 0 : Not overflowed

0x0090 MDDI_FWD_REG_CTRL

Type: Read/Write
Clock: PCLK
Reset State: 0x00000000

Bits	Name	Description
5:0	LAST_RD	Last address read from buffer.

0x0100 START_OF_MDDI_FWD_REG_BUF

Type: Read
Clock: PCLK

Bits	Name	Description
31:0	FWD_REG_BUF	This register contains the 128 byte ram (32 words by 32 bits) that saves forward link register and acknowledge data.

0x017C END_OF_FWD_REG_BUF

Type: Read
Clock: PCLK

Bits	Name	Description
31:0	END_OF_FWD_REG_BUF	This is a dummy register that signifies the end of the MDDI_FWD_REG_BUFFER

29.1.6.1 Forward link host register buffer read procedures

1. When an RX_INTR interrupt arrives, check the OVERFLOW register for overflow status and the LAST_WR and LAST_RD registers for the last write address and last read address.
2. If overflow occurred, write the value of LAST_WR to LAST_RD to exit overflow condition.
3. If there is no overflow, read the data out of the buffer, starting from the last read address and reading to the last write address, wrapping around at address 63 if the last write address is less than the last read address. The reported addresses will be indexed to zero, so they will need to be added to the START_OF_MDDI_FWD_REG_BUF address in order to correctly read from the buffer. The last address of the buffer is the END_OF_FWD_REG_BUF address.
4. Write the last address read into the LAST_RD field.
5. If the RX_INTR is cleared, then no other packets arrived while reading. If the interrupt is not cleared, then the LAST_WR address was updated with a new packet since it was last read and you need to start the process over at step one to get the new packet.

30 Mobile Display Processor (MDP)

30.1 ARM registers

Addresses are offsets from zero. When constants are exported, they are re-assigned to the appropriate base address.

- Block name: MDP
- Number of word addresses: 64
- Byte Address: 0x4900- 0x49FC
- OFFSET=CHIP_BASE+0x4900
- CHIP_BASE = 0x80000000
- chipaddr range = 14:2

30.1.1 MDP read/write registers

0x0004 MDP_COMMAND

Type: Read/Write
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:8	RESERVED_BITS31_8	Field has no function and should be set to zero for future compatibility
7:3	RESERVED_BITS7_3	Field has no function and should be set to zero for future compatibility. There are five spare read/write register bits in this location.
2	LOAD_RD_PTR	Setting (1) this bit loads the read pointer with the value specified in MDP_ADDRESS_14 (INIT_SST_READ_PTR). Returns 1 if read pointer has been set by this mechanism since reset. Returns zero if register has been cleared (0) to zero and at reset.
1	RESERVED_BIT_1	Field has no function and should be set to zero for future compatibility. Always returns zero.
0	BEGIN_SCRIPT	Setting (1) this bit initiates the MDP command (script). Always returns zero.

0x0008 MDP_SCRIPT_ADDR

Type: Read/ Write
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:0	CIP_START_ADDR	The MDP assumes a script in memory defines how the MDP operates. CIP_START_ADDR is the beginning of that script. All scripts are 4-byte words that start at 32-bit boundaries, so the bottom two bits are always “00”. When read, this register returns the current script address.

0x000C Reserved.

0x0010 MDP_SYNC_CONFIG

Type: Read/ Write
Clock: HCLK

Bits	Name	Description
31:18	RESERVED_BITS31_18	
17:16	SST_SEL_VSYNC	Setting this bit selects the right source for the LCD vsync ”00” : primary vsync “01” : secondary vsync “10” : non-wired vsync (software maintained)
15:0	SST_TERM_DIV_CNT	This PRE_SCALE value is to be programmed to the number of HCLK cycles required for the LCD (running on its own clock) to refresh a single row of the panel. The ratio of HLKC frequency to LCD panel frequency divided by the number of rows in the LCD panel. For example, if the AHB clock (HCLK) frequency is 75 MHz, and the LCD is a 60 Hz EVGA with 220 rows and 176 columns, then the value to be programmed here is 75MHz / (60 Hz x 220) = 5682,.

0x0014 MDP_SYNC_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
31:24	RESERVED_BITS31_24	
23:16	FRAME_COUNT	This is a free running counter that is incremented once per frame. There is no known use for it, but software may find one as time progresses
15:12	RESERVED_BITS15_12	
11:0	LINE_COUNT	This is the present state of the line counter, or read pointer. Scripts that progress based on WAIT_LINE_xxx use this counter.

0x0018 MDP_SST_REFRESH_COUNT

Type: Read
Clock: HCLK
Reset State:

Bits	Name	Description
31:24	RESERVED_BITS31_24	
23:0	REFRESH_COUNT	This counter shows the number. of AHB clocks (HCLK) in a refresh cycle (the time between VSYNCs) of the LCD panel. The intended purpose of this register is for software development. The MDP does not use this counter during operation.

0x001C MDP_SCRIPT_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
31:1	RESERVED_BITS31_1	
0	SCRIPT_STATUS	When a script is active, this bit will be set. When a script is completed (the script encounters the HALT command), this bit is cleared. The register allows software to poll for completion completion.

0x0020 MDP_INTR_ENABLE

Type: Read/ Write
Clock: HCLK

Bits	Name	Description
31:6	RESERVED_BITS31_6	
5:0	INT_ENABLE	Writing '1' to each bit will enable the respective interrupt from the interrupting the ARM. Clearing the bit will disable the interrupt. Bit 5: Master Error Bit 4: Slave Error Bit 3: End Script Bit 2: Job Done Bit 1: Interrupt from MDP Script Bit 0: VSYNC

0x0024 MDP_INTR_STATUS

Type: Read
Clock: HCLK

Bits	Name	Description
31:6	RESERVED_BITS31_6	
5:0	STATUS	If a MDP interrupt has occurred, then reading this register will indicate what caused the interrupt since the particular bit will be set to '1'. If multiple events trigger of the interrupt, then the multiple bits will be set, provided the enable for that interrupt is set. Bit 5: Master Error Bit 4: Slave Error Bit 3: End Script Bit 2: Job Done Bit 1: Interrupt from MDP Script Bit 0: VSYNC

0x0028 MDP_INTR_CLEAR

Type: Write
Clock: HCLK

Bits	Name	Description
31:6	RESERVED_BITS31_6	
5:0	CLEAR	To clear and interrupt, ensure that the source of the interrupt has gone away and then write to the respective bit. Bit 5: Master Error Bit 4: Slave Error Bit 3: End Script Bit 2: Job Done Bit 1: Interrupt from MDP Script Bit 0: VSYNC

0x0038 MDP_ADDRESS_14

Type: Read/ Write
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:12	RESERVED_BITS_31_12	
11:0	INIT_SST_READ_PTR	Value is to be loaded into the sst read pointer (line counter) upon issuing a LD_RD_PTR command (writing 0x4 to MDP_COMMAND register). This feature is used on panels that do not provide a vsync reference for the MDP to synchronize to, but do provide some other mechanism (for example, a read register in the LCD controller). The procedure is as follows: 1.) Set-up the MAX_SST_READ_PTR in MDP_ADDRESS_15 to reflect the number of lines in the LCD. 2.) Set the SST_SEL_VSYNC in MDP_SYNC_CONFIG register to point to “10” or non-wired vsync. 3.) Read the location of the read pointer from the LCD controller (through EBI2 or though MDDI). 4.) Write read pointer location to this register, 5.) Issue a LD_RD_PTR command through the MDP_COMMAND register. The above resynchronization operation will have to repeated periodically, based upon the drifts in HCLK verses the LCDs clock and accuracy in the MDP’s counters (how well they model the LCD position). Synchronization is easily monitored by comparing the MDP’s read pointer to that of the LCD. Note: This register is identical to MDP_INIT_SST_RD_PTR in later chips, but the address file was frozen before this feature was added.

0x003C MDP_ADDRESS_15

Type: Read/ Write
Clock: HCLK
Reset State: 0x0

Bits	Name	Description
31:12	RESERVED_BITS31_12	
11:0	MAX_SST_READ_PTR	These bits set the maximum value for the sst line counter (read pointer). When the sst line count reaches this, line counter rolls over to zero. This feature is used to allow the MDP to stay synchronized even when a vsync is missing for some reason. The feature is also used when there is no physical vsync to reset the line counter. Note: This register is identical to MDP_MAX_SST_RD_PTR in later chips, but the address file was frozen before this feature was added.

0x0040 Reserved.

0x0044 Reserved.

0x0048 Reserved.

0x004C Reserved.

Registers by Name