MT7628DAN DATASHEET

Overview

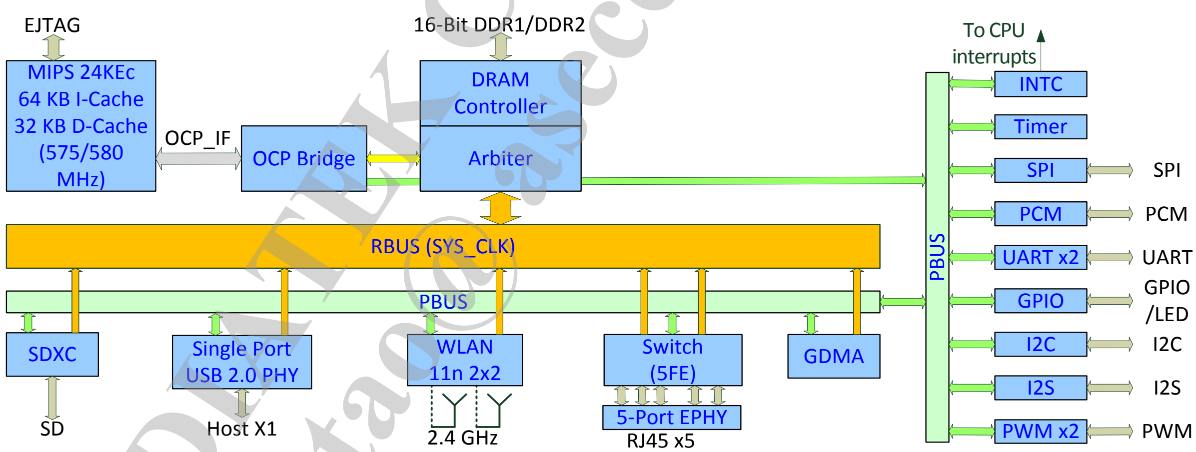
The MT7628DAN router-on-a-chip includes an 802.11n MAC and baseband, a 2.4 GHz radio and FEM, a 575/580 MHz MIPS® 24K™ CPU core, a 5-port 10/100 fast ethernet switch. The MT7628DAN includes everything needed to build an AP router from a single chip. The embedded high performance CPU can process advanced applications effortlessly, such as routing, security and VoIP. The MT7628DAN also includes a selection of interfaces to support a variety of applications, such as a USB port for accessing external storage.

Applications:
- Routers
- NAS devices
- Dual band concurrent routers

Features

- Embedded MIPS24KEc (575/580 MHz) with 64 KB I-Cache and 32 KB D-Cache
- 2T2R 2.4 GHz with 300 Mbps PHY data rate
- Legacy 802.11b/g and HT 802.11n modes
- 20/40 MHz channel bandwidth
- Reverse Data Grant (RDG)
- Maximal Ratio Combining (MRC)
- Space Time Block Coding (STBC)
- MCM 64 Mbytes DDR2 KGD
- SPI/SD-XC
- x1 USB 2.0 Host, x1 PCIe Root Complex
- 5-port 10/100 FE PHY
- Internet Of Thing
- An optimized PMU
- Green AP
  - Intelligent Clock Scaling (exclusive)
  - DDRII: ODT off, Self-refresh mode
- I2C, I2S, SPI, PCM, UART, JTAG, GPIO
- 16 Multiple BSSID
- WEP64/128, TKIP, AES, WPA, WPA2, WAPI
- QoS: WMM, WMM-PS
- WPS: PBC, PIN
- Voice Enterprise: 802.11k+r
- AP Firmware: Linux 2.6 SDK, eCOS with IPv6

Functional Block Diagram



Ordering Information

Part Number	Package (Green/RoHS Compliant)
MT7628DAN	DR-QFN 156 pin (12 mm x 12 mm)

Table of Contents

Table of Figures

List of Tables

1. Main Features

The following table covers the main features offered by the MT7628DAN. Overall, the MT7628DAN supports the requirements of an entry-level AP/router.

Features	MT7628DAN
CPU	MIPS24KEc (575/580 MHz)
Total DMIPs	580 x 1.6 DMIPs
I-Cache, D-Cache	64 KB, 32 KB
L2 Cache	n/a
Memory	
DDR2	512 Gb, 193 MHz, MCM
SPI Flash	3B addr mode (max 128Mbit) 4B addr mode (max 512Mbit)
SD	SD-XC (class 10)
RF	2T2R 802.11n 2.4 GHz
PCIe	1
USB 2.0	1
Switch	5p FE SW
I2S	1
PCM	1
I2C	1
UART	2 (Lite)
JTAG	1
Package	DR-QFN156- 12 mm x 12 mm

Table 1-1 Main Features

2. Pins

2.1 MT7628DAN DR-QFN (12 mm x 12 mm) 156-Pin Package Diagram

2.1.1 Up-left side

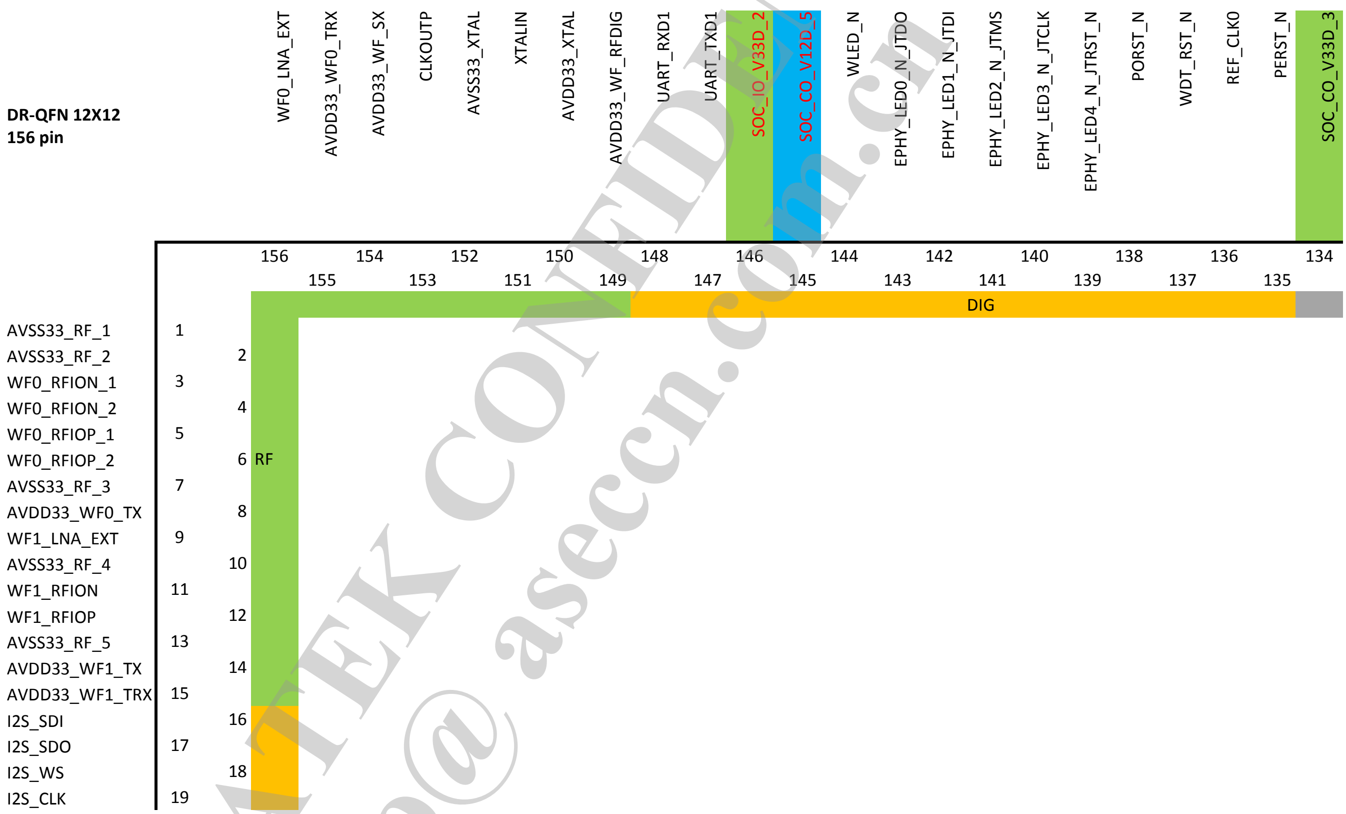


Figure 2-1 MT7628DAN DR-QFN Pin Diagram (up-left view)

2.1.2 Down-left side

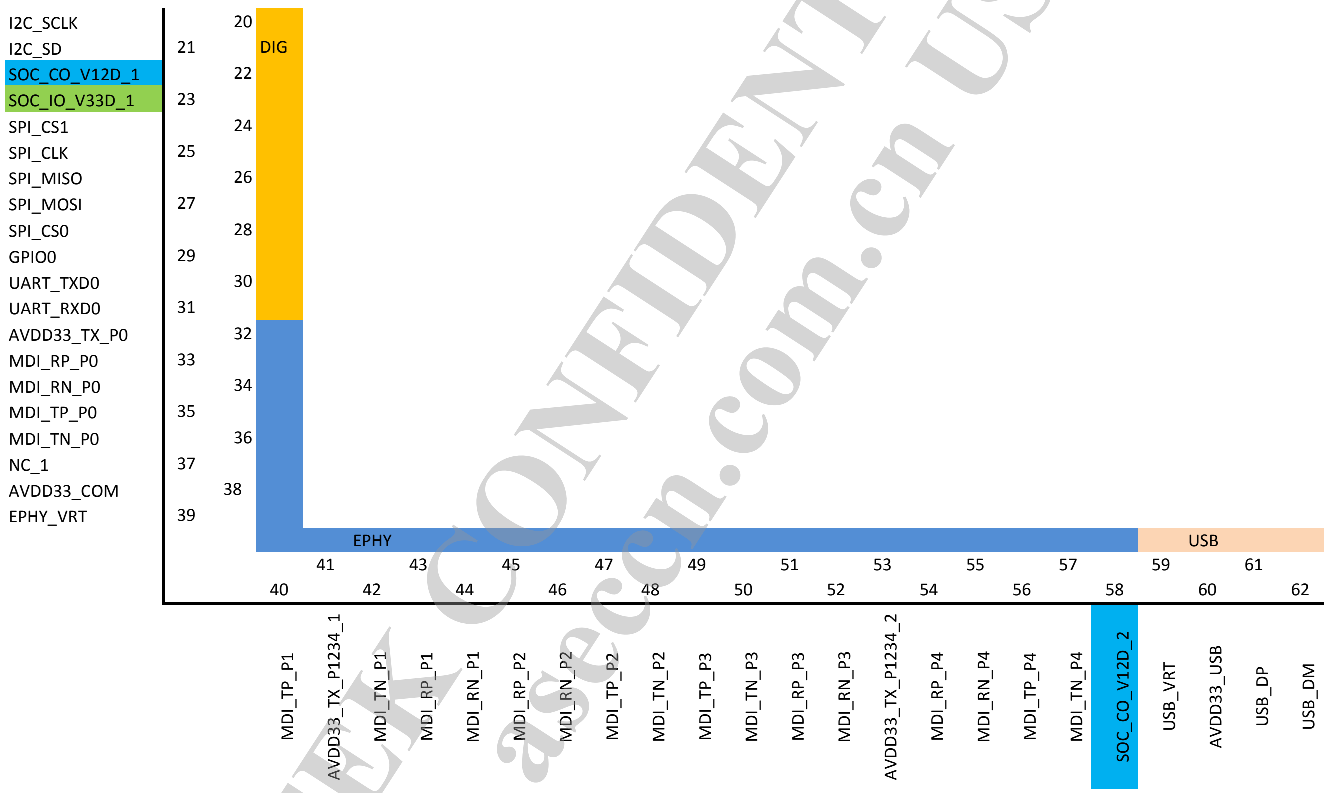


Figure 2-2 MT7628DAN DR-QFN Pin Diagram (down-left view)

2.1.3 Down-right side

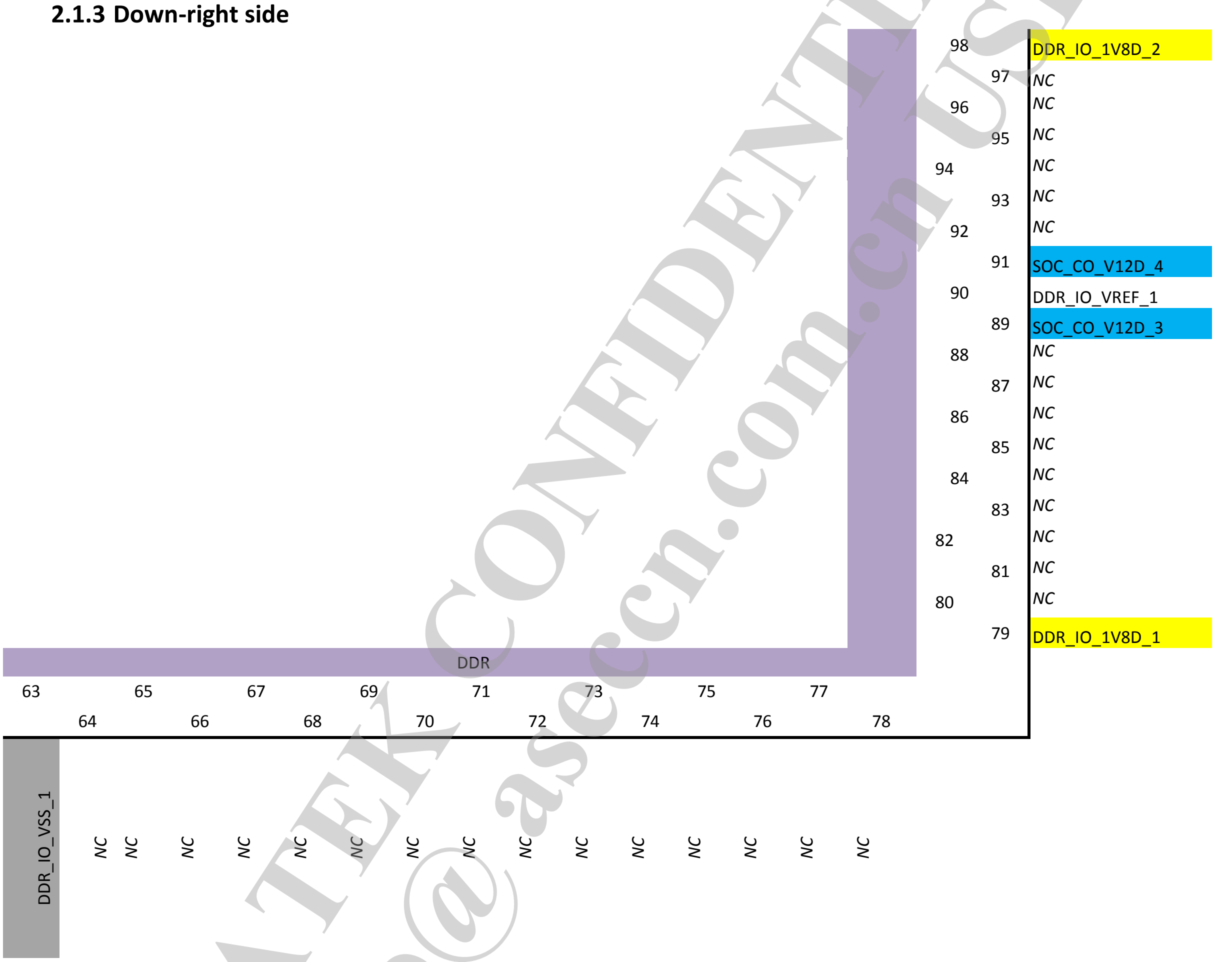


Figure 2-3 MT7628DAN DR-QFN Pin Diagram (down-right view)

2.1.4 Up-right side

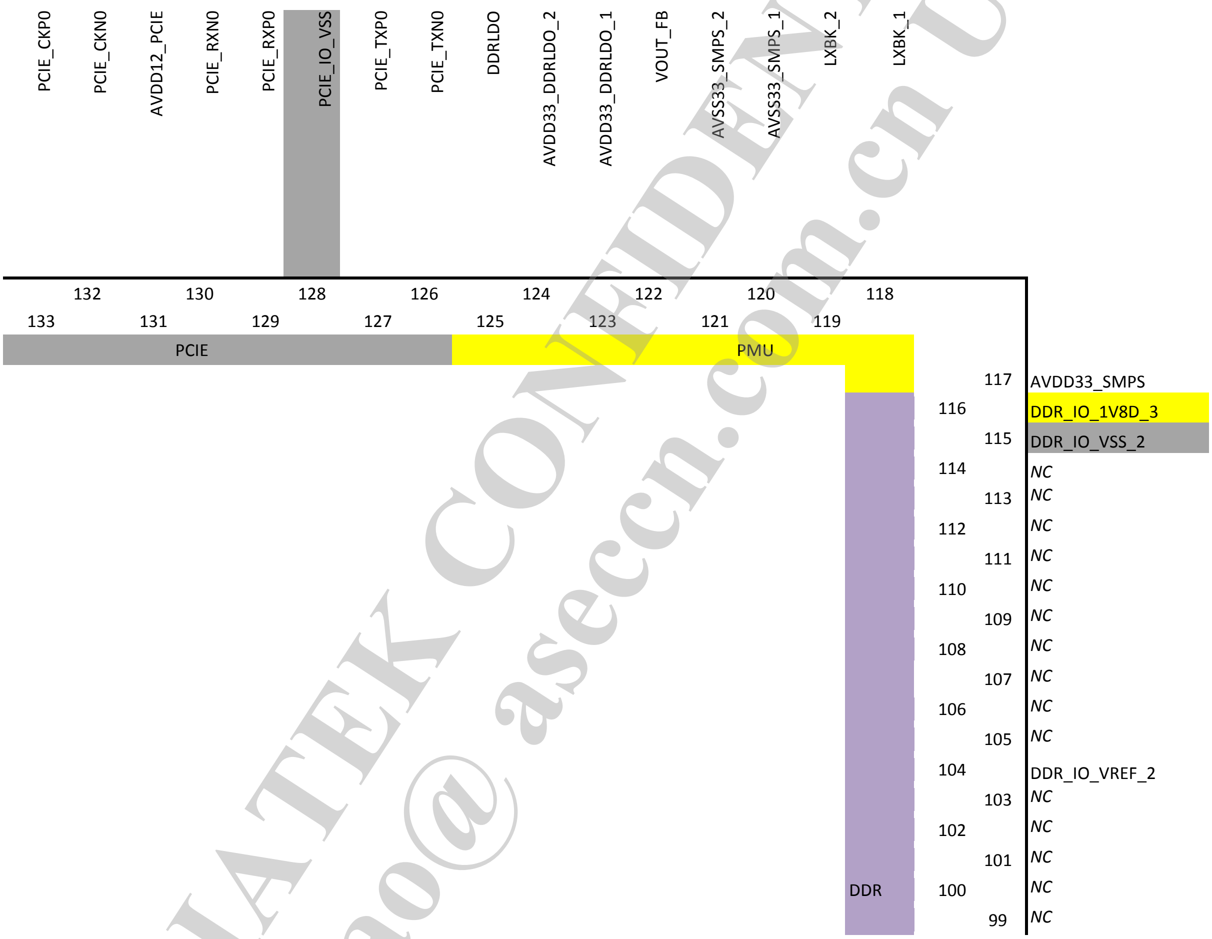


Figure 2-4 MT7628DAN DR-QFN Pin Diagram (up-right view)

2.1.5 Pin Description

Pins	Name	Type	Driv.	Description
RF				
3,4	WF0_RFION_1 WF0_RFION_2	A		WF0 main path RF I/O
5,6	WF0_RFIOP_1 WF0_RFIOP_2	A		WF0 main path RF I/O
11	WF1_RFION	A		WF1 main path RF I/O
12	WF1_RFIOP	A		WF1 main path RF I/O
9	WF1_LNA_EXT	A		WF1 aux. path LNA input
156	WF0_LNA_EXT	A		WF0 aux. path LNA input
151	XTALIN	I		Crystal oscillator input
153	CLKOUTP	O		XO reference clock output
150	AVDD33_XTAL	P		3.3V XTAL Power Supply Pin
152	AVSS33_XTAL	G		3.3V XTAL Ground Pin
8	AVDD33_WF0_TX	P		3.3V RF Channel 0 Suppoly Power
14	AVDD33_WF1_TX	P		3.3V RF Channel 1 Suppoly Power
15	AVDD33_WF1_TRX	P		1.65V to 3.3V RF Channel 1 Suppoly Power
149	AVDD33_WF_RFDIG	P		1.65V to 3.3V RF DIG and AFE Suppoly Power
154	AVDD33_WF_SX	P		1.65V to 3.3V RF Supply Power
155	AVDD33_WF0_TRX	P		1.65V to 3.3V RF Channel 0 Suppoly Power
1,2 7,10,13	AVSS33_RF	G		3.3V RF Shielding Ground Pin
WLAN LED				
144	WLED_N	O	4 mA	WLAN Activity LED
UART0 Lite				
31	UART_RXD0	I	4 mA	UART0 Lite RXD
30	UART_TXD0	O, IPD	4 mA	UART0 Lite TXD
UART1 Lite				
147	UART_TXD1	O, IPU	4 mA	UART1 Lite TXD

148	UART_RXD1	I	4 mA	UART1 Lite RXD
I2S				
16	I2S_SDI	I	4 mA	I2S data input
17	I2S_SDO	O, IPD	4 mA	I2S data output
18	I2S_WS	I/O	4 mA	I2S word select
19	I2S_CLK	I/O	4 mA	I2S clock
I2C				
21	I2C_SD		4 mA	I2C Data
20	I2C_SCLK	I/O	4 mA	I2C Clock
SPI				
26	SPI_MISO	I/O	4 mA	SPI Master input/Slave output
27	SPI_MOSI	I/O, IPD	4 mA	SPI Master output/Slave input
25	SPI_CLK	O, IPU	4 mA	SPI clock
28	SPI_CS0	O	4 mA	SPI chip select0
24	SPI_CS1	O, IPD	4 mA	SPI chip select1
GPIO				
29	GPIO0	I/O, IPD	4 mA	General Purpose I/O
5-Port EPHY				
143	EPHY_LED0 _N_JTDO	I/O	4 mA	10/100 PHY Port #0 activity LED, JTAG_TDO
142	EPHY_LED1 _N_JTDI	I/O	4 mA	10/100 PHY Port #1 activity LED, JTAG_TDI
141	EPHY_LED2 _N_JTMS	I/O	4 mA	10/100 PHY Port #2 activity LED, JTAG_TMS
140	EPHY_LED3 _N_JTCLK	I/O	4 mA	10/100 PHY Port #3 activity LED, JTAG_CLK
139	EPHY_LED4 _N_JTRST_N	I/O,	4 mA	10/100 PHY Port #4 activity LED, JTAG_TRST_N
39	EPHY_VRT	A		Connect to an external resistor to provide accurate bias current
33	MDI_RP_P0	A		10/100 PHY Port #0 RXN
34	MDI_RN_P0	A		10/100 PHY Port #0 RXP
35	MDI_TP_P0	A		10/100 PHY Port #0 TXN
36	MDI_TN_P0	A		10/100 PHY Port #0 TXP

40	MDI_TP_P1	A		10/100 PHY Port #1 RXN
42	MDI_TN_P1	A		10/100 PHY Port #1 RXP
43	MDI_RP_P1	A		10/100 PHY Port #1 TXN
44	MDI_RN_P1	A		10/100 PHY Port #1 TXP
45	MDI_RP_P2	A		10/100 PHY Port #2 RXN
46	MDI_RN_P2	A		10/100 PHY Port #2 RXP
47	MDI_TP_P2	A		10/100 PHY Port #2 TXN
48	MDI_TN_P2	A		10/100 PHY Port #2 TXP
49	MDI_TP_P3	A		10/100 PHY Port #3 RXN
50	MDI_TN_P3	A		10/100 PHY Port #3 RXP
51	MDI_RP_P3	A		10/100 PHY Port #3 TXN
52	MDI_RN_P3	A		10/100 PHY Port #3 TXP
54	MDI_RP_P4	A		10/100 PHY Port #4 RXN
55	MDI_RN_P4	A		10/100 PHY Port #4 RXP
56	MDI_TP_P4	A		10/100 PHY Port #4 TXN
57	MDI_TN_P4	A		10/100 PHY Port #4 TXP
32	AVDD33_TX_P0	P		3.3V Supply Power for P0
38	AVDD33_COM	P		3.3V Supply Power for EPHY COM
41, 53	AVDD33_TX_P1234_1 AVDD33_TX_P1234_2	P		3.3V Supply Power for P1 ~ P4
Misc.				
136	REF_CLKO	O, IPD	4 mA	Reference Clock Ouptut
138	PORST_N	I, IPU	4 mA	Power on reset
137	WDT_RST_N	O	4 mA	Watchdog timeout reset
USB PHY				
60	AVDD33_USB	P		3.3 V USB PHY analog power supply
59	USB _VRT	I/O		Connect to an external 5.1 kΩ resistor for band-gap reference circuit
62	USB_DM	I/O		USB Port0 data pin Data-
61	USB _DP	I/O		USB Port0 data pin Data+

PCIe PHY				
135	PERST_N	O, IPD	4mA	PCIe device reset
131	AVDD12_PCIE	P		1.2 V PCIE PHY digital power supply
134	AVDD33_PCIE	P		3.3 V PCIE PHY analog power supply
128	PCIE_IO_VSS	G		PCIE PHY Ground Pin
133	PCIE_CKP0	I/O		External reference clock output (positive)
132	PCIE_CKN0	I/O		External reference clock output (negative)
127	PCIE_TXP0	I/O		PCIe0 differential transmit TX +
126	PCIE_TXN0	I/O		PCIe0 differential transmit TX -
129	PCIE_RXP0	I/O		PCIe0 differential receiver RX +
130	PCIE_RXN0	I/O		PCIe0 differential receiver RX -
DDR2 MCM				
63 115	DDR_IO_VSS_1 DDR_IO_VSS_2	G		DDR IO Ground pins
79 98 116	DDR_IO_1V8D_1 DDR_IO_1V8D_2 DDR_IO_1V8D_3	P		DDR io Supply power
90 104	DDR_IO_VREF_1 DDR_IO_VREF_2	A		DDR reference voltage
PMU				
118 119	LXBK_1 LXBK_2	O		Buck Switching node
122	VOUT_FB	A		Buck vout feedback pin
117	AVDD33_SMPS	P		Buck 3.3V Supply power
120 121	AVSS33_SMPS_1 AVSS33_SMPS_2	G		Buck Gound pin
123 124	AVDD33_DDRLDO_1 AVDD33_DDRLDO_2	P		DDRLDO 3.3V Supply power
125	DDRLDO	O		DDRLDO 1.8V output voltage

Power			
23	SOC_IO_V33D_1	P	3.3 V digital I/O power supply
146	SOC_IO_V33D_2		
22	SOC_CO _V12D_1	P	1.2 V digital core power supply
58	SOC_CO _V12D_2		
89	SOC_CO _V12D_3		
91	SOC_CO _V12D_4		
145	SOC_CO _V12D_5		
EPAD	GND	G	Ground pin
NC			
37	NC_1	NC	No connected
65	NC	NC	No connected
114	NC	NC	No connected
67	NC	NC	No connected
111	NC	NC	No connected
110	NC	NC	No connected
68	NC	NC	No connected
112	NC	NC	No connected
66	NC	NC	No connected
70	NC	NC	No connected
109	NC	NC	No connected
73	NC	NC	No connected
106	NC	NC	No connected
105	NC	NC	No connected
69	NC	NC	No connected
107	NC	NC	No connected
71	NC	NC	No connected
83	NC	NC	No connected
96	NC	NC	No connected
85	NC	NC	No connected
92	NC	NC	No connected

94	NC	NC	No connected
84	NC	NC	No connected
95	NC	NC	No connected
86	NC	NC	No connected
93	NC	NC	No connected
82	NC	NC	No connected
97	NC	NC	No connected
87	NC	NC	No connected
88	NC	NC	No connected
80	NC	NC	No connected
101	NC	NC	No connected
99	NC	NC	No connected
100	NC	NC	No connected
74	NC	NC	No connected
81	NC	NC	No connected
75	NC	NC	No connected
102	NC	NC	No connected
77	NC	NC	No connected
76	NC	NC	No connected
64	NC	NC	No connected
108	NC	NC	No connected
78	NC	NC	No connected
72	NC	NC	No connected
113	NC	NC	No connected
103	NC	NC	No connected
Total: 156 pins			

Note:
- IPD : Internal pull-down
- IPU : Internal pull-up
- I : Input
- O : Output
- IO : Bi-directional
- P : Power

G : Ground
NC : Not connected

2.2 Pin Sharing Schemes

Some pins are shared with GPIO to provide maximum flexibility for system designers. The MT7628DAN provides up to 41 GPIO pins. Users can configure GPIO1_MODE and GPIO2_MODE registers in the System Control block to specify the pin function, or they can use the registers specified below. For more information, see the Programmer's Guide. Unless specified explicitly, all the GPIO pins are in input mode after reset.

2.2.1 GPIO pin share scheme

I/O Pad Group	Normal Mode	GPIO Mode
UART1	UART_RXD1	GPIO#46
	UART_TXD1	GPIO#45
WLED_AN	WLED_N	GPIO#44
P0_LED_AN	EPHY_LED0_N_JTDO	GPIO#43
P1_LED_AN	EPHY_LED1_N_JTDI	GPIO#42
P2_LED_AN	EPHY_LED2_N_JTMS	GPIO#41
P3_LED_AN	EPHY_LED3_N_JTCLK	GPIO#40
P4_LED_AN	EPHY_LED4_N_JTRST_N	GPO#39
WDT	WDT_RST_N	GPO#38
REFCLK	REF_CLKO	GPIO#37
PERST	PERST_N	GPIO#36
UART0	UART_RXD0	GPIO#13
	UART_TXD0	GPIO#12
GPIO	GPIO0	GPIO#11
SPI	SPI_CS0	GPIO#10
	SPI_MISO	GPIO#9
	SPI_MOSI	GPIO#8
	SPI_CLK	GPIO#7
SPI_CS1	SPI_CS1	GPIO#6
I2C	I2C_SD	GPO#5
	I2C_SCLK	GPO#4
I2S	I2S_CLK	GPIO#3
	I2S_WS	GPIO#2
	I2S_SDO	GPIO#1
	I2S_SDI	GPO#0

2.2.2 UART1 pin share scheme

Controlled by the UART1_MODE register.

Pin Name	2'b00 UART-Lite #1	2'b01 GPIO	2'b10 PWM	2'b11 TRX_SW
UART1_RXD	UART1_RXD	GPIO#46	PWM_CH1	
UART1_TXD	UART1_TXD	GPIO#45	PWM_CH0	

2.2.3 MT7628DAN EPHY LED pin share scheme

Controlled by the P#_LED_AN_MODE registers

Pin Name	Bootstrapping (DBG_JTAG_MODE=0)	Bootstrapping (DBG_JTAG_MODE=1)	
		P4_LED_AN_MODE =2'b00	P4_LED_AN_MODE =2'b01
EPHY_LED4_N_JTRST_N	JTAG_RST_N	EPHY_LED4_N	GPIO#39
		P3_LED_AN_MODE =2'b00	P3_LED_AN_MODE =2'b01
EPHY_LED3_N_JTCLK	JTAG_CLK	EPHY_LED3_N	GPIO#40
		P2_LED_AN_MODE =2'b00	P2_LED_AN_MODE =2'b01
EPHY_LED2_N_JTMS	JTAG_TMS	EPHY_LED2_N	GPIO#41
		P1_LED_AN_MODE =2'b00	P1_LED_AN_MODE =2'b01
EPHY_LED1_N_JTDI	JTAG_TDI	EPHY_LED1_N	GPIO#42
		P0_LED_AN_MODE =2'b00	P0_LED_AN_MODE =2'b01
EPHY_LED0_N_JTDO	JTAG_TDO	EPHY_LED0_N	GPIO#43

2.2.4 MT7628DAN WLAN LED pin share scheme

Controlled by the WLED_AN_MODE registers

Pin Name	2'b00	2'b01
WLED_N	WLED_N	GPIO#44

2.2.5 PERST_N pin share scheme

Controlled by the PERST_ MODE register.

Pin Name	1'b0	1'b1
PERST_N	PERST_N	GPIO#36

2.2.6 WDT_RST_N pin share scheme

Controlled by the WDT _MODE register.

Pin Name	1'b0	1'b1
WDT_RST_N	WDT_RST_N	GPIO#38

2.2.7 REF_CLKO pin share scheme

Controlled by the REFCLK _MODE register.

Pin Name	1'b0	1'b1
REF_CLKO	REF_CLKO	GPIO#37

2.2.8 UART0 pin share scheme

Controlled by the UART0 _MODE register.

Pin Name	1'b0	1'b1
UART_TXD0	UART_TXD0	GPIO#12
UART_RXD0	UART_RXD0	GPIO#13

2.2.9 GPIO0 pin share scheme

Controlled by GPIO_MODE register.

Pin Name	2'b00	2'b01	2'b10	2'b11
GPIO0	GPIO#11	GPIO#11	REF_CLKO	PERST_N

2.2.10 SPI pin share scheme

Controlled by SPI_ MODE register.

Pin Name	1'b0	1'b1
SPI_CLK	SPI_CLK	GPO#7
SPI_MOSI	SPI_MOSI	GPO#8
SPI_MISO	SPI_MISO	GPIO#9
SPI_CS0	SPI_CS0	GPIO#10

2.2.11 SPI_CS1 pin share scheme

Controlled by SPI_CS1_MODE register.

Pin Name	2'b00	2'b01	2'b10
SPI_CS1	SPI_CS1	GPIO#6	REF_CLKO

2.2.12 I2C pin share scheme

Controlled by I2C_MODE register.

Pin Name	2’b00	2’b01
I2C_SCLK	I2C_SCLK	GPIO#4
I2C_SD	I2C_SD	GPIO#5

2.2.13 I2S pin share scheme

Controlled by I2S_MODE register.

Pin Name	2’b00	2’b01	2’b10
I2S_SDI	I2C_SCLK	GPIO#0	PCMDRX
I2S_SDO	I2C_SD	GPIO#1	PCMDTX
I2S_WS	I2C_SCLK	GPIO#2	PCMCLK
I2S_CLK	I2C_SD	GPIO#3	PCMFS

2.2.14 Pin share function description

Pin Share Name	I/O	Pin Share Function description
PCMDTX	O	PCM Data Transmit DATA signal sent from the PCM host to the external codec.
PCMDRX	I	PCM Data Receive DATA signal sent from the external codec to the PCM host.
PCMCLK	I/O	PCM Clock The clock signal can be generated by the PCM host (Output direction), or provided by an external clock (input direction). The clock frequency should match the slot configuration of the PCM host. e.g. 4 slots, PCM clock out/in should be 256 kHz. 8 slots, PCM clock out/in should be 512 kHz. 16 slots, PCM clock out/in should be 1.024 MHz. 32 slots, PCM clock out/in should be 2.048 MHz. 64 slots, PCM clock out/in should be 4.096 MHz. 128 slots, PCM clock out/in should be 8.192 MHz.
PCMFS	I/O	PCM SYNC signal. In our design, the direction of this signal is independent of the direction of PCMCLK. Its direction and mode is configurable.
PWM_CH0	O	Pulse Width Modulation Channle 0
PWM_CH1	O	Pulse Width Modulation Channle 1

2.3 Bootstrapping Pins Description

Pin Name	Boot Strapping Signal Name	Description
UART_TXD1	DBG_JTAG_MODE	0: JTAG_MODE 1: EPHY_LED (default)
PERST_N	XTAL_FREQ_SEL	0: 25 MHz DIP 1: 40 MHz SMD
I2S_SDO	DRAM_TYPE	0: DDR2 [note] It needs to be pull-low for 7628DAN which only supports DDR2.
{SPI_MOSI SPI_CLK, SPI_CS1}	CHIP_MODE[2:0]	A vector to set chip function/test/debug modes. 000: Boot from PLL (boot from SPI 3-Byte Addr) 001: Boot from PLL (boot from SPI 4-Byte Addr) 010: Boot from XTAL (boot from SPI 3-Byte Addr) 011: Boot from XTAL (boot from SPI 4-Byte Addr)
UART_TXD0	EXT_BGCK	1: Test Mode 0: Normal (default)

3. Maximum Ratings and Operating Conditions

3.1 Absolute Maximum Ratings

I/O supply voltage	3.63 V
Input, Output, or I/O Voltage	GND -0.3 V to Vcc +0.3 V

Table 3-1 Absolute Maximum Ratings

3.2 Maximum Temperatures

Maximum Junction Temperature (Plastic Package)	125 °C
Maximum Lead Temperature (Soldering 10 s)	260 °C

Table 3-2 Maximum Temperatures

3.3 Operating Conditions

I/O supply voltage	3.3 V +/- 10%
DDR2 supply voltage	1.8 V +/- 5%
Core supply voltage	1.2 V +/- 10%
Ambient Temperature Range	-20 to 55 °C

Table 3-3 Operating Conditions

Table 3-4 Thermal Characteristics

3.4 Storage Conditions

The calculated shelf life in a sealed bag is 12 months if stored between 0 °C and 40 °C at less than 90% relative humidity (RH). After the bag is opened, devices that are subjected to solder reflow or other high temperature processes must be handled in the following manner:

- Mounted within 168 hours of factory conditions, i.e. < 30 °C at 60% RH.
- Storage humidity needs to maintained at < 10% RH.
- Baking is necessary if the customer exposes the component to air for over 168 hrs, baking conditions: 125 °C for 8 hrs.

3.5 External Xtal Specfication

Frequency	25 MHz/ 40 Mhz
Frequency offset	+/-7ppm @ 25 ℃ +/-15ppm @ -40~85 ℃
Load Capacitance (CL)	13pF
Shunt Capacitance (Co)	7.0 pF MAX

Pulling Sensitivity (TS)	20ppm /pF (Load @ 13pF)

Table 3-5 External Xtal Specifications

3.6 DC Electrical Characteristics

MT7628DAN (2T2R(HT40/MCS15), LAN x 4,WANx1, LAN to WAN, USB (SAMBA), PCIe OFF)						
Parameters	Sym	Conditions	Min	Typ	Max	Unit
3.3 V supply voltage (IO)	Vddc33		2.97	3.3	3.63	V
1.8 V supply voltage (DDR2)	Vdd18		1.71	1.8	1.89	V
1.2 V supply voltage	Vdd12		1.14	1.2	1.32	V
3.3 V current consumption	Icc33			440	1000	mA
1.2 V current consumption	Icc12			150	380	mA
1.8V DDR2 Current	Icc18			50	170	mA

Table 3-6 DC Electrical Characteristics

Vdd=1.8V (DDR2)	Min	Typ	Max
Vdd	1.71	1.8	1.89
VIH	VREF+0.125		Vdd18+0.3
VIL	-0.3		VREF-0.125
VOH	1.42		
VOL			0.28
IOL			
IOH			

Table 3-7 Vdd 1.8V Electrical Characteristics

Vdd=3.3V	Min	Typ	Max
Vdd	2.97V	3.3V	3.63V
VIH	2.0V		Vdd33+0.3
VIL	-0.3		0.8V
VOH	2.4V		
VOL			0.4V
IOL			

IOH			

Table 3-8 Vdd 3.3V Electrical Characteristics

3.7 AC Electrical Characteristics

3.7.1 DDR2 SDRAM Interface

The DDR2 SDRAM interface complies with 200 MHz timing requirements for standard DDR2 SDRAM. The interface drivers are SSTL_18 drivers matching the EIA/JEDEC standard JESD8-15A.

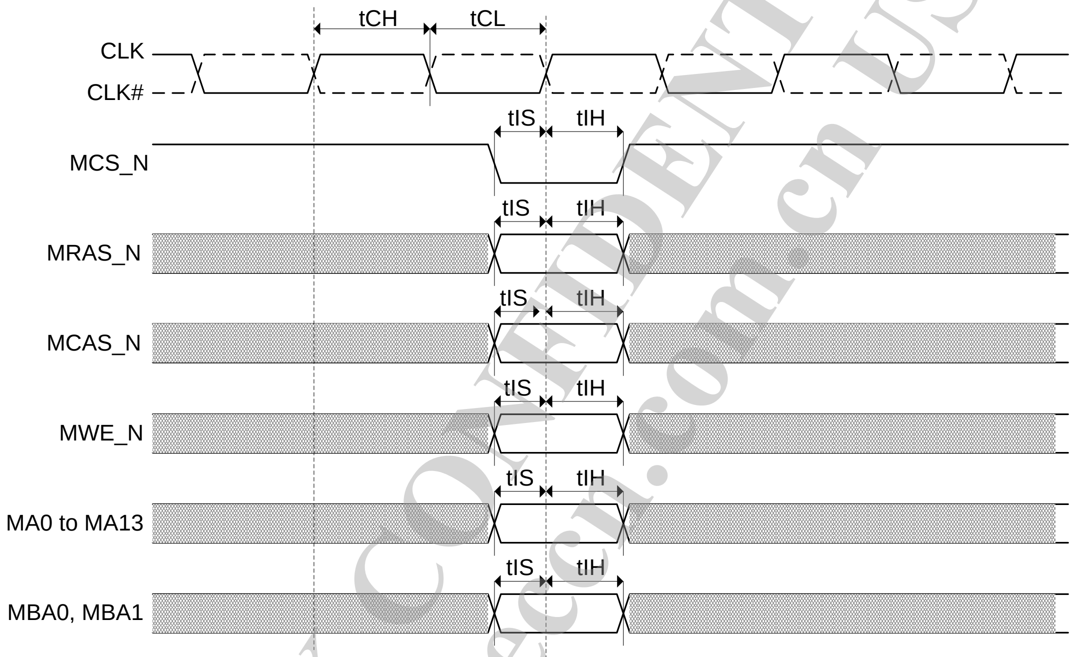


Figure 3-1 DDR2 SDRAM Command

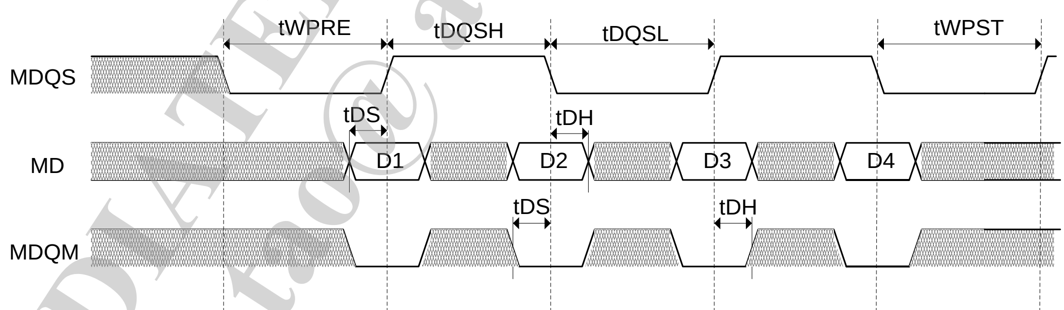


Figure 3-2 DDR2 SDRAM Write data

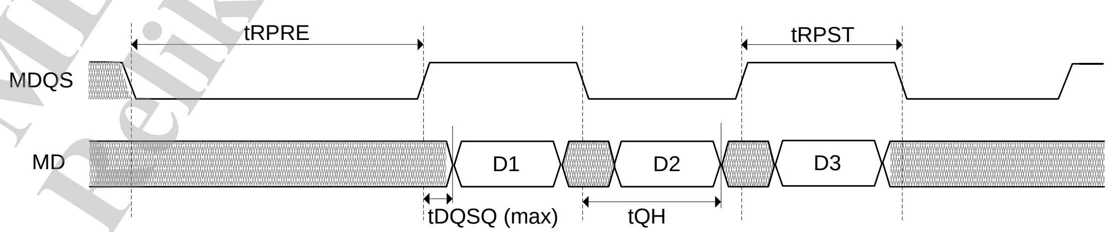


Figure 3-3 DDR2 SDRAM Read data

Symbol	Description	Min	Max	Unit	Remark
tCK(avg)	Clock cycle time	5	-	ns	
tAC	DQ output access time from SDRAM CLK	-0.6	0.6	ns	
tDQSCK	DQS output access time from SDRAM CLK	-0.5	0.5	ns	
tCH	SDRAM CLK high pulse width	0.48	0.52	tCK(avg)	
tCL	SDRAM CLK low pulse width	0.48	0.52	tCK(avg)	
tHP	SDRAM CLK half period	Min(tCH,tCL)	-	ns	
tIS	Address and control input setup time	0.75	-	ns	
tIH	Address and control input hold time	0.75	-	ns	
tDQSQ	Data skew of DQS and associated DQ	-	0.4	ns	
tQH	DQ/DQS output hold time from DQS	tHP-0.5	-	ns	
tRPRE	DQS read preamble	0.9	1.1	tCK	
tRPST	DQS read postamble	0.4	0.6	tCK	
tDQSS	DQS rising edge to CK rising edge	-0.25	0.25	tCK	
tDQSH	DQS input-high pulse width	0.35	-	tCK	
tDQSL	DQS input-low pulse width	0.35	-	tCK	
tDSS	DQS falling edge to SDRAM CLK setup time	0.2	-	tCK	
tDSH	DQS falling edge hold time from SDRAM CLK	0.2	-	tCK	
tWPRE	DQS write preamble	0.35	-	tCK	
tWPST	DQS write postamble	0.4	0.6	tCK	
tDS	DQ and DQM input setup time	*0.4	-	ns	
tDH	DQ and DQM input hold time	*0.4	-	ns	

Table 3-9 DDR2 SDRAM Interface Diagram Key

NOTE: Depends on slew rate of DQS and DQ/DQM for single ended DQS.

3.7.2 SPI Interface

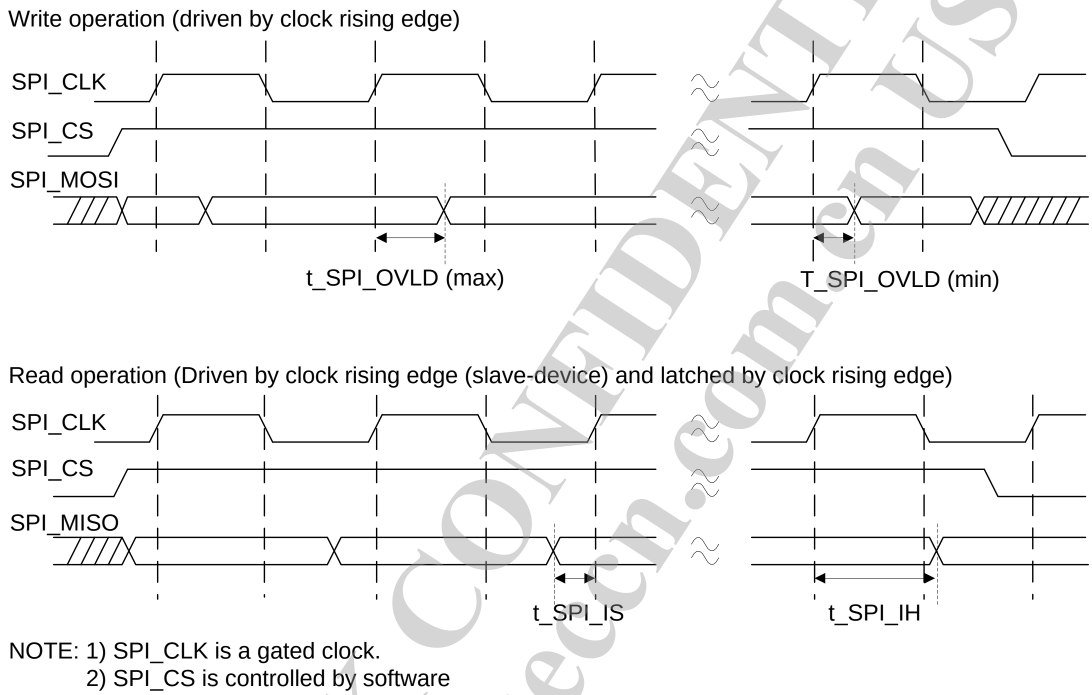


Figure 3-4 SPI Interface

Symbol	Description	Min	Max	Unit	Remark
t_SPI_IS	Setup time for SPI input	6.0	-	ns	
t_SPI_IH	Hold time for SPI input	-1.0	-	ns	
t_SPI_OVLD	SPI_CLK to SPI output valid	-2.0	3.0	ns	output load: 5 pF

Table 3-10 SPI Interface Diagram Key

3.7.3 I²S Interface

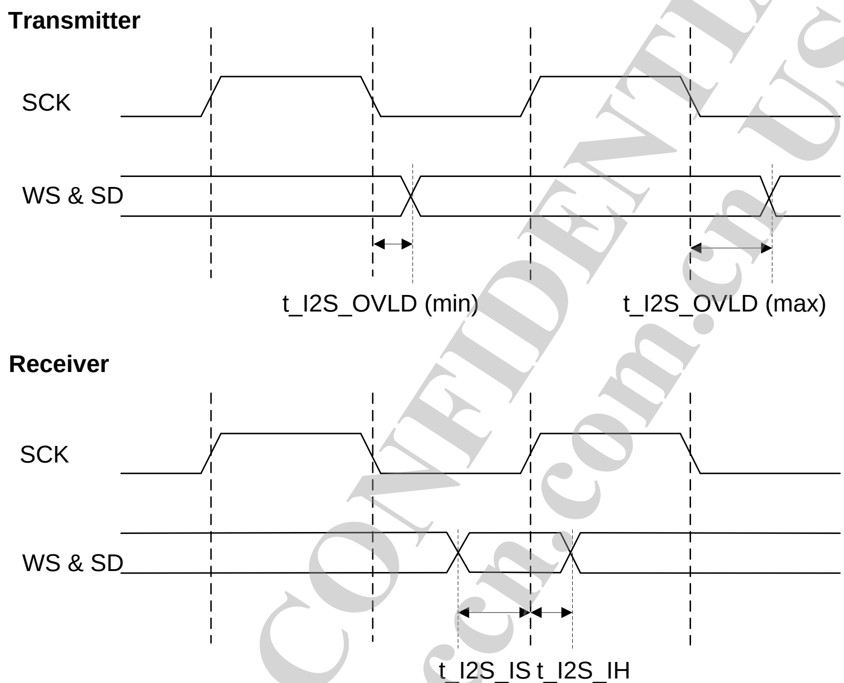


Figure-3-5 I2S Interface

Symbol	Description	Min	Max	Unit	Remark
t_I2S_IS	Setup time for I2S input (data & WS)	3.5	-	ns	
t_I2S_IH	Hold time for I2S input (data & WS)	0.5	-	ns	
t_I2S_OVLD	I2S_CLK to I2S output (data & WS) valid	2.5	10.0	ns	output load: 5 pF

Table 3-11 I2S Interface Diagram Key

3.7.4 PCM Interface

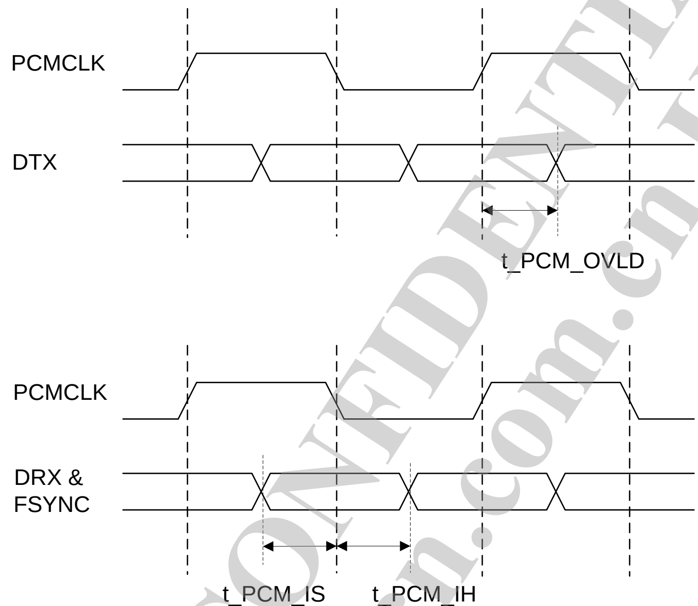
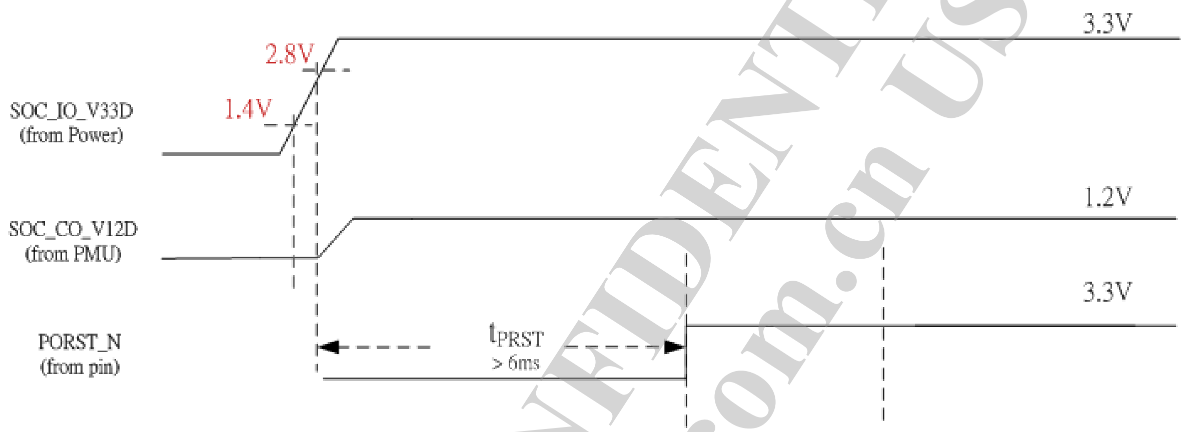


Figure 3-6 PCM Interface

Symbol	Description	Min	Max	Unit	Remark
t_PCM_IS	Setup time for PCM input to PCM_CLK fall	3.0	-	ns	
t_PCM_IH	Hold time for PCM input to PCM_CLK fall	1.0	-	ns	
t_PCM_OVLD	PCM_CLK rise to PCM output valid	10.0	35.0	ns	output load: 5 pF

Table 3-12 PCM Interface Diagram Key

3.7.5 Power On Sequence



Symbol	Description	Min	Max	Unit
t_{PRST}	External Power-on Reset Period	6.0		ms

Figure 3-7 Power ON Sequence

Table 3-13 Power ON Sequence Diagram Key

3.8 **Package Physical Dimensions**

3.8.1 DR-QFN (12 mm x 12 mm) 156 pins

3.8.1.1 Top View

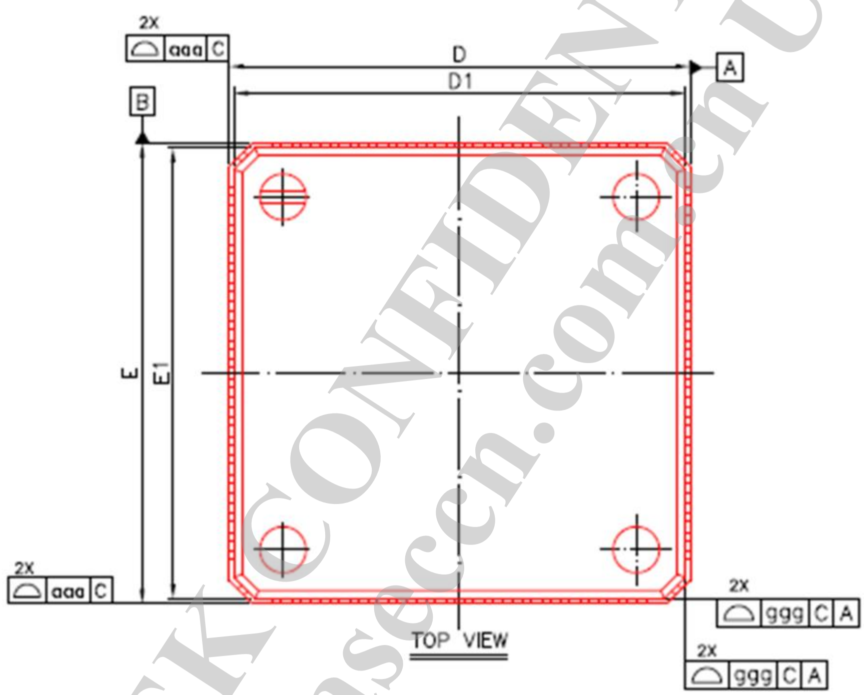


Figure 3-8 Top View

3.8.1.2 Side View

3.8.1.3 "B" Expanded

Figure 3-9 Side View

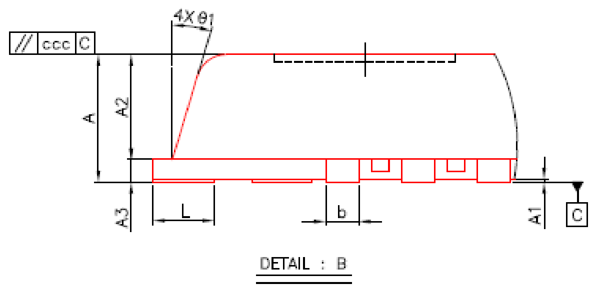


Figure 3-10 "B" Expanded

3.8.1.4 Bottom View

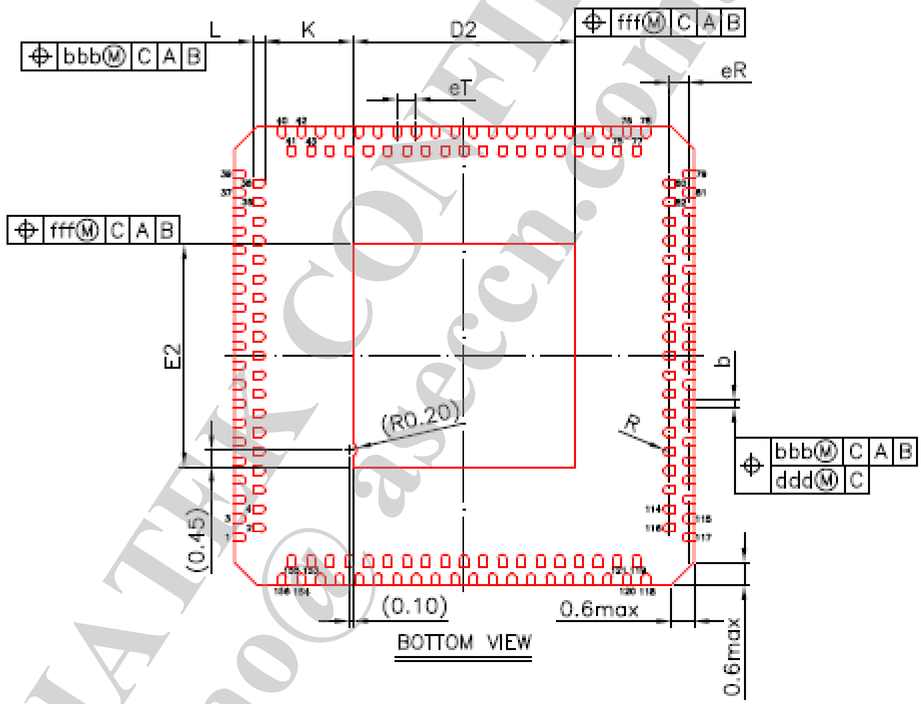
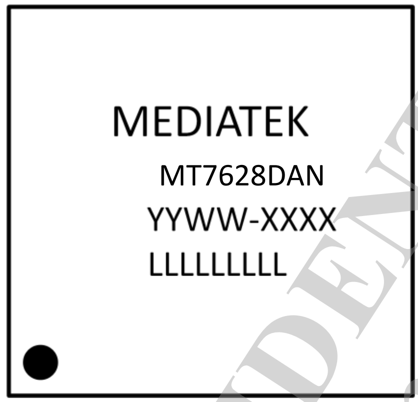


Figure 3-11 Bottom View

3.8.1.5 Package Diagram Key

Item	SYMBOL	MIN.	NOM.	MAX.
TOTAL THICKNESS	A	0.80	0.85	0.90
LEAD STAND OFF.	A1	0.00	0.02	0.05
MOLD THICKNESS	A2	0.65	0.70	0.75
L/F THICKNESS	A3	0.15 REF.		
LEAD WIDTH	b	0.18	0.22	0.30
PACKAGE SIZE	D	11.90	12.00	12.10
	E			
Mold Edge size	D1	11.75 BSC		
	E1	11.75 BSC		
E-PAD size	D2	5.70	5.80	5.90
	E2	5.70	5.80	5.90
LEAD LENGTH	L	0.20	0.30	0.40
LEAD PITCH (BSC.)	eT	0.50 BSC		
LEAD PITCH (BSC.)	eR	0.50 BSC		
ANGLE	θ1	5°	---	15°
LEAD ARC	R	0.09	---	0.14
Lead to E-PAD Toler-ance	K	0.20	---	---
PKG EDGE TOLER-ANCE	aaa	0.10		
PACKAGE PROFILE OF A SURFACE	bbb	0.10		
LEAD PROFILE OF A SURFACE	ccc	0.10		
LEAD POSITION	ddd	0.05		
LEAD PROFILE OF A SURFACE	eee	0.08		
EPAD POSTION	fff	0.10		
Mold edge OF A & C SURFACE	ggg	0.20		

3.8.2 MT7628 DN marking



YYWW: Date code
LLLLLLLLL : Lot number
"." : Pin #1 dot

Figure 3-12 MT7628DAN top marking

3.8.3 Reflow profile guideline

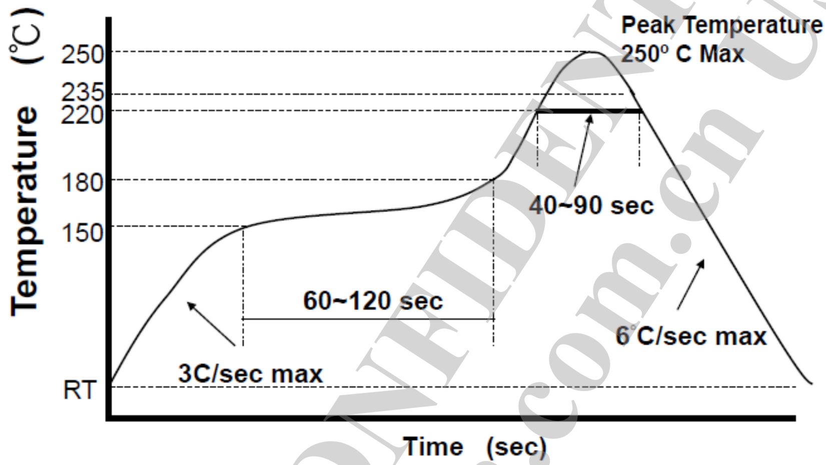


Figure 3-13 Reflow profile for MT7628

Notes;
1. Reflow profile guideline is designed for SnAgCulead-free solder paste.
2. Reflow temperature is defined at the solder ball of package/or the lead of package.
3. MTK would recommend customer following the solder paste vendor's guideline to design a profile appropriate your line and products.
4. Appropriate N2 atmosphere is recommended since it would widen the process window and mitigate the risk for having solder open issues.

4. Register

4.1 **Nomenclature**

The following nomenclature is used for register types:

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

4.2 System Control

4.2.1 Features

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

4.2.2 Block Diagram

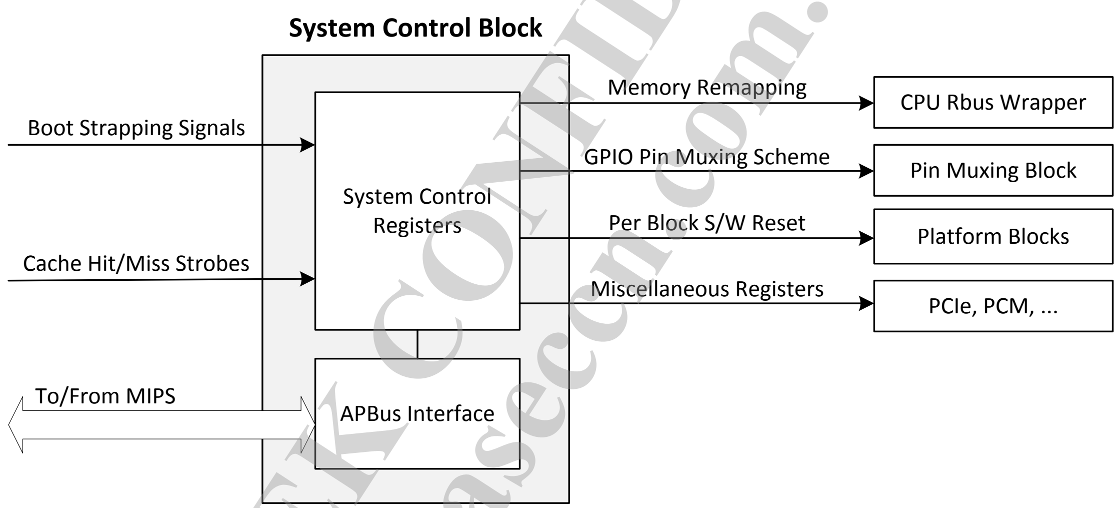


Figure 4-1 System Control Block Diagram

4.2.3 Registers

SYSCTL Changes LOG

Revision	Date	Author	Change Log
0.1	2013/10/3	PeterCT Wu	Initial for MT7628
0.2	2014/4/28	PeterCT Wu	MT7628 E2
0.3	2014/5/21	Morrie Lin	Add 4-bit SDXC on Router mode

Module name: SYSCTL Base address: (+10000000h)

Address	Name	Width	Register Function
10000000	CHIPID0_3	32	CHIP ID ASCII Character 0-3
10000004	CHIPID4_7	32	CHIP ID ASCII Character 4-7
10000008	EE_CFG	32	E-Fuse Configuration
1000000C	CHIP_REV_ID	32	Chip Revision Identification
10000010	SYSCFG0	32	System Configuration Register 0
10000014	SYSCFG1	32	System Configuration Register 1
10000018	TESTSTAT	32	Firmware Test Status
1000001C	TESTSTAT2	32	Firmware Test Status 2
10000028	ROM_STATUS	32	Andes ROM Status
1000002C	CLKCFG0	32	Clock Configuration Register 0
10000030	CLKCFG1	32	Clock Configuration Register 1
10000034	RSTCTL	32	Reset Control Register
10000038	RSTSTAT	32	Reset Status Register
1000003C	AGPIO_CFG	32	Analog GPIO Configuration
10000040	N9_GPIO_INT	32	Andes GPIO Interrupt
10000044	N9_GPIO_MASK	32	Andes GPIO Mask
10000060	GPIO1_MODE	32	GPIO1 purpose selection
10000064	GPIO2_MODE	32	GPIO2 purpose selection
10000068	MEMO1	32	Memory1
1000006C	MEMO2	32	Memory2
10000070	EXT_MEMO1	32	Extend Application #1
10000074	EXT_MEMO2	32	Extend Application #2
10000078	EXT_MEMO3	32	Extend Application #3
1000007C	EXT_MEMO4	32	Extend Application #4

10000000 CHIPID0_3 CHIP ID ASCII Character 0-3 3637544D

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

Type	RO								RO							
Reset	0	1	0	1	0	1	0	0	0	1	0	0	1	1	0	1

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

10000004 CHIPID4_7 CHIP ID ASCII Character 4-7 20203832

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

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

10000008 EE_CFG E-Fuse Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EE_CFG1															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EE_CFG0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	EE_CFG1	**E-Fuse Configuration 1**
15:0	EE_CFG0	**E-Fuse Configuration 0**

1000000C CHIP_REV_ID Chip Revision Identification 00010102

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

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

Bit(s)	Name	Description
16	PKG_ID	**Package ID** 0: DRQFN10x10-110 1: DRQFN12x12-156
11:8	VER_ID	**Chip Version ID**
3:0	ECO_ID	**Chip ECO ID**

10000010 SYSCFG0 System Configuration Register 0 00000010

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**TEST_CODE**											**BS_SHADOW[8:4]**				
Type	RW											RO				
Reset	0	0	0	0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**BS_SHADOW[3:0]**							**DBG_JTAG_MODE**	**TEST_MODE_1**	**XTAL_FREQ_SEL**	**EXT_BG**	**TEST_MODE_0**	**CHIP_MODE**			**DRAM_TYPE**
Type	RO							RO	RO	RO	RO	RO	RO			RO
Reset	0	0	0	0				1	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	TEST_CODE	**Default value is from bootstrap and can be modified by software.**
20:12	BS_SHADOW	**BS shadow register for last boot-up value (by manual boot-strap SYSCFG1.PULL_EN)** Displays a backup copy of the last bootup value
8	DBG_JTAG_MODE	**JTAG for MIPS and Andes** 1: Normal Boot-up 0: JTAG mode(MIPS & Andes)
7	TEST_MODE_1	**Test Mode[1:0]**
6	XTAL_FREQ_SEL	**XTAL Frequency Selection** 0: 25MHz DIP 1: 40MHz SMD (3225)
5	EXT_BG	**External BG Clock** 0: BG clock from PMU 1: BG clock from the external pin
4	TEST_MODE_0	**Test Mode[1:0]** 0: SUTIF 1: 3-wire SPI
3:1	CHIP_MODE	Chip Mode A vector to set chip function/test/debug modes in non-test/debug operation. For more information see the Bootstrapping Pins Description in the datasheet

Bit(s)	Name	Description
		for this chip. 000: Boot from PLL (boot from SPI 3-Byte ADR) 001: Boot from PLL (boot from SPI 4-Byte ADR) 010: Boot from XTAL (boot from SPI 3-Byte ADR) 011: Boot from XTAL (boot from SPI 4-Byte ADR) 100: SCAN mode 101: IDDQ mode 110: Power-On mode 111: UTIF test mode
0	DRAM_TYPE	**DDR type** [note] This DDR attribute is not valid for KN package.. (7628KN has DDR1 KGD) 0: DDR2 1: DDR1

10000014 SYSCFG1 System Configuration Register 1 00000000

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

Bit(s)	Name	Description
16	PULL_EN	**Internal Manual Boot-Strap** 1: enable 0: disable

10000018 TESTSTAT Firmware Test Status 00000000

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

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

1000001C **TESTSTAT2** **Firmware Test Status 2** **00000000**

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

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

10000028 **ROM_STATUS** **Andes ROM Status** **00000000**

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

Bit(s)	Name	Description
7:0	STATUS	**Andes ROM Status** 0: Power-on default 1: ROM initialization done 2: Wifi driver loaded

1000002C **CLKCFG0** **Clock Configuration Register 0** **00201000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			OSC_1US_DIV							INT_CLK_FDIV						INT_CLK_FFRAC[4:4]
Type			RW							RW						RW
Reset			0	0	0	0	0	0		0	1	0	0	0		0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	INT_CLK_FFRAC[3:0]				REFCLK0_RATE				DIS_N9		PCIE_EXT_125M	PERI_CLK_SEL	DIS_BBP_SLEEP	EN_BBP_CLK	CPU_FRM_BBP	CPU_FRM_XTAL
Type	RW				RW				RW		RW	RW	RW	RW	RW	RW
Reset	0	0	0	1	0	0	0		0		0	0	0	0	0	0

Bit(s)	Name	Description
29:24	OSC_1US_DIV	**Oscillator 1 usec Divider** Sets the maximum for the reference clock counter for either a 20 MHz or 40 MHz external XTAL input. The count increments each 1usec (indicating 1 MHz), up to the maximum, before resetting to zero. This counts the frequency of an external XTAL. This count is used to output a 32 KHz frequency to the REFCLK0 pin. 0: Automatically generates a 1 usec system tick regardless of whether XTAL frequency is 20 MHz or 40 MHz. 39: Default value for an external 40 MHz XTAL. 19: Default value for an external 20 MHz XTAL. Others: Manual mode for tick generation.
22:18	INT_CLK_FDIV	**Internal Clock Frequency Divider for I2S/PCM** The frequency divider used to generate the Fraction-N clock frequency. Valid values range from 1 to 31. Fraction-N clock frequency = (INT_CLK_FFRAC/INT_CLK_FDIV)*PLL_FREQ
16:12	INT_CLK_FFRAC	**Internal Clock Fraction-N Frequency for I2S/PCM** A parameter used in conjunction with INT_CLK_FDIV to generate the Fraction-N clock frequency. Valid values range from 0 to 31. Fraction-N clock Frequency = (INT_CLK_FFRAC/INT_CLK_FDIV)*PLL_FREQ
11:9	REFCLK0_RATE	**Output clock rate of reference Clock 0** 7: CPUPLL Clock/8 6: Off 5: Internal Fraction-N_CLK/2 (I2S/PCM) 4: 48 MHz 3: 40 MHz 2: 25 MHz 1: 12 MHz 0: Xtal clock(25/40 MHz by boot strap)
7	DIS_N9	**Pause Andes Execution** [Note] This bit is initialized by HW STRAP and can be changed by SW afterwards. 1: Enable 0: default
5	PCIE_EXT_125M	**PCIe 125MHZ Clock Source** 1: Ext. 125MHz Source (EPHY) 0: PCIe PHY 125M
4	PERI_CLK_SEL	**Peripheral Clock Source Select** 1: XTAL input 0: 40 MHz from BBP 480 MHz divided by 12
3	DIS_BBP_SLEEP	**BBPPLL Sleep Mode Control** 1: Disable BBPPLL entering SLEEP mode 0: BBPPLL SLEEP mode
2	EN_BBP_CLK	**BBPPLL 480MHz Clock** 1: BBPPLL Clock Enable 0: BBPPLL Clock Disable
1	CPU_FRM_BBP	**CPU clock from BBPPLL** 1: 480MHz BBPPLL 0: 580MHz CPUPLL
0	CPU_FRM_XTAL	**CPU clock from XTAL** [Note] This bit is initialized by HW STRAP and can be changed by SW afterwards.

Bit(s)	Name	Description
		1: XTAL input 0: CPUPLL

10000030 CLKCFG1 Clock Configuration Register 1 F69F7F00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PWM_CLK_EN	SDXC_CLK_EN	CRYPTO_CLK_EN	MIPSC_CLK_EN		PCIE_CLK_EN	UPHY_CLK_EN		ETH_CLK_EN			UART2_CLK_EN	UART1_CLK_EN	SPI_CLK_EN	I2S_CLK_EN	I2C_CLK_EN
Type	RW	RW	RW	RW		RW	RW		RW			RW	RW	RW	RW	RW
Reset	1	1	1	1		1	1		1			1	1	1	1	1
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name		GDMA_CLK_EN	PIO_CLK_EN	UART0_CLK_EN	PCM_CLK_EN	MC_CLK_EN	INT_CLK_EN	TIMER_CLK_EN								
Type		RW	RW	RW	RW	RW	RW	RW								
Reset		1	1	1	1	1	1	1								

Bit(s)	Name	Description
31	PWM_CLK_EN	**PWM clock control** 1: Clock Enable 0: Clock Disable
30	SDXC_CLK_EN	**SDXC clock control** 1: Clock Enable 0: Clock Disable
29	CRYPTO_CLK_EN	**AUX system tick counter clock control** 1: Clock Enable 0: Clock Disable
28	MIPSC_CLK_EN	**MIPS Counter clock control** 1: Clock Enable 0: Clock Disable
26	PCIE_CLK_EN	**PCIE2 clock control** 1: Clock Enable 0: Clock Disable
25	UPHY_CLK_EN	**UPHY clock control** 1: Clock Enable 0: Clock Disable
23	ETH_CLK_EN	**ETH clock control** 1: Clock Enable 0: Clock Disable
20	UART2_CLK_EN	**UART2 clock control** 1: Clock Enable 0: Clock Disable
19	UART1_CLK_EN	**UART1 clock control** 1: Clock Enable 0: Clock Disable

Bit(s)	Name	Description
18	SPI_CLK_EN	**SPI clock control** 1: Clock Enable 0: Clock Disable
17	I2S_CLK_EN	**I2S clock control** 1: Clock Enable 0: Clock Disable
16	I2C_CLK_EN	**I2C clock control** 1: Clock Enable 0: Clock Disable
14	GDMA_CLK_EN	**GDMA clock control** 1: Clock Enable 0: Clock Disable
13	PIO_CLK_EN	**PIO clock control** 1: Clock Enable 0: Clock Disable
12	UART0_CLK_EN	**UART0 clock control** 1: Clock Enable 0: Clock Disable
11	PCM_CLK_EN	**PCM clock control** 1: Clock Enable 0: Clock Disable
10	MC_CLK_EN	**MC clock control** 1: Clock Enable 0: Clock Disable
9	INT_CLK_EN	**INT clock control** 1: Clock Enable 0: Clock Disable
8	TIMER_CLK_EN	**TIMER clock control** 1: Clock Enable 0: Clock Disable

10000034 **RSTCTL** **Reset Control Register** **04000400**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PWM_RST	SDXC_RST	CRYPTO_RST	AUX_STCK_RST		PCIE_RST		EPHY_RST	ETH_RST	UHST_RST		UART2_RST	UART1_RST	SPI_RST	I2S_RST	I2C_RST
Type	RW	RW	RW	RW		RW		RW	RW	RW		RW	RW	RW	RW	RW
Reset	0	0	0	0		1		0	0	0		0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GDMA_RST	PIO_RST	UART0_RST	PCM_RST	MC_RST	INT_RST	TIMER_RST			HIF_RST	WIFI_RST	SPIS_RST			SYS_RST
Type		RW	RW	RW	RW	RW	RW	RW			RW	RW	RW			W1C
Reset		0	0	0	0	1	0	0			0	0	0			0

Bit(s)	Name	Description
31	PWM_RST	**PWM reset control** 1: Reset Assert 0: Reset Deassert
30	SDXC_RST	**SDXC reset control** 1: Reset Assert 0: Reset Deassert
29	CRYPTO_RST	**Crypto engine reset control** 1: Reset Assert 0: Reset Deassert
28	AUX_STCK_RST	**AUX system tick counter clock control** 1: Reset Assert 0: Reset Deassert
26	PCIE_RST	**PCIE reset control** 1: Reset Assert 0: Reset Deassert
24	EPHY_RST	**EPHY reset control** 1: Reset Assert 0: Reset Deassert
23	ETH_RST	**ETH reset control** 1: Reset Assert 0: Reset Deassert
22	UHST_RST	**USB PHY reset control** 1: Reset Assert 0: Reset Deassert
20	UART2_RST	**UART2 reset control** 1: Reset Assert 0: Reset Deassert
19	UART1_RST	**UART1 reset control** 1: Reset Assert 0: Reset Deassert
18	SPI_RST	**SPI reset control** 1: Reset Assert 0: Reset Deassert
17	I2S_RST	**I2S reset control** 1: Reset Assert 0: Reset Deassert
16	I2C_RST	**I2C reset control** 1: Reset Assert 0: Reset Deassert
14	GDMA_RST	**GDMA reset control** 1: Reset Assert 0: Reset Deassert
13	PIO_RST	**PIO reset control** 1: Reset Assert 0: Reset Deassert
12	UART0_RST	**UART0 reset control** 1: Reset Assert 0: Reset Deassert
11	PCM_RST	**PCM reset control**

Bit(s)	Name	Description
		1: Reset Assert 0: Reset Deassert
10	MC_RST	**MC reset control** 1: Reset Assert 0: Reset Deassert
9	INT_RST	**INT reset control** 1: Reset Assert 0: Reset Deassert
8	TIMER_RST	**TIMER reset control** 1: Reset Assert 0: Reset Deassert
5	HIF_RST	**WIFI HIF reset control** [Note] WPDMA reset control 1: Reset Assert 0: Reset Deassert
4	WIFI_RST	**WIFI reset control** [Note] This bit will reset Andes and initialize XTAL and BBPPLL again, MIPS must carefully use it. 1: Reset Assert 0: Reset Deassert
3	SPIS_RST	**SPI Slave control** 1: Reset Assert 0: Reset Deassert
0	SYS_RST	**Whole System Reset Control** [Note] Except for power-on CR, this bit reset the whole system include itself. 1: Whole System Reset 0: NA

10000038 **RSTSTAT** **Reset Status Register** **C0030000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WDT2SYSRST_EN	WDT2RSTO_EN	WDTRSTPD													
Type	RW	RW	RW													
Reset	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							WDRST_TON9_EN	N9_WDRST_EN					N9SYSRST	SWSYSRST	WDRST	
Type							RW	RW					W1C	W1C	W1C	
Reset							0	0					0	0	0	

Bit(s)	Name	Description
31	WDT2SYSRST_EN	**WDT reset apply to System Reset**

Bit(s)	Name	Description
		Enables watchdog timeout to trigger a system reset. 1: Enable 0: Disable
30	WDT2RSTO_EN	**WDT reset apply to watch dog reset pin out.** 1: Enable 0: Disable
29:16	WDTRSTPD	**Watchdog Reset Output Low Period** Controls the WDT reset output low period. For example: If the pin share mode was set correctly and WDT2RSTO_EN=1, When WDTRSTPD= 0, you can see duration of 1 usec low on the WDT reset output pin. When WDTRSTPD= 3, you can see duration of 4 usec low on the WDT reset output pin. (unit: 1 usec)
9	WDRST_TON9_EN	**MIPS software reset or watch-dog reset apply to N9 subsys.** When this bit is set, MIPS can reset N9 or N9 is reset when MISP watch-dog reset happen. 0: disable 1: Enable
8	N9_WDRST_EN	**N9 watch-dog reset applies to MIPS subsys.** When N9 WDRST happens, N9 will also reset MIPS system. 0: disable 1: Enable
3	N9SYSRST	**N9 watch-dog reset occurred** This bit will be set if N9 wifisys is reset by its watch-dog mechanism. Writing a '1' will clear this bit. Writing a '0' has not effect. NOTE: This register is reset only by a power on reset. 0: Has no effect. 1: Clears this bit.
2	SWSYSRST	**Software system reset occurred** This bit will be set if software reset the chip by writing to the RSTSYS bit in RSTCTL. Writing a '1' will clear this bit. Writing a '0' has not effect. NOTE: This register is reset only by a power on reset. 0: Has no effect. 1: Clears this bit.
1	WDRST	**Watchdog reset occurred** This bit will be set if the watchdog timer reset the chip. Writing a '1' will clear this bit. Writing a '0' has not effect. NOTE: This register is reset only by power-on reset. 0: Has no effect. 1: Clears this bit.

1000003C **AGPIO_CFG** **Analog GPIO Configuration** **001F001F**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												EPHY_GPIO_AIO_EN				EPHY_P0_DIS
Type												RW				RW
Reset												1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name				RF_OLT_MODE			EINT_SEL	WLED_OD_EN				REF_CLKO_AIO_EN	I2S_CLK_AIO_EN	I2S_WS_AIO_EN	I2S_SDO_AIO_EN	I2S_SDI_AIO_EN
Type				RW			RW	RW				RW	RW	RW	RW	RW
Reset				0			0	0				1	1	1	1	1

Bit(s)	Name	Description
20:17	EPHY_GPIO_AIO_EN	**EPHY P1 ~ P4 digital PAD selection (P1 ~ P4 Disable)** (note: When any bit of bit[20:17] is set to 1, P1 ~ P4 will be swtiched to digital PADs together.) 0: Analog PAD 1: Digital PAD
16	EPHY_P0_DIS	**EPHY P0 Disable** 0: Enable 1: Disable
12	RF_OLT_MODE	**Enable RF OLT mode** 0: DIsable 1: Enable
9	EINT_SEL	**Andes EINT Source** 0: from W_UTIF 1: from GPIO [23:20]
8	WLED_OD_EN	**WLED Open-Drain** 0: DIsable 1: Open-Drain
4	REF_CLKO_AIO_EN	**REF Clock Output PAD Selection** 0: Analog PAD 1: Digital PAD
3	I2S_CLK_AIO_EN	**I2S Clock PAD Selection** 0: Analog PAD 1: Digital PAD
2	I2S_WS_AIO_EN	**I2S WS PAD Selection** 0: Analog PAD 1: Digital PAD
1	I2S_SDO_AIO_EN	**I2S CSDO PAD Selection** 0: Analog PAD 1: Digital PAD
0	I2S_SDI_AIO_EN	**I2S SDI PAD Selection** 0: Analog PAD 1: Digital PAD

10000040 **N9_GPIO_INT** **Andes GPIO Interrupt** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																GPIO_INT[16:16]
Type																W1

																C
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**GPIO_INT[15:0]**															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16:0	GPIO_INT	**Andes GPIO INT**

10000044 N9_GPIO_MASK Andes GPIO Mask 0001FFFF

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

Bit(s)	Name	Description
16:0	GPIO_MASK	**Andes GPIO MASK**

10000060 GPIO1_MODE GPIO1 purpose selection 54050404

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**PWM1_MODE**		**PWM0_MODE**		**UART2_MODE**		**UART1_MODE**				**I2C_MODE**			**REFCLK_MODE**		**PERST_MODE**
Type	RW		RW		RW		RW				RW			RW		RW
Reset	0	1	0	1	0	1	0	0			0	0		1		1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		**WDT_MODE**		**SPI_MODE**	**SD_MODE**		**UART0_MODE**		**I2S_MODE**		**SPI_CS1_MODE**		**SPIS_MODE**		**GPIO_MODE**	
Type		RW		RW	RW		RW		RW		RW		RW		RW	
Reset		0		0	0	1	0	0	0	0	0	0	0	1	0	0

Bit(s)	Name	Description
31:30	PWM1_MODE	**PWM1 GPIO mode** 3: SDXC D6 2: UTIF[5] 1: GPIO 0: PWM ch1

Bit(s)	Name	Description
29:28	PWM0_MODE	**PWM0 GPIO mode** 3: SDXC D7 2: UTIF[4] 1: GPIO 0: PWM ch0
27:26	UART2_MODE	**UART2 GPIO mode** 3: SDXC D5/D4 2: PWM ch2/ch3 1: GPIO 0: UART-Lite #2
25:24	UART1_MODE	**UART1 GPIO mode** 3: SW_R, SW_T 2: PWM ch0/ch1 1: GPIO 0: UART-Lite #1
21:20	I2C_MODE	**I2C GPIO mode** 2: S-UART (debug) 1: GPIO 0: I2C
18	REFCLK_MODE	**REFCLK GPIO mode** 1: GPIO 0: REFCLK (12M)
16	PERST_MODE	**PCIe RESET GPIO mode** 1: GPIO 0: PCIe reset
15	ESD_MODE	**SDXC Router mode** 1: SDXC from I2S/I2C/GPIO0/UART1 pins 0: SDXC from EPHY pins
14	WDT_MODE	**Watch dog timeout GPIO mode** 1: GPIO 0: Watch dog
12	SPI_MODE	**SPI GPIO mode** 1: GPIO 0: SPI
11:10	SD_MODE	**SDXC GPIO mode** 3: Andes JTAG 2: UTIF[17:10] 1: GPIO 0: SDXC
9:8	UART0_MODE	**UART0 GPIO mode** 1: GPIO 0: UART-Lite #0
7:6	I2S_MODE	**I2S GPIO mode** 3: ANTSEL[5:2] 2: PCM 1: GPIO 0: I2S
5:4	SPI_CS1_MODE	**SPI CS1 GPIO mode** 2: REFCLK 1: GPIO 0: SPI CS1

Bit(s)	Name	Description
3:2	SPIS_MODE	**SPI Slave GPIO mode** 3: PWM CH0/1 and UART2 2: UTIF[3:0] 1: GPIO 0: SPI Slave
1:0	GPIO_MODE	**GPIO mode** 3: PCIe Reset 2: REFCLK (12M) 1: GPIO 0: GPIO

10000064 GPIO2_MODE GPIO2 purpose selection 05550555

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					P4_LED_KN_MODE		P3_LED_KN_MODE		P2_LED_KN_MODE		P1_LED_KN_MODE		P0_LED_KN_MODE		WLED_KN_MODE	
Type					RW		RW		RW		RW		RW		RW	
Reset					0	1	0	1	0	1	0	1	0	1	0	1
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name					P4_LED_AN_MODE		P3_LED_AN_MODE		P2_LED_AN_MODE		P1_LED_AN_MODE		P0_LED_AN_MODE		WLED_AN_MODE	
Type					RW		RW		RW		RW		RW		RW	
Reset					0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
27:26	P4_LED_KN_MODE	**EPHY P4 LED GPIO mode** [Note] Only valid for MT7628KN. 3: JTAG (JTRST_N) 2: UTIF[6] 1: GPIO 0: EPHY P4 LED
25:24	P3_LED_KN_MODE	**EPHY P3 LED GPIO mode** [Note] Only valid for MT7628KN. 3: JTAG (JTCLK) 2: UTIF[7] 1: GPIO 0: EPHY P3 LED
23:22	P2_LED_KN_MODE	**EPHY P2 LED GPIO mode** [Note] Only valid for MT7628KN. 3: JTAG (JTMS) 2: UTIF[8] 1: GPIO 0: EPHY P2 LED
21:20	P1_LED_KN_MODE	**EPHY P1 LED GPIO mode** [Note] Only valid for MT7628KN. 3: JTAG (JTDI) 2: UTIF[9] 1: GPIO 0: EPHY P1 LED
19:18	P0_LED_KN_MODE	**EPHY P0 LED GPIO mode** [Note] Only valid for MT7628KN. 3: JTAG(JTDO)

Bit(s)	Name	Description
		2: Reserved 1: GPIO 0: EPHY P0 LED
17:16	WLED_KN_MODE	**WLED GPIO mode** [Note] Only valid for MT7628KN. 3: Reserved 2: Reserved 1: GPIO 0: WLED
11:10	P4_LED_AN_MODE	**EPHY P4 LED GPIO mode** [Note] Only valid for MT7628AN. 3: JTAG (JTRST_N) 2: UTIF[6] 1: GPIO 0: EPHY P4 LED
9:8	P3_LED_AN_MODE	**EPHY P3 LED GPIO mode** [Note] Only valid for MT7628AN. 3: JTAG (JTCLK) 2: UTIF[7] 1: GPIO 0: EPHY P3 LED
7:6	P2_LED_AN_MODE	**EPHY P2 LED GPIO mode** [Note] Only valid for MT7628AN. 3: JTAG (JTMS) 2: UTIF[8] 1: GPIO 0: EPHY P2 LED
5:4	P1_LED_AN_MODE	**EPHY P1 LED GPIO mode** [Note] Only valid for MT7628AN. 3: JTAG (JTDI) 2: UTIF[9] 1: GPIO 0: EPHY P1 LED
3:2	P0_LED_AN_MODE	**EPHY P0 LED GPIO mode** [Note] Only valid for MT7628AN. 3: JTAG(JTDO) 2: Reserved 1: GPIO 0: EPHY P0 LED
1:0	WLED_AN_MODE	**WLED GPIO mode** [Note] Only valid for MT7628AN. 3: Reserved 2: Reserved 1: GPIO 0: WLED

10000068 MEMO1 Memory1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMO1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMO1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000006C **MEMO2** **Memory2** **00000000**

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

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

10000070 **EXT_MEMO1** **Extend Application #1** **00000000**

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

Bit(s)	Name	Description
31:0	MEMO1	**Extend Application #1**

10000074 **EXT_MEMO2** **Extend Application #2** **00000000**

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

Bit(s)	Name	Description

Bit(s)	Name	Description
31:0	MEMO2	**Extend Application #2**

10000078 **EXT_MEMO3** **Extend Application #3** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMO3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMO3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEMO3	**Extend Application #3**

1000007C **EXT_MEMO4** **Extend Application #4** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMO4[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMO4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEMO4	**Extend Application #4**

4.3 **Timer**

4.3.1 Features

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

4.3.2 Block Diagram

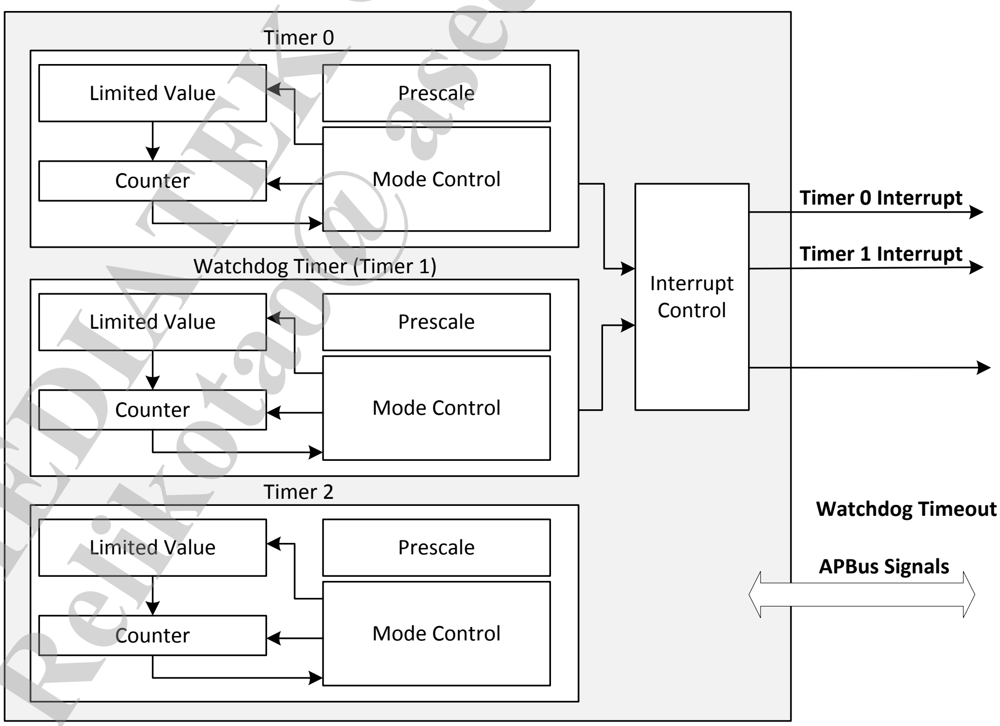


Figure 4-2 Timer Block Diagram

4.3.3 Registers

TIMER Changes LOG

Revision	Date	Author	Change Log
0.1	2012/8/24	Leon Chung	Initialization
0.2	2013/12/10	Rick Ho	1. Modify T0CTL_REG Bit[4] to WO and add Bit[3] RO 2. Modify WDTCTL_REG Bit[4] to WO and add Bit[3] RO 3. Modify T1CTL_REG Bit[4] to WO and add Bit[3] RO

Module name: TIMER Base address: (+10000100h)

Address	Name	Width	Register Function
10000100	TGLB_REG	32	RISC Global Control Register
10000110	T0CTL_REG	32	RISC Timer 0 Control Register
10000114	T0LMT_REG	32	RISC Timer 0 Limit Register
10000118	T0_REG	32	RISC Timer 0 Register
10000120	WDTCTL_REG	32	Watch Dog Timer Control Register
10000124	WDTLMT_REG	32	Watch Dog Timer Limit Register
10000128	WDT_REG	32	Watch Dog Timer Register
10000130	T1CTL_REG	32	RISC Timer 1 Control Register
10000134	T1LMT_REG	32	RISC Timer 1 Limit Register
10000138	T1_REG	32	RISC Timer 1 Register

10000100 TGLB_REG RISC Global Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV1[20:5]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV1[4:0]					T1RST	WDTRST	T0RST	RESV0					T1INT	WDTINT	T0INT
Type	RO					W1C	W1C	W1C	RO					W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:11	RESV1	**Reserved**
10	T1RST	**Timer 1 reset** 1: to reset timer 1 to T1LMT value
9	WDTRST	**Watch dog timer reset**

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

10000110 **T0CTL_REG** **RISC Timer 0 Control Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	T0PRES															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2								T0EN	RESV1		T0AL	T0AL_STATUS	RESV0		
Type	RO								RW	RO		WO	RO	RO		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	T0PRES	**Timer 0 count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	T0EN	**Timer 0 count down enable**
6:5	RESV1	**Reserved**
4	T0AL	**Timer 0 auto load enable** 1: Enable 0: Disable
3	T0AL_STATUS	**Timer 0 auto load enable status** 1: Enable 0: Disable
2:0	RESV0	**Reserved**

10000114 **T0LMT_REG** **RISC Timer 0 Limit Register** **00000000**

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

Bit(s)	Name	Description

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

10000118 **T0_REG** **RISC Timer 0 Register** **0000FFFF**

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

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

10000120 **WDTCTL_REG** **Watch Dog Timer Control Register** **00000000**

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

Bit(s)	Name	Description
31:16	WDTPRES	**Watch dog timer count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	WDTEN	**Watch dog timer count down enable**
6:5	RESV1	**Reserved**
4	WDTAL	**Watch dog timer auto load enable** 1: Enable 0: Disable
3	WDTAL_STATUS	**Watch dog timer auto load enable status** 1: Enable 0: Disable
2:0	RESV0	**Reserved**

10000124 WDTLMT_REG Watch Dog Timer Limit Register 00000000

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

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

10000128 WDT_REG Watch Dog Timer Register 0000FFFF

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

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

10000130 T1CTL_REG RISC Timer 1 Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	T1PRES															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2								T1EN	RESV1		T1AL	T1AL_STATUS	RESV0		
Type	RO								RW	RO		WO	RO	RO		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31:16	T1PRES	**Timer 1 count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	T1EN	**Timer 1 count down enable**
6:5	RESV1	**Reserved**
4	T1AL	**Timer 1 auto load enable** 1: Enable 0: Disable
3	T1AL_STATUS	**Timer 1 auto load enable status** 1: Enable 0: Disable
2:0	RESV0	**Reserved**

10000134 T1LMT_REG RISC Timer 1 Limit Register 00000000

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

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

10000138 T1_REG RISC Timer 1 Register 0000FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RESV0**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**T1**															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

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

4.4 Interrupt Controller

4.4.1 Registers

CIRQ Changes LOG

Revision	Date	Author	Change Log
0.1	2012/6/15	YuShu Xiao	Initialization

Module name: CIRQ Base address: (+10000200h)

Address	Name	Width	Register Function
10000200	IRQ_SEL0	32	**IRQ Selection 0 Register** The registers allow the interrupt sources to be mapped onto interrupt requests IRQ. When write data to this register, the FIQ_SEL register will be update to the inverse data at the same time.
10000204	IRQ_SEL1	32	**Reserved** Reserved
10000208	IRQ_SEL2	32	**Reserved** Reserved
1000020C	IRQ_SEL3	32	**Reserved** Reserved
1000026C	FIQ_SEL	32	**FIQ Selection Register** The registers allow the interrupt sources to be mapped onto interrupt requests FIQ. When write data to this register, the IRQ_SEL0 register will be update to the inverse data at the same time.
10000270	IRQ_MASK	32	**IRQ Mask Register** This register contains a mask bit for each interrupt line in IRQ Controller.
10000274	FIQ_MASK	32	**FIQ Mask Register** This register contains a mask bit for each interrupt line in FIQ Controller
10000278	IRQ_MASK_CLR	32	**IRQ Mask Clear Register** This register is used to clear bits in IRQ Mask Register.
1000027C	FIQ_MASK_CLR	32	**FIQ Mask Clear Register** This register is used to clear bits in FIQ Mask Register.
10000280	IRQ_MASK_SET	32	**IRQ Mask Set Register** This register is used to set bits in the IRQ Mask Register.
10000284	FIQ_MASK_SET	32	**FIQ Mask Set Register** This register is used to set bits in the FIQ Mask Register.
10000288	IRQ_EOI	32	**IRQ End of Interrupt Register** This register provides a mean for software to relinquish and to refresh the interrupt controller. Writing a 1 to a specific bit results in an IRQ End of Interrupt command issued internally to the corresponding interrupt line.
1000028C	FIQ_EOI	32	**FIQ End of Interrupt Register** This register provides a mean for software to relinquish and to refresh the interrupt controller. Writing a 1 to a specific bit results in an FIQ End of Interrupt

			command issued internally to the corresponding interrupt line.
10000290	**IRQ_SENS**	32	**IRQ Sensitive Register** This register is used to set the IRQ interrupts as either edge or level sensitive.
10000294	**FIQ_SENS**	32	**FIQ Sensitive Register** This register is used to set the FIQ interrupts as either edge or level sensitive.
10000298	**INT_SOFT**	32	**Software Interrupt Register** Setting 1 to the specific bit position generates a software interrupt for corresponding interrupt line before interrupt input multiplex. This register is used for debug purpose.
1000029C	**IRQ_STAT**	32	**IRQ Status Register** Reading this register will get the IRQ interrupt sources with masking.
100002A0	**FIQ_STAT**	32	**FIQ Status Register** Reading this register will get the FIQ interrupt sources with masking.
100002A4	**INT_PURE**	32	**Interrupt Pure Register** Reading this register will get the pure interrupt sources without masking.
100002A8	**INT_MSEL**	32	**Interrupt Mode Selection Register** This register is used to select the interrupt modes of MIPS1004Kc.

10000200 IRQ_SEL0 IRQ Selection 0 Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IRQ0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IRQ0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Selection 0** 0: Clear IRQ_SEL0 and Set FIQ_SEL 1: Set IRQ_SEL0 and Clear FIQ_SEL

10000204 IRQ_SEL1 Reserved 00000000

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

Bit(s)	Name	Description
31:0	RESV	**Reserved**

10000208 **IRQ_SEL2** **Reserved** **00000000**

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

Bit(s)	Name	Description
31:0	RESV	**Reserved**

1000020C **IRQ_SEL3** **Reserved** **00000000**

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

Bit(s)	Name	Description
31:0	RESV	**Reserved**

1000026C **FIQ_SEL** **FIQ Selection Register** **00000000**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Selection** 0: Clear FIQ_SEL and Set IRQ_SEL0 1: Set FIQ_SEL and Clear IRQ_SEL0

10000270 **IRQ_MASK** **IRQ Mask Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**IRQ0[31:16]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IRQ0[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Mask** 0: Interrupt is disabled 1: Interrupt is enabled

10000274 **FIQ_MASK** **FIQ Mask Register** **00000000**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Mask** 0: Interrupt is disabled 1: Interrupt is enabled

10000278 **IRQ_MASK_CLR** **IRQ Mask Clear Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**IRQ0[31:16]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IRQ0[15:0]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Mask Clear** 0: No effect 1: Clear the corresponding MASK bit

1000027C **FIQ_MASK_CLR** **FIQ Mask Clear Register** **00000000**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Mask Clear** 0: No effect 1: Clear the corresponding MASK bit

10000280 **IRQ_MASK_SET** **IRQ Mask Set Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IRQ0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IRQ0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Mask Set** 0: No effect 1: Set the corresponding MASK bit

10000284 **FIQ_MASK_SET** **FIQ Mask Set Register** **00000000**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Mask Set** 0: No effect 1: Set the corresponding MASK bit

10000288 **IRQ_EOI** **IRQ End of Interrupt Register** **0000000**
0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IRQ0[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IRQ0[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IRQ0	**IRQ End of Interrupt** 0: No service is currently in progress or pending 1: Interrupt request is in-service

1000028C **FIQ_EOI** **FIQ End of Interrupt Register** **0000000**
0

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ End of Interrupt** 0: No service is currently in progress or pending 1: Interrupt request is in-service

10000290 **IRQ_SENS** **IRQ Sensitive Register** **0000000**
1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IRQ0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IRQ0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Sensitive** 0: Edge sensitivity with Pos-edge Edge 1: Level sensitivity with active High

10000294 **FIQ_SENS** **FIQ Sensitive Register** **00000001**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Sensitive** 0: Edge sensitivity with Pos-edge Edge 1: Level sensitivity with active High

10000298 **INT_SOFT** **Software Interrupt Register** **00000000**

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

Bit(s)	Name	Description
31:0	INT	**Software Interrupt**

1000029C **IRQ_STAT** **IRQ Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IRQ0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IRQ0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IRQ0	**IRQ Status** 0: No interrupt request is generated 1: Interrupt request is pending

100002A0 **FIQ_STAT** **FIQ Status Register** **00000000**

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

Bit(s)	Name	Description
31:0	FIQ	**FIQ Status** 0: No interrupt request is generated 1: Interrupt request is pending

100002A4 **INT_PURE** **Interrupt Pure Register** **00000000**

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

Bit(s)	Name	Description
31:0	INT	**Pure Interrupt** 0: No interrupt source is asserted 1: Interrupt source is asserted

100002A8 **INT_MSEL** **Interrupt Mode Selection Register** **00000000**

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

Bit(s)	Name	Description
31:1	RESV	**Reserved**
0	SEL	**Interrupt Mode Selection** 0: Compatibility & Vectored Interrupt Mode 1: External Interrupt Controller Mode

4.5 EMC Controller

4.5.1 Regsiter

EXT_MC_ARB Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/5	Lancelot	Initialization
0.2	2013/8/19	YS Xiao	Modify to MT7628

Module name: EXT_MC_ARB Base address: (+10000300h)

Address	Name	Width	Register Function
10000300	SDRAM_CFG0	32	SDRAM Configuration 0
10000304	SDRAM_CFG1	32	SDRAM Configuration 1
10000308	ILL_ACC_ADDR	32	Illegal Access Address Capture
1000030C	ILL_ACC_TYPE	32	Illegal Access Type Capture
10000310	DDR_SELF_REFRESH	32	ODT and Self-Refresh Configuration
10000314	SDR_DDR_PWR_SAVE_CNT	32	Self-Refresh Time Count
10000320	DLL_DBG	32	DRAM DLL Debug Probe
10000340	DDR_CFG0	32	DDR1/DDR2 controller configuration 0 register
10000344	DDR_CFG1	32	DDR1/DDR2 controller configuration 1 register
10000348	DDR_CFG2	32	DDR1/DDR2 controller configuration 2 register
1000034C	DDR_CFG3	32	DDR1/DDR2 controller configuration 3 register
10000350	DDR_CFG4	32	DDR1/DDR2 controller configuration 4 register
10000360	DDR_DQ_DLY	32	DDR1/DDR2 DQ delay control register
10000364	DDR_DQS_DLY	32	DDR1/DDR2 DQS delay control register
10000368	DDR_DLL_SLV	32	DDR1/DDR2 DLL slave control register
1000036C	DDR_DLL_MST	32	DDR1/DDR2 DLL master control register
10000380	MC_ARB_CFG	32	MC 2 to 1 arbiter setting
10000384	MC_AG_BW	32	MC Channel BW/QoS_Type/DueDate Setting
10000390	RB_DBG	32	RB Debug
10000394	RB_STATE	32	RB Debug State
10000398	RB_BW	32	RB Bandwidth
1000039C	RB_LAT	32	RB Latency

10000300 SDRAM_CFG0 SDRAM Configuration 0 51925282

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DIS_CLK_GT	CLK_SLEW		TWR	TMRD				TRFC				RSV0		TCAS	
Type	RW	RW		RW	RW				RW				RO		RW	

Reset	0	1	0	1	0	0	0	1	1	0	0	1	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TRAS**				**RSV1**		**TRCD**		**TRC**				**RSV2**		**TRP**	
Type	RW				RO		RW		RW				RO		RW	
Reset	0	1	0	1	0	0	1	0	1	0	0	0	0	0	1	0

Bit(s)	Name	Description
31	DIS_CLK_GT	**Disable Clock Gating** Disables clock gating of the SDR DRAM controller. 0: Enable 1: Disable
30:29	CLK_SLEW	**Reserved**
28	TWR	**Write Recovery Time** (unit: system clock cycles - 1)
27:24	TMRD	**Load Mode Register command to any other command delay.** (unit: system clock cycles - 1)
23:20	TRFC	**Auto Refresh period** (unit: system clock cycles - 1)
19:18	RSV0	**Reserved**
17:16	TCAS	**CAS Latency Time** (unit: system clock cycles - 1)
15:12	TRAS	**The Active To Precharge command delay.** (unit: system clock cycles - 1)
11:10	RSV1	**Reserved**
9:8	TRCD	**Active To Read or Write delay (RAS to CAS delay)** (unit: system clock cycles - 1)
7:4	TRC	**Active To Active command period** (unit: system clock cycles - 1)
3:2	RSV2	**Reserved**
1:0	TRP	**Precharge command period** (unit: system clock cycles - 1)

10000304 <u>SDRAM_CFG1</u> **SDRAM Configuration 1** **01120600**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**SDRAM_INIT_START**	**SDRAM_INIT_DONE**	**RBC_MAPPING**	**PWR_DOWN_EN**	**PWR_DOWN_MODE**	**RSV0**		**SDRAM_WIDTH**	**RSV1**		**NUMCOLS**		**RSV2**		**NUMROWS**	
Type	RW	RO	RW	RW	RW	RO		RW	RO		RW		RO		RW	
Reset	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TREFR**															
Type	RW															
Reset	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31	SDRAM_INIT_START	**SDRAM Initialization Start performs the SDRAM initialization sequence.** Can not set this bit to 0 after initialization. 1: Start intialization
30	SDRAM_INIT_DONE	**SDRAM Initialization Done** Indicates the SDRAM has been initialized. 0: Not initialized. 1: Initialized.
29	RBC_MAPPING	**RBC Mapping** Selects the address mapping scheme. 0: {BANK ADDR, ROW ADDR, COL ADDR} address mapping scheme 1: {ROW ADDR, BANK ADDR, COL ADDR} address mapping scheme
28	PWR_DOWN_EN	**Power Down Enable** Enables the SDRAM precharge power-down mode to save standby power. 0: Disable 1: Enable
27	PWR_DOWN_MODE	**Power Down Mode** 0: Precharge power down mode 1: Active power down
26:25	RSV0	**Reserved**
24	SDRAM_WIDTH	**SDRAM Width** Selects the number of SDRAM data bus bits. 0: 16 bits 1: 32 bits
23:22	RSV1	**Reserved**
21:20	NUMCOLS	**Number of Columns** Selects the number of column address bits. 0: 8 Column address bits 1: 9 Column address bits (default) 2: 10 Column address bits 3 11 Column address bits
19:18	RSV2	**Reserved**
17:16	NUMROWS	**Number of Rows** Selects the number of row address bits. 0: 11 Row address bits 1: 12 Row address bits (default) 2: 13 Row address bits 3: 14 Row address bits
15:0	TREFR	**AUTO REFRESH period** (unit: SDRAM clock cycles - 1).

10000308 **ILL_ACC_ADDR** **Illegal Access Address Capture** **00000000**

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

Bit(s)	Name	Description
31:0	ILL_ACC_ADDR	**Illegal Access Address if any bus masters (including CPU) issue illegal accesses (e.g. accesses to reserved memory space, or non-double-word accesses to configuration registers), the address of the illegal transaction is captured in this register.** An illegal interrupt is generated to indicate this exception.

1000030C **ILL_ACC_TYPE** **Illegal Access Type Capture** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ILL_INT_STATUS	ILL_ACC_WR	RSV0										ILL_ACC_BSEL			
Type	W1C	RO	RO										RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1					ILL_IID			ILL_ACC_LEN							
Type	RO					RO			RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	ILL_INT_STATUS	**Illegal Access Interrupt Status** Indicates whether the ilegal access interrupt is cleared or pending. Read 0: Cleared 1: Pending Write 1: Clear both the ILL_ACC_ADDR and ILL_ACC_TYPE registers and thus clear ILL_INT_STATUS.
30	ILL_ACC_WR	**Illegal Access Write** Indicates the illegal access is a read or a write. 0: A read access 1: A write access
29:20	RSV0	**Reserved**
19:16	ILL_ACC_BSEL	**Illegal Access Byte Select** Indicates which bytes were illegally accessed.
15:11	RSV1	**Reserved**
10:8	ILL_IID	**Illegal Access Initiator ID** Indicates the initiator ID of the illegal access. 0: CPU 1: DMA 2: PPE 3: Ethernet PDMA Rx 4: Ethernet PDMA Tx 5: PCI/PCIE 6: Embedded WLAN MAC/BBP 7: USB
7:0	ILL_ACC_LEN	**Illegal Access Length** Indicates the access size of the illegal access.

Bit(s)	Name	Description
		(unit: bytes)

10000310 **DDR_SELF_REFRESH** **ODT and Self-Refresh Configuration** **0E120001**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RSV0**				**ODT_SRC_SEL**				**ODT_OFF_DLY**				**ODT_ON_DLY**			
Type	RO				RW				RW				RW			
Reset	0	0	0	0	1	1	1	0	0	0	0	1	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RSV1**											**SR_AUTO_EN**	**RSV2**		**SRACK_B**	**SRREQ_B**
Type	RO											RW	RO		RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:28	RSV0	**Reserved**
27:24	ODT_SRC_SEL	**ODT Source Select** Sets the DDR pad ODT control source. 0: Dasavtive[0] 1: Dasavtive[1] ... 11: Dasavtive[11] 12: DQS_WINDOW 13: ODT_LOCAL 14: Always on 15: Always off
23:20	ODT_OFF_DLY	**ODT Off Delay** Sets the delay time of the ODT_OFF signal based on the ODT_ON signal. 0: 0 T 1: 0.5 T 2: 1.5 T 3: 2.5 T ... 15: 14.5 T
19:16	ODT_ON_DLY	**ODT On Delay** Sets the delay time of the ODT_ON signal based on the ODT source signal. 0: 0 T 1: 1 T 2: 2 T ... 15: 15 T
15:5	RSV1	**Reserved**
4	SR_AUTO_EN	**Auto Self-Refresh Enable** Enables auto self-refresh for power saving. 0: Disable 1: Enable
3:2	RSV2	**Reserved**
1	SRACK_B	**Self-Refresh Acknowledge Status** Indicates whether DDR2 is in self-refresh mode or has exited from self-refresh mode.

Bit(s)	Name	Description
		When DDR2 changes from self-refresh mode to normal mode, it takes about 200 clock cycles. 0: The DDR2 is in self-refresh mode. 1: The DDR2 has exited from self-refresh mode.
0	SRREQ_B	**Self-Refresh Request Control** Requests DDR2 to enter or exit self-refresh mode. It is low active. 0: Enter self-refresh mode. 1: Exit self-refresh mode.

10000314 **SDR_DDR_PWR_SAVE_CNT** **Self-Refresh Time Count** **0003FFFF**

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

Bit(s)	Name	Description
31:24	PD_CNT	**Power Down Count** Counts the times self-refresh mode is entered
23:0	SR_TAR_CNT	**Self-Refresh Time Count** This counter is only referenced when the SDR (PWR_DOWN_EN) or DDR (SR_AUTO_EN) is set. This counter measures the period SDR or DDR is in IDLE status. When the IDLE period has reached the specified time period, the SDR or DDR automatically enter power-saving or selfrefresh mode. Use the following equations to configure the counter. DRAM_CLK_FREQ is PLL_CLK (600 MHz) divided by 3 DDR: (SR_TAR_CNT * 256 + 255) / DRAM_CLK_FREQ SDR: (SR_TAR_CNT * 256) / DRAM_CLK_FREQ DDR reference table 200 MHz: (32'h03FFFF * 256 + 255) * 5 ns ~= 335 ms SDRAM reference table 120 MHz: 32'h03FFFF * 256 * 8.3 ns ~= 560 ms

10000320 **DLL_DBG** **DRAM DLL Debug Probe** **00000000**

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

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

Bit(s)	Name	Description
31:20	RSV0	Reserved
19:18	RSV1	Reserved
17:12	TDC_STABLE	ADLL master coarse-grain delay code
11:4	MST_DLY_SEL	ADLL master final delay code
3	RSV2	Reserved
2:1	CURR_STATE	ADLL controller FSM current state
0	ADLL_LOCK_DONE	ADLL lock done signal

10000340 DDR_CFG0 DDR1/DDR2 controller configuration 0 register 249B425B

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

Bit(s)	Name	Description
31:28	T_RRD	The minimum number of clock cycles from an active command to the next active command for different banks (TRRD). For DDR2 devices, this is required to be a minimum of 2 regardless of the cycle time.
27:23	T_RAS	The number of clock cycles from an active command until a pre-charge command is allowed. To obtain this value, one should divide the minimum RAS# to pre-charge delay of the SDRAM by the clock cycle time (TRAS). The sum of Active-to-Pre-charge and Pre-charge-to-Active should be equal or larger than active-to-active delay of the same ban (TRC)
22:19	T_RP	The number of clock cycles needed for the SDRAM to recover from a pre-charge command and ready to accept the next active command. To obtain this value, one should divide the RAS# pre-charge time of the SDRAM (TRP) by the clock cycle time. The sum of Active-to-Pre-charge and Pre-charge-to-Active should be equal or larger than active-to-active delay of the same bank (TRC)
18:13	T_RFC	Half the number of clock cycles needed for the SDRAM to recover from a refresh signal to be ready to take the next command. To obtain this value, one should divide the SDRAM row cycle time (TRFC) by the clock cycle time.
12:0	T_REFI	The number of clock cycles from one refresh command to the next refresh command. To obtain this value, one should divide the periodic refresh interval (TREFI) by the clock cycle time. The actual timing of issuing a pre-charge command may be delayed by if the SDRAM is processing a normal access. However, the delay is not accumulative so

Bit(s)	Name	Description
		there is no need to shorten the refresh interval to account for memory access time. The non-accumulative refresh delay typically increases memory bandwidth by a few percentage points.

10000344 **DDR_CFG1** **DDR1/DDR2 controller configuration 1 register** **222E2424**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	T_WTR				T_RTP				RSV0		USER_DATA_WIDTH	IND_SDRAM_SIZE			IND_SDRAM_WIDTH	
Type	RW				RW				RO		RW	RW			RW	
Reset	0	0	1	0	0	0	1	0	0	0	1	0	1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXT_BANK		TOTAL_SDRAM_WIDTH		T_WR				T_MRD				T_RCD			
Type	RW		RW		RW				RW				RW			
Reset	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0

Bit(s)	Name	Description
31:28	T_WTR	**The write-to-read delay (TWTR) (last write data to the next read command) as specified by the DDR2 data sheet**
27:24	T_RTP	**The read-to-pre-charge delay (TRTP) as specified by the DDR2 data sheet. Note that this is a DDR2 requirement, and requires a minimum of 2 cycles. These bits are ignored in DDR mode.**
23:22	RSV0	**Reserved**
21	USER_DATA_WIDTH	**Specify user data width** 0: 32-bit 1: 64-bit When user data width is 32-bit, total SDRAM width (bit[13:12]) must be 10. NOTE: This system is always 64-bit. Please do not modify this setting.
20:18	IND_SDRAM_SIZE	**Specify individual SRAM size** 000: Reserved 001: Individual SDRAM is 64 Mbit, (DDR only) 010: Individual SDRAM is 128 Mbit, (DDR only) 011: Individual SDRAM is 256 Mbit. 100: Individual SDRAM is 512 Mbit. 101: Individual SDRAM is 1 Gbit. 110: Individual SDRAM is 2 Gbit, (DDR2 only). 111: Reserved
17:16	IND_SDRAM_WIDTH	**Specify individual SRAM data width** 00: Reserved 01: 8-bit. 10: 16-bit. 11: Reserved
15:14	EXT_BANK	**Specify bank/module configuration** 00: 1 external bank, 1 module. (CS#[0]) 01: 2 external bank, 1 module. (CS#[1:0]), 10: Reserved 11: 2 external banks, 2 modules. (CS#[1:0])

Bit(s)	Name	Description
		NOTE: only one CS pin.
13:12	TOTAL_SDRAM_WIDTH	**This field specifies the total data width to the SDRAM. For example, if four 8-bit wide DDR2 chips are used in parallel to form a 32-bit DDR2 data width, this field should be defined as 11 to indicate a 32-bit width. In this case, bit[17:16] should be defined as 01.** 00: Reserved 01: Reserved 10: 16-bit 11: 32-bit. Allowed only when user data width is 64-bit (bit21 is 1).
11:8	T_WR	**The clock cycles needed for the DDR to recover from a write command and be able to accept a pre-charge command. To obtain this value, divide the SDRAM write recovery time by the clock cycle time (TWR)**
7:4	T_MRD	**The number of clock cycles after the setting of the mode registers in the DDR and before the issue of the next command. To obtain this value, divide the Mode Register Set Cycle time (TMRD) by the clock cycle time.**
3:0	T_RCD	**The number of clock cycles from an active command to a read/write assertion. To obtain this value, divide the RAS# to CAS# delay time (TRCD) by the clock cycle time.**

10000348 DDR_CFG2 DDR1/DDR2 controller configuration 2 register 43FFE443

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REGE	DDR2_MODE	DQS0_GATING_WINDOW		DQS1_GATING_WINDOW		RSV0[12:3]									
Type	RW	RW	RW		RW		RO									
Reset	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[2:0]			PD	WR			DLL RESET	TESTMODE	CAS_LATENCY			BURST_TYPE	BURST_LENGTH		
Type	RO			RW	RW			RW	RW	RW			RO	RW		
Reset	1	1	1	0	0	1	0	0	0	1	0	0	0	0	1	1

Bit(s)	Name	Description
31	REGE	**This bit should be high when external registers are inserted in the controller and address signals are sent between the controller and the DDR SDRAM. One example of such instance is when register mode SDRAM DIMM is used. This bit should be low when the control and address signals from the controller is connected to the SDRAM without register delay.**
30	DDR2_MODE	**This bit determines whether the memory controller is in DDR1 or DDR2 mode.** 0: DDR1 mode 1: DDR2 mode
29:28	DQS0_GATING_WINDOW	**Controls the mask for the data strobe 0 (DQS0) window leading and trailing edge.** 00: Half extended cycle for the leading and trailing edge of DQS window (maximum window) 01: Only half extended cycle for leading edge of DQS window

Bit(s)	Name	Description
		10: Only half extended cycle for trailing edge of DQS window 11: No extended cycle for leading and trailing edge of DQS window (minimum window)
27:26	DQS1_GATING_WINDOW	**Controls the mask for the data strobe 1 DQS1 window leading and trailing edge.** 00: Half extended cycle for the leading and trailing edge of DQS window (maximum window) 01: Only half extended cycle for leading edge of DQS window 10: Only half extended cycle for trailing edge of DQS window 11: No extended cycle for leading and trailing edge of DQS window (minimum window)
25:13	RSV0	**Reserved**
12	PD	**Active Memory Power Down Exit Time** 0: Fast exit time (TXARD) 1: Slow exit time(TXARDS) This bit is used for DDR2 only. This bit must be 0 for DDR1.
11:9	WR	**Auto Pre-charge Write Recovery (TDAL)** These bits must be 0 for DDR1.
8	DLLRESET	**SDRAM Delay Locked Loop (DLL) Reset** 0: Normal operation 1: Normal operation with DLL reset
7	TESTMODE	**Set SDRAM to run test mode.** 0: Normal operation. 1: Test mode. The user must keep this bit at 0 if SDRAM does not support TESTMODE bit.
6:4	CAS_LATENCY	**Specifies the number of the clock cycles from the assertion of a read/write signal to the SDRAM until the first valid data on the output from the SDRAM. The valid numbers are:** 101: 1.5 for DDR1 or 5 for DDR2. 010: 2 110: 2.5 (DDR1 only) 011: 3 100: 4 (DDR2 only)
3	BURST_TYPE	**This register is hardwired to 0 to indicate a sequential burst type.**
2:0	BURST_LENGTH	**Indicates the burst length of the read/write transaction.** 010: 4 bursts 011: 8 bursts NOTE: 1. A burst of 4 is not allowed when user data is 64-bit while SDRAM data is 16-bit. 2. A burst of 8 is allowed in all user/SDRAM data width combination. 3. Other values for burst length are not allowed.

1000034C **DDR_CFG3** **DDR1/DDR2 controller configuration 3 register** **FFFFE412**

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

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				OF F	OS	_DI FF_ DQ S				1		CY		0		
Type		RO		RW	RW	RW		RW		RW		RW		RW	RW	RW
Reset	1	1	1	0	0	1	0	0	0	0	0	1	0	0	1	0

Bit(s)	Name	Description
31:13	RSV0	**Reserved**
12	Q_OFF	**Output Buffer Disable** 0: Enabled 1: Disabled This bit is used for DDR2 only. This bit must be 0 for DDR1.
11	RDOS	**Redundant Data Strobe (DQS)** This bit enables the redundant DQS function if supported by the SDRAM. 0: Disable 1: Enable This bit is used for DDR2 only and must be 0 for DDR1.
10	DIS_DIFF_DQS	**Disable differential DQS** 0: Enable 1: Disable This bit is used for DDR2 only and must be 0 for DDR1.
9:7	OCD	**Off-Chip Driver Impedance Calibration (OCD)** These bits support the OCD function if supported by the SDRAM. The value programmed in these register bits will be programmed into the SDRAM at EMR1 programming. Settings are vendor-dependant.
6	RTT1	**Internal Termination Resistor (RTT) bit 1** Used together with bit 2 (RTT0) to control On-Die Termination (ODT). Combine values for (RTT1, RTT0) to select ODT settings. 00: ODT disabled. 01: 75 ohm 10: 150 ohm 11: Reserved This bit is used for DDR2 only and must be 0 for DDR1.
5:3	ADDITIVE_LATENCY	**Additive Latency** 000: 0 cycle 001: 1 cycle 010: 2 cycles 011: 3 cycles 100: 4 cycles 101: 5 cycles Others: Reserved This bit is used for DDR2 only and must be 0 for DDR1.
2	RTT0	**Internal Termination Resistor (RTT) bit 0** Used together with bit 6 (RTT1) to control ODT. This bit is used for DDR2 only and must be 0 for DDR1.
1	DS	**SDRAM drive Strength** 0: 100% drive strength. 1: 60% drive strength.
0	DLL	**SDRAM Delay Locked Loop (DLL) Enable** 0: Disable 1: Enable

10000350 **DDR_CFG4** **DDR1/DDR2 controller configuration 4 register** **FFFFFFF4**

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

Bit(s)	Name	Description
31:5	RSV0	**Reserved**
4:0	FAW	**Four Activated Windows (FAW) Period** DDR2 devices impose a restriction in that no more than 4 ACTIVE commands may be issued in a given FAW period. To obtain this value, one should divide the Four Bank Activate period (TFAW) of the DDR by the clock cycle time. These bits are ignored in 4 bank devices.

10000360 **DDR_DQ_DLY** **DDR1/DDR2 DQ delay control register** **00008888**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ_GROUP1_DELAY_SEL								DQ_GROUP0_DELAY_SEL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ_GROUP1_DELAY_COARSE_TUNING				DQ_GROUP1_DELAY_FINE_TUNING				DQ_GROUP0_DELAY_COARSE_TUNING				DQ_GROUP0_DELAY_FINE_TUNING			
Type	RW				RW				RW				RW			
Reset	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	DQ_GROUP1_DELAY_SEL	**Force Data Group 1 (MD8 to MD15) Output Delay. Valid when DQ_DLY_SEL_EN is 1.** bit7~4: for coarse-grain delay setting bit3~0: for fine-grain delay setting
23:16	DQ_GROUP0_DELAY_SEL	**Force Data Group 0 (MD0 to MD7) Output Delay. Valid when DQ_DLY_SEL_EN is 1.** bit7~4: for coarse-grain delay setting bit3~0: for fine-grain delay setting
15:12	DQ_GROUP1_DELAY_COARSE_TUNING	**Data Group 1 (MD8 to MD15) Output Delay Coarse-Grain Tuning** 0x0 to 0x7: Decrease delay by 250 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 250 ps per step.
11:8	DQ_GROUP1_DELAY_FINE_TUNING	**Data Group 1 (MD8 to MD15) Output Delay Fine-Grain Tuning** 0x0 to 0x7: Decrease delay by 30 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 30 ps per step.
7:4	DQ_GROUP0_DELAY_COARSE_TUNING	**Data Group 0 (MD0 to MD7) Output Delay Coarse-Grain Tuning** 0x0 to 0x7: Decrease delay by 250 ps per step. 0x8: Keep DLL delay.

Bit(s)	Name	Description
		0x9 to 0xF: Increase delay by 250 ps per step.
3:0	DQ_GROUP0_DELAY_FINE_TUNING	**Data Group 0 (MD0 to MD7) Output Delay Fine-Grain Tuning** 0x0 to 0x7: Decrease delay by 30 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 30 ps per step.

10000364 **DDR_DQS_DLY** **DDR1/DDR2 DQS delay control register** **00008888**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS1_DELAY_SEL								DQS0_DELAY_SEL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1_DELAY_COARSE_TUNING				DQS1_DELAY_FINE_TUNING				DQS0_DELAY_COARSE_TUNING				DQS0_DELAY_FINE_TUNING			
Type	RW				RW				RW				RW			
Reset	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	DQS1_DELAY_SEL	**Force Data Strobe 1 (MDQS1) Input Delay. Valid when DQS_DLY_SEL_EN is 1** bit7~4: for coarse-grain delay setting bit3~0: for fine-grain delay setting
23:16	DQS0_DELAY_SEL	**Force Data Strobe 0 (MDQS0) Input Delay. Valid when DQS_DLY_SEL_EN is 1** bit7~4: for coarse-grain delay setting bit3~0: for fine-grain delay setting
15:12	DQS1_DELAY_COARSE_TUNING	**Data Strobe 1 Input Delay Coarse-Grain Tuning** 0x0 to 0x7: Decrease delay by 250 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 250 ps per step.
11:8	DQS1_DELAY_FINE_TUNING	**Data Strobe 1 Input Delay Fine-Grain Tuning** 0x0 to 0x7: Decrease delay by 30 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 30 ps per step.
7:4	DQS0_DELAY_COARSE_TUNING	**Data Strobe 0 Input Delay Coarse-Grain Tuning** 0x0 to 0x7: Decrease delay by 250 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 250 ps per step.
3:0	DQS0_DELAY_FINE_TUNING	**Data Strobe 0 Input Delay Fine-Grain Tuning** 0x0 to 0x7: Decrease delay by 30 ps per step. 0x8: Keep DLL delay. 0x9 to 0xF: Increase delay by 30 ps per step.

10000368 **DDR_DLL_SLV** **DDR1/DDR2 DLL slave control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[22:7]															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RSV0[6:0]**							**DLL_SLV_UPDATE_MODE**	**RSV1**			**DQS_DLY_SEL_EN**	**RSV2**			**DQ_DLY_SEL_EN**
Type	RO							RW	RO			RW	RO			RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:9	RSV0	**Reserved**
8	DLL_SLV_UPDATE_MODE	**Set DLL slave update mode.** 0: Update delay code only when bank is activated. 1: Continous update
7:5	RSV1	**Reserved**
4	DQS_DLY_SEL_EN	**0: DQS Input Delay decided by DLL.** 1: Force DQS Input Delay by DQS0_DELAY_SEL / DQS1_DELAY_SEL.
3:1	RSV2	**Reserved**
0	DQ_DLY_SEL_EN	**0: DQ Output Delay decided by DLL.** 1: Force DQ Output Delay by DQ_GROUP0_DELAY_SEL / DQ_GROUP1_DELAY_SEL.

1000036C **DDR_DLL_MST** **DDR1/DDR2 DLL master control register** **00000000**

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

Bit(s)	Name	Description
31	DLL_MAS_RELOCK_EN	**Delayed Locked Loop (DLL) Master Relock Enable** 0: Disable relocking scheme. 1: Enable relocking scheme. DLL supports restarting locking from initial value if DLL is not locked after waiting 512 cycles.
30:26	RSV0	**Reserved**
25	DLL_MAS_BYPASS_FD	**DLL Bypass Fine Grain Delay** 0: Fine-grain delay code is determined by DLL. 1: Fine-grain delay code is fixed by DLL_MAS_FIXED_FD.
24	DLL_MAS_BYPASS	**DLL Bypass Coarse Grain Delay**

Bit(s)	Name	Description
	_CD	0: Coarse-grain delay code is determined by DLL 1: Coarse-grain delay code is fixed by DLL_MAS_FIXED_CD.
23:12	RSV1	**Reserved**
11:8	DLL_MAS_FIXED_FD	**DLL Fixed Fine Grain Delay** Specifies the fine-grain delay. The effective range is 0 to 15. Each step is about 30 ps.
7:6	RSV2	**Reserved**
5:0	DLL_MAS_FIXED_CD	**DLL Fixed Coarse Grain Delay** Specifies the coarse-grain delay. The delay = ((x-2)/4-1)*250 ps, the effective range of x is 10 to 52.

10000380 **MC_ARB_CFG** **MC 2 to 1 arbiter setting** **07FAC688**

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

Bit(s)	Name	Description
31:27	RSV0	**Reserved**
26	preempt_en	**Preemption Enable** Request preemption, higher priority requestor may change current request transaction 0: Disable Preemption 1: Enable Preemption
25	trtc_en	**Two Rate Three Color Bandwidth (TRTC) Meter Enable** 0: Disable TRTC 1: Enable TRTC
24	class_en	**QoS Classifier Enable** 0: Disable CLASS 1: Enable CLASS TRTC (0) CLASS (0) Round Robin TRTC (0) CLASS (1) Fixed Priority TRTC (1) CLASS (0) BW RR TRTC (1) CLASS (1) QoS Arb
23:0	cls_priority	**Class Priority** This field is used for class priority for second arbitration. {BEy(3'd7), LCg(3'd6), BSy(3'd5), LSy(3'd4), BEg (3'd3), BSg (3'd2), LSg(3'd1), LCgd(3'd0)}

10000384 **MC_AG_BW** **MC Channel BW/QoS_Type/DueDate Setting** **0110FF4**

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ag_wr	RSV0		ag_sel	RSV1		ag_qos_type		ag_duedate							
Type	WO	RO		RW	RO		RW		RW							
Reset	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ag_pir								ag_cir							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0

Bit(s)	Name	Description
31	ag_wr	**Agent Write** 0: Read 1: Write
30:29	RSV0	**Reserved**
28	ag_sel	**DMA Agent Select** Selects a DMA agent to configure. 0: CPU (Rbus0) 1: DMA (Rbus1)
27:26	RSV1	**Reserved**
25:24	ag_qos_type	**Agent QoS Type** 0: Latency critical 1: Latency sensitive (CPU) 2: Bandwidth sensitive (DMA) 3: Best Effort
23:16	ag_duedate	**Due date for latency critical agent** (unit: system bus clock cycle - system bus is 300 MHz or 225 MHz depending on bootstrap value.)
15:8	ag_pir	**Peak Information Rate for the Agent** The PIR is greater than or equal to the CIR. Bandwidth which exceeds PIR is marked red. 0x00: 0 MB/s 0x01: 8 MB/s ... 0x40: 512 MB/s ... 0xFF: 2040 MB/s (Max)
7:0	ag_cir	**Committed Information Rate for the Agent** Bandwidth which falls below the CIR is marked green. BW which exceeds the CIR but is below the EIR is marked yellow. 0x00: 0 MB/s 0x01: 8 MB/s ... 0x40: 512 MB/s (default) ... 0xFF: 2040 MB/s (Max)

10000390 **RB_DBG** **RB Debug** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[30:15]															
Type	RO															

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

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	rb_sel	**RB channel select for debug message dump**

10000394 RB_STATE RB Debug State 00000000

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

Bit(s)	Name	Description
31:11	RSV0	**Reserved**
10	rb_rw	**RB channel RW**
9:8	rb_state	**RB channel State** 2'b00: IDLE 2'b01: REQ 2'b10: ACK 2'b11: DATA
7:0	rb_length	**RB channel burst length (Byte)**

10000398 RB_BW RB Bandwidth 00000000

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

Bit(s)	Name	Description
31	bw_rst	**Write 1 will reset BW values.**
30	RSV0	**Reserved**
29:20	avg_bw	**Average BW (MB/S)**

Bit(s)	Name	Description
19:10	peak_bw	**Peak BW (MB/S)**
9:0	rb_bw	**RB channel BW (MB/S)**

1000039C **RB_LAT** **RB Latency** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	lat_rst	RSV0	avg_lat										peak_lat[9:6]			
Type	WO	RO	RO										RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

Bit(s)	Name	Description
31	lat_rst	**Write 1 will reset latency values**
30	RSV0	**Reserved**
29:20	avg_lat	**Average read latency (T)**
19:10	peak_lat	**Peak read latency (T)**
9:0	rd_lat	**RB channel read latency (T)**

4.6 **R-Bus Controller**

4.6.1 Features

- 8 channel QoS Arbiter
- Configurable Bandwidth and Duedate for each agent
- QoS classifier can be programmed for RR, BW RR, Fixed Priority and QoS Arb

4.6.2 Block Diagram

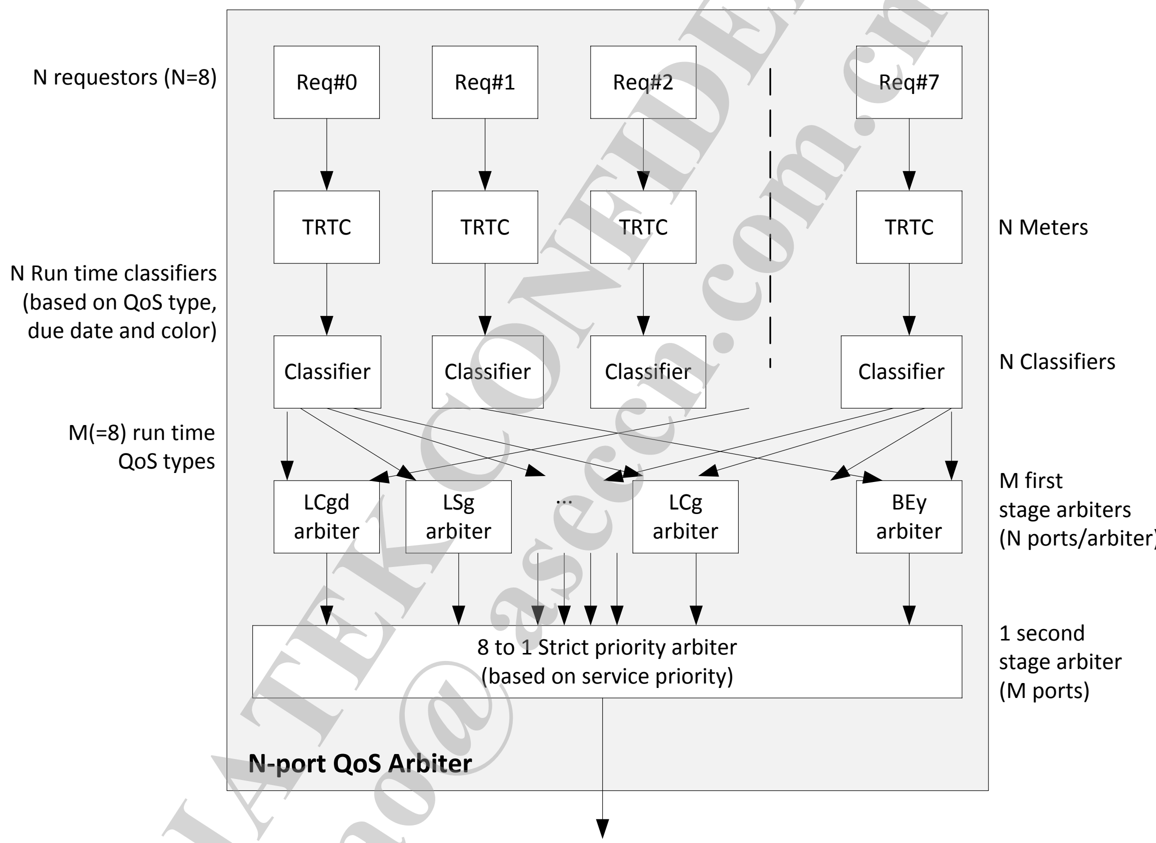


Figure 4-3 QoS Arbitration Block Diagram

4.6.3 Regsiter

Rbus_Matrix_CTRL Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/2	Lancelot	Initialization
0.2	2013/1/3	Lancelot	Add sleep count
0.2	2013/8/19	YS Xiao	Modify to as MT7621's dma_ch_csr

Module name: Rbus_Matrix_CTRL Base address: (+10000400h)

Address	Name	Width	Register Function
10000400	DMA_ARB_CFG	32	DMA 8 to 1 arbiter setting
10000404	DMA_AG_BW_CFG	32	DMA Channel BW/QoS_Type/DueDate Setting
1000040C	DMA_ROUTE	32	DMA Routing
10000410	DMA_MON_AG_SEL	32	DMA Monitor Agent Select
10000414	DMA_STATE	32	DMA State
10000418	DMA_BW	32	DMA Bandwidth
1000041C	DMA_LAT	32	DMA Latency
10000420	OCP_CFG0	32	OCP to Rbus configuration
10000424	OCP_CFG1	32	Read bypass write mask
10000430	R2P_MONITOR	32	Rbus to APbus monitor
10000434	R2P_ERR_ADDR	32	Rbus to APbus error address
10000440	DYN_CFG0	32	Dynamic cpu/ocp frequency control
10000444	DYN_CFG1	32	CPU sleep step frequency control
10000448	DYN_CFG2	32	Dyn CFG Probe
1000044C	DYN_CFG3	32	SI_Sleep Serial Counter Setting
10000450	DYN_CFG4	32	SI_Sleep Issue Count Counter
10000454	DYN_CFG5	32	Sleep Time Counter for SI_Sleep
10000458	DYN_CFG6	32	Operation Time Counter for non SI_Sleep

10000400 DMA_ARB_CFG DMA 8 to 1 arbiter setting 04FAC688

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

Bit(s)	Name	Description
31:27	RSV0	**Reserved**
26	preempt_en	**Preemption Enable** Request preemption, higher priority requestor may change current request transaction 0: Disable Preemption 1: Enable Preemption
25	trtc_en	**Two Rate Three Color Bandwidth (TRTC) Meter Enable** 0: Disable TRTC

Bit(s)	Name	Description
		1: Enable TRTC
24	class_en	**QoS Classifier Enable** 0: Disable CLASS 1: Enable CLASS TRTC (0) CLASS (0) Round Robin TRTC (0) CLASS (1) Fixed Priority TRTC (1) CLASS (0) BW RR TRTC (1) CLASS (1) QoS Arb
23:0	cls_priority	**Class Priority** This field is used for class priority for second arbitration. {BEy(3'd7), LCg(3'd6), BSy(3'd5), LSy(3'd4), BEg (3'd3), BSg (3'd2), LSg(3'd1), LCgd(3'd0)}

10000404 **DMA_AG_BW_CFG** **DMA Channel BW/QoS_Type/DueDate Setting** **02208020**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**ag_wr**	**ag_sel**			**RSV0**		**ag_qos_type**		**ag_duedate**							
Type	W1C	RW			RO		RW		RW							
Reset	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**ag_pir**								**ag_cir**							
Type	RW								RW							
Reset	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit(s)	Name	Description
31	ag_wr	**Agent Write** 0: Read 1: Write
30:28	ag_sel	**DMA Agent Select** Selects a DMA agent to configure. 0: SDXC 1: GDMA 2: SPI Slave/3-Wire SPI Slave/PUTIF 3: Switch 4: WLAN 5: PCIe 6: AES 7: USB20
27:26	RSV0	**Reserved**
25:24	ag_qos_type	**Agent QoS Type** 0: Latency critical 1: Latency sensitive 2: Bandwidth sensitive (default) 3: Best Effort
23:16	ag_duedate	**Due date for latency critical agent** (unit: system bus clock cycle - system bus is 300 MHz or 225 MHz depending on bootstrap value.)
15:8	ag_pir	**Peak Information Rate for the Agent** The PIR is greater than or equal to the CIR. Bandwidth which exceeds PIR is marked red.

Bit(s)	Name	Description
		0x00: 0 MB/s 0x01: 4 MB/s ... 0x80: 512 MB/s (default) ... 0xFF: 1020 MB/s (Max)
7:0	ag_cir	**Committed Information Rate for the Agent** Bandwidth which falls below the CIR is marked green. BW which exceeds the CIR but is below the EIR is marked yellow. 0x00: 0 MB/s 0x01: 4 MB/s ... 0x20: 128 MB/s (default) ... 0xFF: 1020 MB/s (Max)

1000040C **DMA_ROUTE** **DMA Routing** **00000000**

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

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	dma_route	**DMA routing** 0: DMA will access to DRAM 1: DMA will access to CSR

10000410 **DMA_MON_AG_SEL** **DMA Monitor Agent Select** **00000000**

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

Bit(s)	Name	Description
31:3	RSV0	**Reserved**
2:0	dma_sel	**DMA Monitor Agent Select**

Bit(s)	Name	Description
		Selects a DMA agent to dump DMA_STATE, DMA_BW and DMA_LAT's content. 0: SDXC 1: GDMA 2: SPI Slave/3-Wire SPI Slave/PUTIF 3: Switch 4: WLAN 5: PCIe 6: AES 7: USB20

10000414 DMA_STATE DMA State 00000000

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

Bit(s)	Name	Description
31:11	RSV0	**Reserved**
10	dma_rw	**DMA channel RW state**
9:8	dma_state	**DMA channel State** 2'b00: IDLE 2'b01: REQ 2'b10: ACK 2'b11: DATA
7:0	dma_length	**DMA channel burst length (Byte) state**

10000418 DMA_BW DMA Bandwidth 00000000

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

Bit(s)	Name	Description
31	bw_rst	**Write 1 will reset BW values.**
30	RSV0	**Reserved**

Bit(s)	Name	Description
29:20	avg_bw	**Average BW (MB/S)**
19:10	peak_bw	**Peak BW (MB/S)**
9:0	dma_bw	**DMA channel BW (MB/S)**

1000041C **DMA_LAT** **DMA Latency** **00000000**

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

Bit(s)	Name	Description
31	lat_rst	**Write 1 will reset latency values**
30	RSV0	**Reserved**
29:20	avg_lat	**Average read latency (T)**
19:10	peak_lat	**Peak read latency (T)**
9:0	rd_lat	**DMA channel read latency (T)**

10000420 **OCP_CFG0** **OCP to Rbus configuration** **00000001**

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

Bit(s)	Name	Description
31:4	RSV0	**Reserved**
3	sync_method	**OCP Synchronization Command Method** 0: All empty (Wait until all FIFOs are empty) 1: CMD empty (Wait until the CMD FIFO is empty)
2	ocp_sync_cmd	**OCP Synchronization Command Method Enable** Remaps this RD CMD to address 0x0000_0000. Initiate DRAM control before enabling this option. 0: Disable 1: Enable

Bit(s)	Name	Description
1	rbus_async	**Async Mode for RBUS** 0: Set HW to switch between sync or async mode dynamically. 1: Force RBUS to A.sync mode.
0	rd_bypass_wr	**Read Bypass Write Enable** Allows read commands to bypass write commands for OCP_IF when the address does not conflict. 0: Disable 1: Enable

10000424 **OCP_CFG1** **Read bypass write mask** **FFFFFFFF**

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

Bit(s)	Name	Description
31:0	rd_bypass_wr_mask	**Mask bit for read bypass write address**

10000430 **R2P_MONITOR** **Rbus to APbus monitor** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0															r2p_inc_clr
Type	RO															W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r2p_err_cnt						r2p_inc_cnt									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:17	RSV0	**Reserved**
16	r2p_inc_clr	**R2APB Interrupt Clear** Write 1 to clear this interrupt.
15:10	r2p_err_cnt	**R2APB error counter**
9:0	r2p_inc_cnt	**R2APB Interrupt Countdown Timer** Sets a delay timer which begins counting down when an R2P error is detected. When the timer reaches zero the R2P interrupt is then triggered. 10'b0000000000: Disable R2P monitoring 10'b0000000001: 20 us

Bit(s)	Name	Description
		10'b0000000010: 40 us ... 10'b1000000000: 40 ms

10000434 **R2P_ERR_ADDR** **Rbus to APbus error address** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	r2p_err_addr[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r2p_err_addr[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	r2p_err_addr	**R2APB address record for previous error found**

10000440 **DYN_CFG0** **Dynamic cpu/ocp frequency control** **00030A01**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0													cpu_ocp_ratio		
Type	RO													RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1				cpu_fdiv				RSV2				cpu_ffrac			
Type	RO				RW				RO				RW			
Reset	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:19	RSV0	**Reserved**
18:16	cpu_ocp_ratio	**CPU OCP Ratio** The ratio between the system bus frequency and the CPU frequency. 3'b011: SYS/CPU = 1/3 3'b100: SYS/CPU = 1/4 (Not used in MT7628)
15:12	RSV1	**Reserved**
11:8	cpu_fdiv	**CPU Frequency Divider** The frequency divider is used to generate the CPU frequency. Valid values range from 1 to 15. NOTE1: CPU_FDIV must be equaled to N*CPU_FFRAC(N is a integer number) when rbus_async equal to 1'b0. NOTE2: CPU_FDIV must be larger than or equal to CPU_FFRAC when rbus_async equal to 1'b1.
7:4	RSV2	**Reserved**
3:0	cpu_ffrac	**CPU Frequency Fractional** A parameter used in conjunction with the CPU frequency divider to determine the CPU frequency. Input a value in the following equation to determine the CPU frequency. CPU frequency = PLL_FREQ*(CPU_FFRAC/CPU_FDIV)

Bit(s)	Name	Description
		NOTE: If the chip runs in USB OHCI mode, the OCP frequency cannot be lower than 30 MHz. It means that PLL_FREQ*(CPU_FFRAC/CPU_FDIV)/CPU_OCP_RATIO >= 30 MHz.

10000444 DYN_CFG1 CPU sleep step frequency control 00230A06

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	slp_en	step_en	RSV0		step_cnt								RSV1	step_ocp_ratio		
Type	RW	RW	RO		RW								RO	RO		
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV2				step_fdiv				RSV3				step_ffrac			
Type	RO				RO				RO				RW			
Reset	0	0	0	0	1	0	1	0	0	0	0	0	0	1	1	0

Bit(s)	Name	Description
31	slp_en	**Sleep Mode Enable** Enables sleep mode when MIPS SI_Sleep is asserted. 0: Disable 1: Enable Sleep Mode CPU Frequency = PLL_FREQ*(1/CPU_FDIV)
30	step_en	**Step Jump Enable** Enables step jump after MIPS exits sleep mode. The CPU will jump to the normal frequency in increments defined by STEP_FFRAC.bit[4:0] of this register. 0: Disable 1: Enable
29:28	RSV0	**Reserved**
27:20	step_cnt	**Step Counter** Sets the period of each step jump. When the counter counts down to zero, the CPU clock automatically changes to the next step frequency. The count period unit is 1 us.
19	RSV1	**Reserved**
18:16	step_ocp_ratio	**Step OCP Ratio (Fix to cpu_ocp_ratio)** The ratio between the system bus frequency and the CPU frequency. 3'b011: SYS/CPU = 1/3 3'b100: SYS/CPU = 1/4 (Not used in MT7628)
15:12	RSV2	**Reserved**
11:8	step_fdiv	**Step Frequency Divider (Fix to CPU_FDIV)** The frequency divider is used to generate the CPU frequency after the CPU exits from sleep mode and returns to normal operation. Valid values range from 1 to 15.
7:4	RSV3	**Reserved**
3:0	step_ffrac	**Step Frequency Fraction** The fractional size of the increment in CPU frequency after the CPU exits from sleep mode and returns to normal operation. This step is only valid when SLP_STEP_EN is enabled. FRAC_VALUE = PREVIOUS_FRAC_VALUE + STEP_FFRAC CPU Frequency = (FRAC_VALUE/CPU_FDIV)*PLL_FREQ

10000448 **DYN_CFG2** **Dyn CFG Probe** **00030A01**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0					dfc_fsm			RSV1			same_freq	RSV2	cpu_ocp_ratio		
Type	RO					RO			RO			RO	RO	RO		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV3				cpu_fdiv				RSV4				cpu_ffrac			
Type	RO				RO				RO				RO			
Reset	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:27	RSV0	**Dynamic frequency controller's main FSM current state**
26:24	dfc_fsm	**Dynamic frequency controller's main FSM current state**
23:21	RSV1	**Reserved**
20	same_freq	**Indicates that the SYS and DRAM clocks are on the same frequency.**
19	RSV2	**Reserved**
18:16	cpu_ocp_ratio	**OCP ratio after changed frequency**
15:12	RSV3	**Reserved**
11:8	cpu_fdiv	**CPU fdiv after changed frequency**
7:4	RSV4	**Reserved**
3:0	cpu_ffrac	**CPU ffrac after changed frequency**

1000044C **DYN_CFG3** **SI_Sleep Serial Counter Setting** **00000000**

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

Bit(s)	Name	Description
31	si_slp_cnt_en	**SI_Sleep Serial Counter Enable**
30:28	RSV0	**Reserved**
27:0	si_slp_time_unit	**SI_Sleep Time Counter unit** 28'h0000000: count per 1us 28'h0000001: count per 2us 28'h0000002: count per 3us ...

Bit(s)	Name	Description
		28'hfffffff: count per 268435456us

10000450 **DYN_CFG4** **SI_Sleep Issue Count Counter** **00000000**

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

Bit(s)	Name	Description
31:0	si_slp_cnt	**SI_Sleep Issue Count Counter** Write to this register will clear the counter value.

10000454 **DYN_CFG5** **Sleep Time Counter for SI_Sleep** **00000000**

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

Bit(s)	Name	Description
31:0	si_slp_time_unit_cnt	**Sleep Time Counter for SI_Sleep** Finally, CPU in SI_Sleep time is "si_slp_time_unit_cnt*si_slp_time_unit(us)". Write to this register will clear the counter value.

10000458 **DYN_CFG6** **Operation Time Counter for non SI_Sleep** **00000000**

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

Bit(s)	Name	Description
31:0	si_opt_time_unit_cnt	**Operation Time Counter for non SI_Sleep** Finally, CPU in non SI_Sleep time is

Bit(s)	Name	Description
		"si_opt_time_unit_cnt*si_slp_time_unit(us)". Write to this register will clear the counter value.

4.7 **MIPS CNT**

4.7.1 Registers

MIPS_CNT Changes LOG

Revision	Date	Author	Change Log
0.1	2013/1/14	YuShu Xiao	Initialization

Module name: MIPS_CNT Base address: (+10000500h)

Address	Name	Width	Register Function
10000500	STCK_CNT_CFG	32	**MIPS Configuration**
10000504	CMP_CNT	32	**MIPS Compare** Sets the cutoff point for the free run counter (MIPS counter). If the free run counter equals the compare counter, then the timer circuit generates an interrupt. The interrupt remains active until the compare counter is written again.
10000508	CNT	32	**MIPS Counter** The MIPS counter (free run counter) increases by 1 every 20 us (50 KHz). The counter continues to count until it reaches the value loaded into CMP_CNT.

10000500 STCK_CNT_CFG MIPS Configuration 00000000

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

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

10000504 CMP_CNT MIPS Compare 0000000

0

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

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

10000508 **CNT** **MIPS Counter** **00000000**

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

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

4.8 General Purpose IO

4.8.1 Features

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

4.8.2 Block Diagram

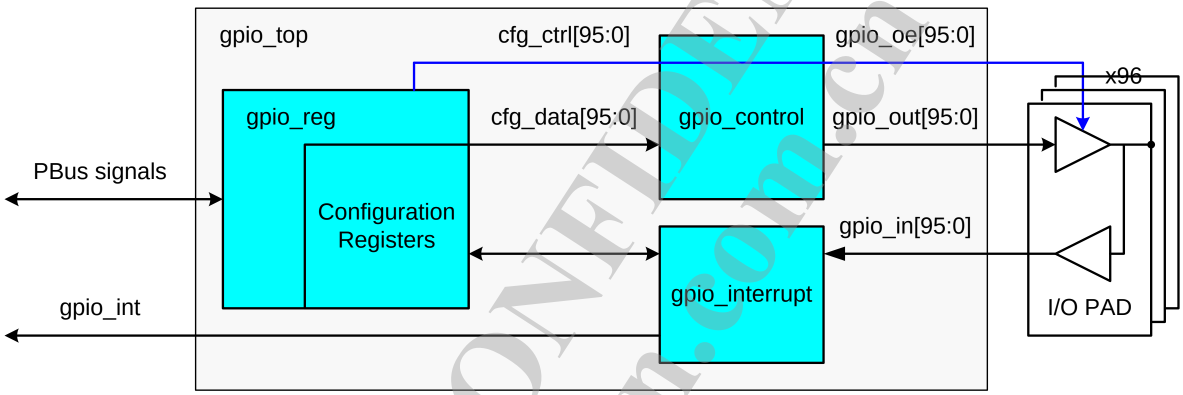


Figure 4-4 Programmable I/O Block Diagram

4.8.3 GPIO pin mapping

PAD Name	Function 0	Function 1	Function 2	Function 3	strap	pmux_group	GPIO
PAD_I2S_SDI	i2ssdi (I)	gpio (I/O)	pcmdrx (I)	antsel[5] (O)		i2s_gpio_psel[2:0]	0
PAD_I2S_SDO	i2ssdo (O)	gpio (I/O)	pcmdtx (O)	antsel[4] (O)	0	i2s_gpio_psel[2:0]	1
PAD_I2S_WS	i2sws(I/O)	gpio (I/O)	pcmclk (I/O)	antsel[3] (O)		i2s_gpio_psel[2:0]	2
PAD_I2S_CLK	i2sclk (I/O)	gpio (I/O)	pcmfs (I/O)	antsel[2] (O)		i2s_gpio_psel[2:0]	3
PAD_I2C_SCLK	i2c_sclk (I/O)	gpio (I/O)	sutif_txd (O)	ext_bgclk (I)		i2c_gpio_psel[2:0]	4
PAD_I2C_SD	i2c_sd (I/O)	gpio (I/O)	sutif_rxd (I)			i2c_gpio_psel[2:0]	5
PAD_SPI_CS1	spi_cs1 (O)	gpio (I/O)	co_clko (O)		1	spi_cs1_psel[2:0]	6
PAD_SPI_CLK	spi_clk (O)	gpio (I/O)			2	spi_gpio_psel[1:0]	7
PAD_SPI_MOSI	spi_mosi (I/O)	gpio (I/O)			3	spi_gpio_psel[1:0]	8
PAD_SPI_MISO	spi_miso (I/O)	gpio (I/O)				spi_gpio_psel[1:0]	9
PAD_SPI_CS0	spi_cs0 (O)	gpio (I/O)				spi_gpio_psel[1:0]	10
PAD_GPIO0	gpio (I/O)	gpio (I/O)	co_clko (O)	perst_n (O)	4	gpio_psel[2:0]	11
PAD_TXD0	txd0 (O)	gpio (I/O)			5	uart0_gpio_psel[2:0]	12
PAD_RXD0	rxd0 (I)	gpio (I/O)				uart0_gpio_psel[2:0]	13
PAD_MDI_TP_P1	spis_cs (I)	gpio (I/O)	w_utif[0] (I/O)	pwm_ch0 (O)		spis_gpio_psel[2:0]	14
PAD_MDI_TN_P1	spis_clk (I)	gpio (I/O)	w_utif[1] (I/O)	pwm_ch1 (O)		spis_gpio_psel[2:0]	15
PAD_MDI_RP_P1	spis_miso (O)	gpio (I/O)	w_utif[2] (I/O)	txd2 (O)		spis_gpio_psel[2:0]	16
PAD_MDI_RN_P1	spis_mosi (I)	gpio (I/O)	w_utif[3] (I/O)	rxd2 (I)		spis_gpio_psel[2:0]	17
PAD_MDI_RP_P2	pwm_ch0 (O)	gpio (I/O)	w_utif[4] (I/O)	sd_d7 (I/O)		pwm0_gpio_psel[2:0]	18
PAD_MDI_RN_P2	pwm_ch1 (O)	gpio (I/O)	w_utif[5] (I/O)	sd_d6 (I/O)		pwm1_gpio_psel[2:0]	19
PAD_MDI_TP_P2	txd2 (O)	gpio (I/O)	pwm_ch2 (O)	sd_d5 (I/O)		uart2_gpio_psel[2:0]	20
PAD_MDI_TN_P2	rxd2 (I)	gpio (I/O)	pwm_ch3 (O)	sd_d4 (I/O)		uart2_gpio_psel[2:0]	21
PAD_MDI_TP_P3	sd_wp (I)	gpio (I/O)	w_utif[10] (I/O)	w_dbgin (I)		sd_gpio_psel[2:0]	22
PAD_MDI_TN_P3	sd_cd (I)	gpio (I/O)	w_utif[11] (I/O)	w_dbgack (O)		sd_gpio_psel[2:0]	23
PAD_MDI_RP_P3	sd_d1 (I/O)	gpio (I/O)	w_utif[12] (I/O)	w_jtclk (I)		sd_gpio_psel[2:0]	24
PAD_MDI_RN_P3	sd_d0 (I/O)	gpio (I/O)	w_utif[13] (I/O)	w_jtdi (I)		sd_gpio_psel[2:0]	25
PAD_MDI_RP_P4	sd_clk (I/O)	gpio (I/O)	w_utif[14] (I/O)	w_jtdo (O)		sd_gpio_psel[2:0]	26
PAD_MDI_RN_P4	sd_cmd (I/O)	gpio (I/O)	w_utif[15] (I/O)	dbg_uart_txd (O)		sd_gpio_psel[2:0]	27
PAD_MDI_TP_P4	sd_d3 (I/O)	gpio (I/O)	w_utif[16] (I/O)	w_jtms (I)		sd_gpio_psel[2:0]	28
PAD_MDI_TN_P4	sd_d2 (I/O)	gpio (I/O)	w_utif[17] (I/O)	w_jtrst_n (I)		sd_gpio_psel[2:0]	29
PAD_EPHY_LED4_K	ephy_led4_k (O)	gpio (I/O)	w_utif_k[6] (I/O)	jtrstn_k (I)		p4_led_kn_psel[2:0]	30
PAD_EPHY_LED3_K	ephy_led3_k (O)	gpio (I/O)	w_utif_k[7] (I/O)	jtclk_k (I)		p3_led_kn_psel[2:0]	31
PAD_EPHY_LED2_K	ephy_led2_k (O)	gpio (I/O)	w_utif_k[8] (I/O)	jtms_k (I)		p2_led_kn_psel[2:0]	32
PAD_EPHY_LED1_K	ephy_led1_k (O)	gpio (I/O)	w_utif_k[9] (I/O)	jtdi_k (I)		p1_led_kn_psel[2:0]	33
PAD_EPHY_LED0_K	ephy_led0_k (O)	gpio (I/O)		jtdo_k (I/O)		p0_led_kn_psel[2:0]	34
PAD_WLED_K	wled_k (I/O)	gpio (I/O)				wled_kn_psel[2:0]	35
PAD_PERST_N	perst_n (O)	gpio (I/O)			6	prest_gpio_psel[1:0]	36
PAD_CO_CLKO	co_clko (O)	gpio (I/O)			7	rclk_gpio_psel[1:0]	37
PAD_WDT_RST_N	wdt (I/O)	gpio (I/O)				wdt_gpio_psel[1:0]	38
PAD_EPHY_LED4_N	ephy_led4_n (O)	gpio (I/O)	w_utif_n[6] (I/O)	jtrstn_n (I)		p4_led_gpio_psel[2:0]	39
PAD_EPHY_LED3_N	ephy_led3_n (O)	gpio (I/O)	w_utif_n[7] (I/O)	jtclk_n (I)		p3_led_gpio_psel[2:0]	40
PAD_EPHY_LED2_N	ephy_led2_n (O)	gpio (I/O)	w_utif_n[8] (I/O)	jtms_n (I)		p2_led_gpio_psel[2:0]	41
PAD_EPHY_LED1_N	ephy_led1_n (O)	gpio (I/O)	w_utif_n[9] (I/O)	jtdi_n (I)		p1_led_gpio_psel[2:0]	42
PAD_EPHY_LED0_N	ephy_led0_n (O)	gpio (I/O)		jtdo_n (I/O)		p0_led_gpio_psel[2:0]	43
PAD_WLED_N	wled_n (I/O)	gpio (I/O)				wled_gpio_psel[2:0]	44
PAD_TXD1	txd1 (O)	gpio (I/O)	pwm_ch0 (O)	antsel[1] (O)	8	uart1_gpio_psel[2:0]	45
PAD_RXD1	rxd1 (I)	gpio (I/O)	pwm_ch1 (O)	antsel[0] (O)		uart1_gpio_psel[2:0]	46

4.8.4 Register

GPIO Changes LOG

Revision	Date	Author	Change Log
0.1	2012/6/21	YuShu Xiao	Initialization

Module name: GPIO Base address: (+10000600h)

Address	Name	Width	Register Function
10000600	GPIO_CTRL_0	32	**GPIO0 to GPIO31 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
10000604	GPIO_CTRL_1	32	**GPIO32 to GPIO63 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
10000608	GPIO_CTRL_2	32	**GPIO64 to GPIO95 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
10000610	GPIO_POL_0	32	**GPIO0 to GPIO31 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
10000614	GPIO_POL_1	32	**GPIO32 to GPIO63 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
10000618	GPIO_POL_2	32	**GPIO64 to GPIO95 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
10000620	GPIO_DATA_0	32	**GPIO0 to GPIO31 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
10000624	GPIO_DATA_1	32	**GPIO32 to GPIO63 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
10000628	GPIO_DATA_2	32	**GPIO64 to GPIO95 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
10000630	GPIO_DSET_0	32	**GPIO0 to GPIO31 data set register** These data set registers are used to set bits in the GPIO_DATA_x registers.
10000634	GPIO_DSET_1	32	**GPIO32 to GPIO63 data set register** These data set registers are used to set bits in the GPIO_DATA_x registers.
10000638	GPIO_DSET_2	32	**GPIO64 to GPIO95 data set register** These data set registers are used to set bits in the

			GPIO_DATA_x registers.
10000640	GPIO_DCLR_0	32	**GPIO0 to GPIO31 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
10000644	GPIO_DCLR_1	32	**GPIO32 to GPIO63 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
10000648	GPIO_DCLR_2	32	**GPIO64 to GPIO95 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
10000650	GINT_REDGE_0	32	**GPIO0 to GPIO31 rising edge interrupt enable register** These registers are used to enable the condition of rising edge triggered interrupt.
10000654	GINT_REDGE_1	32	**GPIO32 to GPIO63 rising edge interrupt enable register** These registers are used to enable the condition of rising edge triggered interrupt.
10000658	GINT_REDGE_2	32	**GPIO64 to GPIO95 rising edge interrupt enable register** These registers are used to enable the condition of rising edge triggered interrupt.
10000660	GINT_FEDGE_0	32	**GPIO0 to GPIO31 falling edge interrupt enable register** These registers are used to enable the condition of falling edge triggered interrupt.
10000664	GINT_FEDGE_1	32	**GPIO32 to GPIO63 falling edge interrupt enable register** These registers are used to enable the condition for falling edge triggered interrupt.
10000668	GINT_FEDGE_2	32	**GPIO64 to GPIO95 falling edge interrupt enable register** These registers are used to enable the condition of falling edge triggered interrupt.
10000670	GINT_HLVL_0	32	**GPIO0 to GPIO31 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_0 can not be set to 1 at the same time.
10000674	GINT_HLVL_1	32	**GPIO32 to GPIO63 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_1 can not be set to 1 at the same time.
10000678	GINT_HLVL_2	32	**GPIO64 to GPIO95 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_2 can not be set to 1 at the same time.
10000680	GINT_LLVL_0	32	**GPIO0 to GPIO31 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_0 can not be set to 1 at the same time.
10000684	GINT_LLVL_1	32	**GPIO32 to GPIO63 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_1 can not be set to 1 at the same time.
10000688	GINT_LLVL_2	32	**GPIO64 to GPIO95 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_2

			can not be set to 1 at the same time.
10000690	GINT_STAT_0	32	**GPIO0 to GPIO31 interrupt status register** These registers are used to record the GPIO current interrupt status.
10000694	GINT_STAT_1	32	**GPIO32 to GPIO63 interrupt status register** These registers are used to record the GPIO current interrupt status.
10000698	GINT_STAT_2	32	**GPIO64 to GPIO95 interrupt status register** These registers are used to record the GPIO current interrupt status.
100006A0	GINT_EDGE_0	32	**GPIO0 to GPIO31 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.
100006A4	GINT_EDGE_1	32	**GPIO32 to GPIO63 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.
100006A8	GINT_EDGE_2	32	**GPIO64 to GPIO95 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.

10000600 GPIO_CTRL_0 GPIO0 to GPIO31 direction control register 00000000

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

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

10000604 GPIO_CTRL_1 GPIO32 to GPIO63 direction control register 00000000

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

Bit(s)	Name	Description

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

10000608 GPIO_CTRL_2 GPIO64 to GPIO95 direction control register 00000000

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

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

10000610 GPIO_POL_0 GPIO0 to GPIO31 polarity control register 00000000

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

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

10000614 GPIO_POL_1 GPIO32 to GPIO63 polarity control register 00000000

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

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

10000618 **GPIO_POL_2** **GPIO64 to GPIO95 polarity control register** **00000000**

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

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

10000620 **GPIO_DATA_0** **GPIO0 to GPIO31 data register** **00000000**

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

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

10000624 **GPIO_DATA_1** **GPIO32 to GPIO63 data register** **00000000**

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

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

10000628 **GPIO_DATA_2** **GPIO64 to GPIO95 data register** **00000000**

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

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

10000630 **GPIO_DSET_0** **GPIO0 to GPIO31 data set register** **FFFFFFFF**

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

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

10000634 **GPIO_DSET_1** **GPIO32 to GPIO63 data set register** **FFFFFFFF**

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

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

10000638 **GPIO_DSET_2** **GPIO64 to GPIO95 data set register** **FFFFFFFF**

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

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

10000640 **GPIO_DCLR_0** **GPIO0 to GPIO31 data clear register** **FFFFFFFF**

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

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

10000644 **GPIO_DCLR_1** **GPIO32 to GPIO63 data clear register** **FFFFFFFF**

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

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

10000648 **GPIO_DCLR_2** **GPIO64 to GPIO95 data clear register** **FFFFFFFF**

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

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

10000650 **GINT_REDGE_0** **GPIO0 to GPIO31 rising edge interrupt enable register** **00000000**

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

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

10000654 **GINT_REDGE_1** **GPIO32 to GPIO63 rising edge interrupt enable register** **00000000**

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

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

10000658 **GINT_REDGE_2** **GPIO64 to GPIO95 rising edge interrupt enable register** **00000000**

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

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

10000660 **GINT_FEDGE_0** **GPIO0 to GPIO31 falling edge interrupt enable register** **00000000**

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

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

10000664 **GINT_FEDGE_1** **GPIO32 to GPIO63 falling edge interrupt enable register** **00000000**

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

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

10000668 **GINT_FEDGE_2** **GPIO64 to GPIO95 falling edge interrupt enable register** **00000000**

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

Bit(s)	Name	Description
31:0	GINTFEDGE2	**GPIO Falling Edge Interrupt Enable** 1: Enable falling edge triggered 0: Disable falling edge triggered

10000670 **GINT_HLVL_0** **GPIO0 to GPIO31 high level interrupt enable register** **00000000**

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

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

10000674 **GINT_HLVL_1** **GPIO32 to GPIO63 high level interrupt enable register** **00000000**

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

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

10000678 GINT_HLVL_2 GPIO64 to GPIO95 high level interrupt enable register 00000000

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

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

10000680 GINT_LLVL_0 GPIO0 to GPIO31 low level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTLLVL0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTLLVL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

10000684 GINT_LLVL_1 GPIO32 to GPIO63 low level interrupt enable register 00000000

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

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

10000688 GINT_LLVL_2 GPIO64 to GPIO95 low level interrupt enable register 00000000

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

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

10000690 GINT_STAT_0 GPIO0 to GPIO31 interrupt status register 00000000

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

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

10000694 GINT_STAT_1 GPIO32 to GPIO63 interrupt status register 00000000

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

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

10000698 GINT_STAT_2 GPIO64 to GPIO95 interrupt status register 00000000

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

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

100006A0 GINT_EDGE_0 GPIO0 to GPIO31 edge status register 00000000

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

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

100006A4 GINT_EDGE_1 GPIO32 to GPIO63 edge status register 00000000

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

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

100006A8 **GINT_EDGE_2** **GPIO64 to GPIO95 edge status register** **00000000**

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

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

4.9 SPI Slave

4.9.1 SPI Slave Control

spis_intf Changes LOG

Revision	Date	Author	Change Log
0.1	2013/9/23	Kaiping Yen	Initialization

Module name: spis_intf Base address: (+0h)

Address	Name	Width	Register Function
00000000	REG00	32	SPI Slave Register 00
00000004	REG01	32	SPI Slave Register 01
00000008	REG02	32	SPI Slave Register 02
0000000C	REG03	32	SPI Slave Register 03
00000010	REG04	32	SPI Slave Register 04

00000000 REG00 SPI Slave Register 00 00000000

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

Bit(s)	Name	Description
31:0	bus_read_data	SPI Slave Register 00 for bus read data

00000004 REG01 SPI Slave Register 01 00000000

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

Bit(s)	Name	Description
31:0	bus_write_data	SPI Slave Register 01 for bus write data

00000008 **REG02** **SPI Slave Register 02** **00000000**

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

Bit(s)	Name	Description
31:0	bus_address	**SPI Slave Register 02 for bus address** This address must be physical address

0000000C **REG03** **SPI Slave Register 03** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**reg03_31_5[26:11]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**reg03_31_5[10:0]**											**bus_pb_rb_sel**	**reg03_3**	**bus_size**		**bus_r_w**
Type	RW											RW	RW	RW		RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:5	reg03_31_5	**reg03[31:5] reserved bit**
4	bus_pb_rb_sel	**Bus interface selection** 0: Bus transaction is asserted by Rbus master interface, can access DRAM and peripheral registers 1: Bus transaction is asserted by Pbus master interface, can peripheral registers only
3	reg03_3	**reg03[3] reserved bit**
2:1	bus_size	**Bus access size** 00: reserved 01: reserved 10: word (4bytes) 11: reserved
0	bus_r_w	**Bus access type** 0: read 1: write

00000010 **REG04** **SPI Slave Register 04** **00000000**

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

Bit(s)	Name	Description
0	bus_busy	**Bus (Internal Rbus/Pbus Master) interface status** 0: SPIS bus interface is idle for next access command 1: SPIS bus interface is busy

4.9.2 Regsiters

spis_pbslv Changes LOG

Revision	Date	Author	Change Log
0.1	**2013/9/23**	**Kaiping Yen**	**Initialization**

Module name: spis_pbslv Base address: (+10000700h)

Address	Name	Width	Register Function
10000700	SPIS_REG0	32	**SPI Slave Register 0**
10000704	SPIS_REG1	32	**SPI Slave Register 1**
10000708	SPIS_REG2	32	**SPI Slave Register 2**
1000070C	SPIS_REG3	32	**SPI Slave Register 3**
10000710	SPIS_REG4	32	**SPI Slave Register 4**
10000740	SPIS_CFG	32	**SPI Slave Configuration**

10000700 SPIS_REG0 SPI Slave Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spis_reg0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spis_reg0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	spis_reg0	**SPI Slave Register 0**

10000704 SPIS_REG1 SPI Slave Register 1 **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spis_reg1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spis_reg1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	spis_reg1	**SPI Slave Register 1**

10000708 SPIS_REG2 SPI Slave Register 2 **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spis_reg2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spis_reg2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	spis_reg2	**SPI Slave Register 2**

1000070C SPIS_REG3 SPI Slave Register 3 **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spis_reg3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spis_reg3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	spis_reg3	**SPI Slave Register 3**

10000710 SPIS_REG4 SPI Slave Register 4 **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spis_reg4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spis_reg4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	spis_reg4	**SPI Slave Register 4**

10000740 **SPIS_CFG** **SPI Slave Configuration** **00000000**

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

Bit(s)	Name	Description
1:0	spis_mode	**SPI slave clock polarity and phase configuration** 2'b00: CPOL=0, CPHA=0 2'b01: CPOL=0, CPHA=1 2'b10: CPOL=1, CPHA=0 2'b11: CPOL=1, CPHA=1

4.10 I²C Controller

4.10.1 Features

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

4.10.2 List of Registers

I2C Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/3	Evan Chou	Initialization

Module name: I2C Base address: (+10000900h)

Address	Name	Width	Register Function
10000908	SM0CFG0	32	SERIAL INTERFACE MASTER 0 CONFIG 0 REGISTER
10000910	SM0DOUT	32	SERIAL INTERFACE MASTER 0 DATAOUT REGISTER
10000914	SM0DIN	32	SERIAL INTERFACE MASTER 0 DATAIN REGISTER
10000918	SM0ST	32	SERIAL INTERFACE MASTER 0 STATUS REGISTER
1000091C	SM0AUTO	32	SERIAL INTERFACE MASTER 0 AUTO-MODE REGISTER
10000920	SM0CFG1	32	SERIAL INTERFACE MASTER 0 CONFIG 1 REGISTER
10000928	SM0CFG2	32	SERIAL INTERFACE MASTER 0 CONFIG 2 REGISTER
10000940	SM0CTL0	32	Serial interface master 0 control 0 register
10000944	SM0CTL1	32	Serial interface master 0 control 1 register
10000950	SM0D0	32	Serial interface master 0 data 0 register
10000954	SM0D1	32	Serial interface master 0 data 1 register
1000095C	PINTEN	32	Peripheral interrupt enable register
10000960	PINTST	32	Peripheral interrupt status register
10000964	PINTCL	32	Peripheral interrupt clear register

10000908 SM0CFG0 SERIAL INTERFACE MASTER 0 CONFIG 0 REGISTER 00000000

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

Bit(s)	Name	Description
6:0	SM0_DEVADDR	**Device address for transmission**

10000910 **SM0DOUT** **SERIAL INTERFACE MASTER 0 DATAOUT REGISTER** **00000000**

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

Bit(s)	Name	Description
7:0	SM0_DATAOUT	**Data out register for auto mode**

10000914 **SM0DIN** **SERIAL INTERFACE MASTER 0 DATAIN REGISTER** **00000000**

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

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

10000918 **SM0ST** **SERIAL INTERFACE MASTER 0 STATUS REGISTER** **00000002**

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

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

1000091C **SM0AUTO** **SERIAL INTERFACE MASTER 0 AUTO-MODE REGISTER** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**SM0_START_RW**
Type																RW
Reset																0

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

10000920 **SM0CFG1** **SERIAL INTERFACE MASTER 0 CONFIG 1 REGISTER** **00000000**

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

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

10000928 **SM0CFG2** **SERIAL INTERFACE MASTER 0 CONFIG 2 REGISTER** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**SM0_I**

																S_AUTOMODE
Type																RW
Reset																0

Bit(s)	Name	Description
0	SM0_IS_AUTOMODE	**Set 1 to configure auto mode**

10000940 SM0CTL0 Serial interface master 0 control 0 register 00008000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SM0_ODRAIN	RESV0			SM0_CLK_DIV											
Type	RW	RO			RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	SIF_VSYNC	RESV1	SM0_VSYNC_MODE		RESV2							SM0_CS_STATUS	SM0_SCL_STATE	SM0_SDA_STATE	SM0_EN	SM0_SCL_STRECH
Type	RO	RO	RW		RO							RO	RO	RO	RW	RW
Reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	SM0_ODRAIN	**Open-drain output configuration** 0: When SIF output is logic 1, the output is pulled high by outer devices. SIF output is open-drained. 1: When SIF output is logic 1, the output is pulled high by SIF master 0.
30:28	RESV0	
27:16	SM0_CLK_DIV	**SIF master 0 clock divide value** This is used to set the divider to generate expected SCL.
15	SIF_VSYNC	
14	RESV1	
13:12	SM0_VSYNC_MODE	**Restrict SIF master 0 trigger within VSYNC pulse** 00: Disable 01: Allow triggered in VSYNC pulse 10: Allow triggered at VSYNC rising edge
11:5	RESV2	
4	SM0_CS_STATUS	**Clock stretching status** 0: no clock stretching 1: clock stretching
3	SM0_SCL_STATE	**SCL value on the bus**
2	SM0_SDA_STATE	**SDA value on the bus**
1	SM0_EN	**SIF master 0 enable bit**

Bit(s)	Name	Description
		0: Disable SIF master 0. 1: Enable SIF master 0.
0	SM0_SCL_STRECH	**Clock stretching enable** 0: Not allow slaves hold SCL 1: Allow slaves hold SCL

10000944 SM0CTL1 Serial interface master 0 control 1 register 0000000 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									SM0_ACK							
Type									RO							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						SM0_PGLEN				SM0_MODE						SM0_TRI
Type						RW				RW						RW
Reset						0	0	0		0	0	0				0

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

10000950 SM0D0 Serial interface master 0 data 0 register FFFFFFFF

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

Bit(s)	Name	Description
31:24	SM0_DATA3	**Serial interface data byte 3**
23:16	SM0_DATA2	**Serial interface data byte 2**

Bit(s)	Name	Description
15:8	SM0_DATA1	**Serial interface data byte 1**
7:0	SM0_DATA0	**Serial interface data byte 0**

10000954 **SM0D1** **Serial interface master 0 data 1 register** **FFFFFFFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SM0_DATA7								SM0_DATA6							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SM0_DATA5								SM0_DATA4							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

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

1000095C **PINTEN** **Peripheral interrupt enable register** **00000000**

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

Bit(s)	Name	Description
0	SM0_INT_EN	**Serial interface master 0 interrupt enable**

10000960 **PINTST** **Peripheral interrupt status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SM0_INT_

																ST
Type																WS
Reset																0

Bit(s)	Name	Description
0	SM0_INT_ST	**Serial interface master 0 interrupt status**

10000964 **PINTCL** **Peripheral interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**SM0_INT_CL**
Type																W1C
Reset																0

Bit(s)	Name	Description
0	SM0_INT_CL	**Serial interface master 0 interrupt clear**

4.11 **I2S Controller**

4.11.1 Features

- I2S transmitter/receiver, which can be configured as master or slave.
- Supports 16-bit data, sampling rates of 8 kHz, 16 kHz, 22.05 kHz, 44.1 kHz, and 48 kHz
- Support stereo audio data transfer.
- 32-byte FIFO are available for data transmission.
- Supports GDMA access
- Supports 12 Mhz bit clock from external source (when in slave mode)

4.11.2 Block Diagram

The I^2S transmitter block diagram is shown as below.

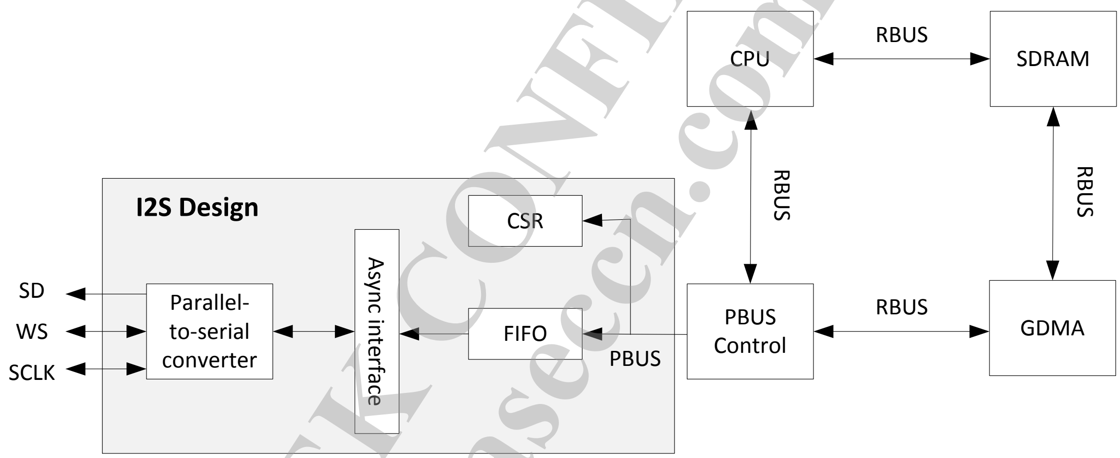


Figure 4-5 I^2S Transmitter Block Diagram

The I^2S interface consists of two separate cores, a transmitter and a receiver. Both can operate in either master or slave mode. The transmitter is only shown here in master or slave mode.

I^2S Signal Timing For I^2S Data Format

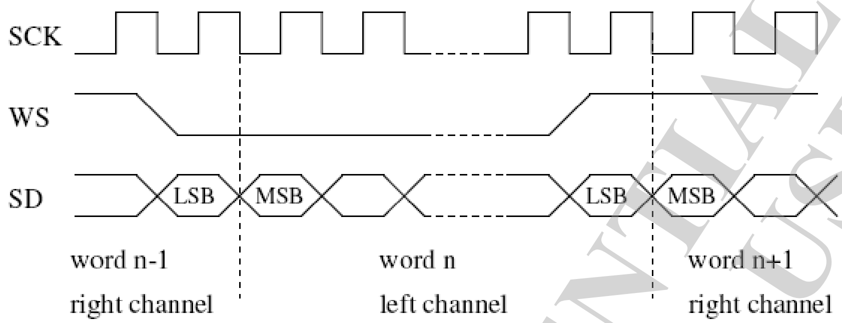


Figure 4-6 I2S Transmit/Receive

Serial data is transmitted in 2's complement with the MSB first. The transmitter always sends the MSB of the next word one clock period after the WS changes. Serial data sent by the transmitter may be synchronized with either the trailing (HIGH-to-LOW) or the leading (LOW-to-HIGH) edge of the clock signal. However, the serial data must be latched into the receiver on the leading edge of the serial clock signal, and so there are some restrictions when transmitting data that is synchronized with the leading edge.

The word select line indicates the channel being transmitted:

- WS = 0; channel 1 (left)
- WS = 1; channel 2 (right)

WS may change either on a trailing or leading edge of the serial clock, but it doesn't need to be symmetrical. In the slave, this signal is latched on the leading edge of the clock signal. The WS line changes one clock period before the MSB is transmitted. This allows the slave transmitter to derive synchronous timing of the serial data that will be set up for transmission. Furthermore, it enables the receiver to store the previous word and clear the input for the next Word.

4.11.3 Registers

I2S Changes LOG

Revision	Date	Author	Change Log
0.1	**2014/1/12**	**Ken Wu**	**Initialization**

Module name: I2S Base address: (+10000A00h)

Address	Name	Width	Register Function
10000A00	I2S_CFG	32	**I2S Configuration** I2S Tx/Rx Configuration Register
10000A04	INT_STATUS	32	**Interrupt Status** I2S Interrupt Status
10000A08	INT_EN	32	**Interrupt Enable** I2S Interrupt Enable Control Register
10000A0C	FF_STATUS	32	**FIFO Status** I2S Tx/Rx FIFO Status
10000A10	TX_FIFO_WRE	32	**Transmit FIFO Write to Register**

	G		Tx Write Data Buffer
10000A14	RX_FIFO_RREG	32	**Receive FIFO Read Register** DRAM PAD CONTROL 3
10000A18	I2S_CFG1	32	**I2S Configuration 1** I2S Loopback Test Control Register
10000A20	DIVCOMP_CFG	32	**Integer Part of the Dividor Register 1** Integer Part of the Dividor Register
10000A28	DIVINT_CFG	32	**Integer Part of the Dividor Register 2** Integer Part of the Dividor Register

10000A00 I2S_CFG I2S Configuration 00014040

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	I2S_EN	DMA_EN	LITTIE_ENDIAN_DATA_FMT	SYS_ENDIAN				TX_EN				RX_EN		NORM_24BIT	DATA_24BIT	SLAVE_MODE
Type	RW	RW	RW	RW				RW				RW		RW	RW	RW
Reset	0	0	0	0				0				0		0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_FF_THRES								TX_FF_THRES							
Type	RW								RW							
Reset	0	1	0	0					0	1	0	0				

Bit(s)	Name	Description
31	I2S_EN	**I2S Enable** Enables I2S. When disabled, all I2S control registers are cleared to their initial values. 0: Disable 1: Enable
30	DMA_EN	**DMA Enable** Enables DMA access. 0: Disable 1: Enable
29	LITTIE_ENDIAN_DATA_FMT	**Little endian audio data** 0: big endian audio data format 1: little endian audio data format
28	SYS_ENDIAN	**System endian setting.** 0: Little endian 1: Big endian
24	TX_EN	**Transmitter on/off control** 0: Disable 1: Enable
20	RX_EN	**Receiver on/off control** 0: Disable 1: Enable
18	NORM_24BIT	**24-bit data format**

Bit(s)	Name	Description
		0: compact data format 1: normal data format
17	DATA_24BIT	**I2S data width** 0: 16-bit data 1: 24-bit data
16	SLAVE_MODE	**Sets master or slave mode.** 0: Master: using internal clock 1: Slave: using external clock
15:12	RX_FF_THRES	**Rx FIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. 2<RX_FF_THRES<6 (unit: word)
7:4	TX_FF_THRES	**Tx FIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. 2<TX_FF_THRES<6 (unit: word)

10000A04 **INT_STATUS** **Interrupt Status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**RX_DMA_FAULT**	**RX_OVRUN**	**RX_UNRUN**	**RX_THRES**	**TX_DMA_FAULT**	**TX_OVRUN**	**TX_UNRUN**	**TX_THRES**
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	RX_DMA_FAULT	**Rx DMA Fault Detected Interrupt** Asserts when a fault is detected in Rx DMA signals.
6	RX_OVRUN	**Rx Overrun Interrupt** Asserts when the Rx FIFO is overrun.
5	RX_UNRUN	**Rx Underrun Interrupt** Asserts when the Rx FIFO is underrun.
4	RX_THRES	**Rx FIFO Below Threshold Interrupt** Asserts when the Rx FIFO is lower than the defined threshold.
3	TX_DMA_FAULT	**Tx DMA Fault Detected Interrupt** Asserts when a fault is detected in Tx DMA signals.
2	TX_OVRUN	**Tx FIFO Overrun Interrupt** Asserts when the Tx FIFO is overrun.
1	TX_UNRUN	**Tx FIFO Underrun Interrupt** Asserts when the Tx FIFO is underrun.
0	TX_THRES	**Tx FIFO Below Threshold Interrupt** Asserts when the FIFO is lower than the defined threshold.

10000A08 INT_EN Interrupt Enable 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RX_INT3_EN	RX_INT2_EN	RX_INT1_EN	RX_INT0_EN	TX_INT3_EN	TX_INT2_EN	TX_INT1_EN	TX_INT0_EN
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	RX_INT3_EN	**INT_STATUS[7] Enable** Enables the Rx DMA Fault Detected Interrupt. This interrupt asserts when a fault is detected in Rx DMA signals.
6	RX_INT2_EN	**INT_STATUS[6] Enable** Enables the Rx Overrun Interrupt. This interrupt asserts when the Rx FIFO is overrun.
5	RX_INT1_EN	**INT_STATUS[5] Enable** Enables the Rx Underrun Interrupt. This interrupt asserts when the Rx FIFO is underrun.
4	RX_INT0_EN	**INT_STATUS[4] Enable** Enables the Rx FIFO Below Threshold Interrupt. This interrupt asserts when the Rx FIFO is lower than the defined threshold.
3	TX_INT3_EN	**INT_STATUS[3] Enable** Enables the Tx DMA Fault Detected Interrupt. This interrupt asserts when a fault is detected in Tx DMA signals.
2	TX_INT2_EN	**INT_STATUS[2] Enable** Enables the Tx FIFO Overrun Interrupt. This interrupt asserts when the Tx FIFO is overrun.
1	TX_INT1_EN	**INT_STATUS[1] Enable** Enables the Tx FIFO Underrun Interrupt. This interrupt asserts when the Tx FIFO is underrun.
0	TX_INT0_EN	**INT_STATUS[0] Enable** Enables the Tx FIFO Below Threshold Interrupt. This interrupt asserts when the FIFO is lower than the defined threshold.

10000A0C FF_STATUS FIFO Status 00000010

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

Reset				0	0	0	0	0				1	0	0	0	0

Bit(s)	Name	Description
12:8	RX_AVCNT	**Rx FIFO Available Space Count** Counts the available space for reads in Rx FIFO. (unit: word)
4:0	TX_EPCNT	**Tx FIFO Available Space Count** Counts the available space for writes in Tx FIFO. (unit: word)

10000A10 **TX_FIFO_WREG** **Transmit FIFO Write to Register** **00000000**

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

Bit(s)	Name	Description
31:0	TX_FIFO_WDATA	**Tx FIFO Write Data Buffer** Buffers data to be written to the Tx FIFO.

10000A14 **RX_FIFO_RREG** **Receive FIFO Read Register** **00000000**

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

Bit(s)	Name	Description
31:0	RX_FIFO_RDATA	**Rx FIFO Read Data Buffer** Buffers data read from the Rx FIFO.

10000A18 **I2S_CFG1** **I2S Configuration 1** **00000000**

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

Reset	0	0														
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**I2S_FMT**
Type																RW
Reset																0

Bit(s)	Name	Description
31	LBK_EN	**Enables loopback mode.** 0: Normal mode 1: Loopback mode ASYNC_TXFIFIO -> Tx -> Rx -> ASYNC_RXFIFIO
30	EXT_LBK_EN	**Enables external loopback.** 0: Normal mode 1: Enables external loop back. External A/D -> Rx -> Tx -> External D/A
0	I2S_FMT	**I2S audio data format** 0: i2s mode 1: left-justified mode

10000A20 **DIVCOMP_CFG** **Integer Part of the Dividor Register 1** **00000000**

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

Bit(s)	Name	Description
31	CLK_EN	**Enables setting of the I2S clock based on DIVCOMP and DIVINT parameters.** 0: Disable 1: Enable
8:0	DIVCOMP	**A parameter in an equation which determines FREQOUT. See DIVINT_CFG.**

10000A28 **DIVINT_CFG** **Integer Part of the Dividor Register 2** **00000000**

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

Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9:0	DIVINT	**Integer Divider** A parameter in an equation which determines FREQOUT: FREQOUT = FREQIN *(1/2) * {1 / [DIVINT+DIVCOMP/(512)]} FREQIN is always fixed to 480 MHz.

4.12 **SPI Controller**

4.12.1 Features

- Supports up to 2 SPI master operations
- Programmable clock polarity
- Programmable interface clock rate
- Programmable bit ordering
- Firmware-controlled SPI enable
- Programmable payload (address + data) length
- Supports 1/2/4 multi-IO SPI flash memory
- Supports command/user mode operation
- Supports SPI direct access
- Extends the addressable range from 24 bits to 32 bits for memory size larger than 128 Mb.

4.12.2 Block Diagram

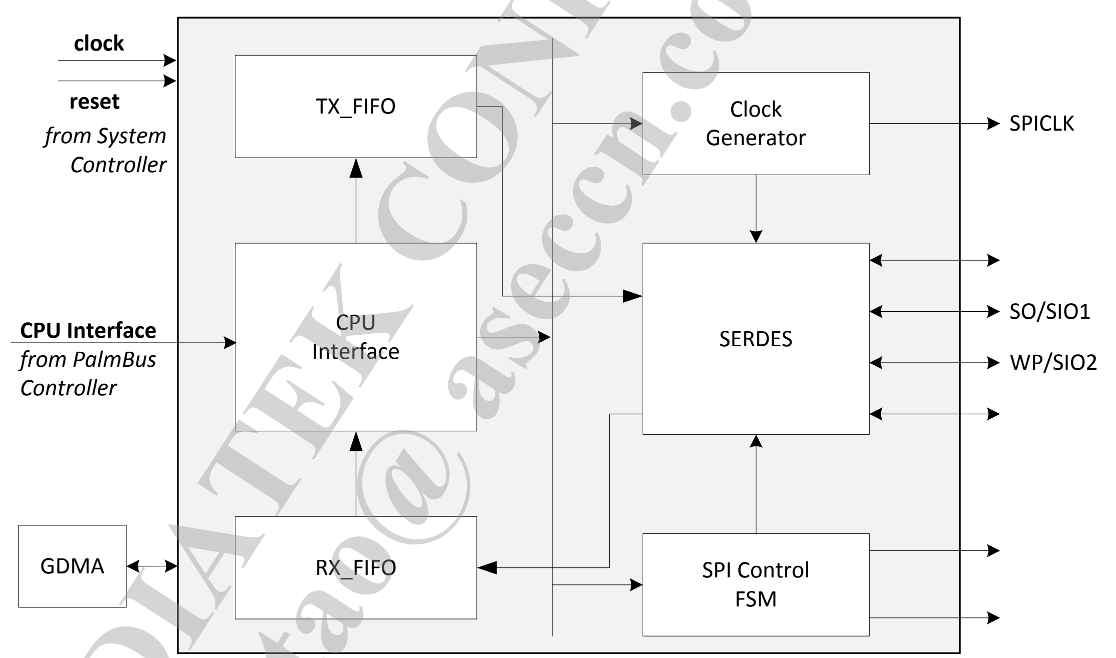


Figure 4-7 SPI Controller Block Diagram

4.12.3 Registers

SPI Changes LOG

Revision	Date	Author	Change Log

0.1	**2012/8/29**	**Lancelot**	**Initialization**
0.2	**2012/11/6**	**Lancelot**	**1. Remove 0x38 SW_RST 2. Add CS_POLAR at 0x38**
0.3	**2012/11/23**	**Lancelot**	**Fix default value**

Module name: SPI Base address: (+10000B00h)

Address	Name	Width	Register Function
10000B00	SPI_TRANS	32	SPI transaction control/status register
10000B04	SPI_OP_ADDR	32	SPI opcode/address register
10000B08	SPI_DIDO_0	32	SPI DI/DO data #0 register
10000B0C	SPI_DIDO_1	32	SPI DI/DO data #1 register
10000B10	SPI_DIDO_2	32	SPI DI/DO data #2 register
10000B14	SPI_DIDO_3	32	SPI DI/DO data #3 register
10000B18	SPI_DIDO_4	32	SPI DI/DO data #4 register
10000B1C	SPI_DIDO_5	32	SPI DI/DO data #5 register
10000B20	SPI_DIDO_6	32	SPI DI/DO data #6 register
10000B24	SPI_DIDO_7	32	SPI DI/DO data #7 register
10000B28	SPI_MASTER	32	SPI master mode register
10000B2C	SPI_MORE_BUF	32	SPI more buf control register
10000B30	SPI_QUEUE_CTL	32	SPI flash queue control register
10000B34	SPI_STATUS	32	SPI controller status register
10000B38	SPI_CS_POLAR	32	SPI chip select polarity
10000B3C	SPI_SPACE	32	SPI flash space control register

10000B00 SPI_TRANS SPI transaction control/status register 00160001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spi_addr_ext								Reserved0			spi_addr_size		Reserved1		spi_master_busy
Type	RW								RO			RW		RO		RO
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	Reserved2							spi_master_start	miso_byte_cnt				mosi_byte_cnt			
Type	RO							WO	RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description

Bit(s)	Name	Description
31:24	spi_addr_ext	**SPI address extention** Address extenstion for 32-bit SPI address size. Usually this field specifies the first byte of the address phase to transmit to SPI device when more_buf_mode = 0 and spi_addr_size = 3. And spi_addr[31:24], spi_addr[23:16], and spi_addr[15:0] are respectively the second, third and fourth byte of the address phase
20:19	spi_addr_size	**SPI address size.** 0: reserved. 1: spi_addr[15:0] of SPI DI data register are valid (16-bit size). 2: spi_addr[23:0] of SPI DI data register are valid (24-bit size). 3: {spi_addr_ext[7:0], spi_addr[23:0]} of SPI DI data register are valid (32-bit size) Note: The spi_addr_size is valid only when more_buf_mode = 0.
16	spi_master_busy	**Transaction busy indication (Read-only). Writes to this bit are ignored.** 0: No SPI transaction is ongoing. Software may start a new SPI transaction by writing to the SPI transaction start bit within this register. 1: An SPI transaction presently is underway. Software must not try to start a new SPI transaction. Software may not alter the value of any field of the SPI master control registers.
8	spi_master_start	**SPI transaction start. Only writes to this field are meaningful, reads always return 0.** Writes: 0: No effect 1: Starts SPI transaction.
7:4	miso_byte_cnt	**SPI MISO (rx) byte count.** Determines the number of bytes received from the SPI device from the SPI opcode/address register and the SPI DI/DO data #0 register. Values of 0 ~ 8 are valid, other values are illegal. Note: The miso_byte_cnt is valid only when more_buf_mode = 0.
3:0	mosi_byte_cnt	**SPI MOSI (tx) byte count.** Determines the number of bytes transmitted from the SPI opcode/address register and the SPI DI/DO data #0 register to the SPI device. Values of 1 ~ 8 are valid, other values are illegal. Note: The mosi_byte_cnt is valid only when more_buf_mode = 0. The transmitted data sequence is as follows: spi_opcode, spi_addr (conditional) and d0_byte ~ d3_byte (conditional).

10000B04 <u>**SPI_OP_ADDR**</u> **SPI opcode/address register** **00000000**

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

Bit(s)	Name	Description
31:8	spi_addr	**SPI address. Usually this field specifies the 24-bits address to transmit to the SPI device when more_buf_mode = 0.** 1: (16-bits SPI address size), spi_addr[23:16] is the 1st byte of the address

Bit(s)	Name	Description
		phase and spi_addr[15:8] is the 2nd byte of the address phase. 2: (24-bits SPI address size), spi_addr[31:24] is the 1st byte of the address phase and spi_addr[23:16] is the 2nd byte of the address phase and spi_addr[15:8] is the 3rd byte of the address phase. 3: (32-bits SPI address size), spi_addr[31:24] is the 2nd byte of the address phase and spi_addr[23:16] is the 3rd byte of the address phase and spi_addr[15:8] is the 4th byte of the address phase Note: For SPI read transaction and more_buf_mode = 0 Field [15:8] is also used to store the 6-th byte of data read phase. Field [23:16] is also used to store the 7-th byte of data read phase. Field [31:24] is also used to store the 8-th byte of data read phase.
7:0	spi_opcode	**SPI opcode. Usually this field specifies the 8-bits opcode (instruction) to transmit to the SPI device as the first byte of a SPI transaction when more_buf_mode = 0.** Note: For SPI read transaction and more_buf_mode = 0, this byte is also used to store the 5-th byte of data read phase according to the rx byte count miso_byte_cnt.

10000B08 SPI_DIDO_0 SPI DI/DO data #0 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B0C SPI_DIDO_1 SPI DI/DO data #1 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**

Bit(s)	Name	Description
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B10 **SPI_DIDO_2** **SPI DI/DO data #2 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B14 **SPI_DIDO_3** **SPI DI/DO data #3 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B18 **SPI_DIDO_4** **SPI DI/DO data #4 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							

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

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B1C **SPI_DIDO_5** **SPI DI/DO data #5 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**d3_byte**								**d2_byte**							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**d1_byte**								**d0_byte**							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B20 **SPI_DIDO_6** **SPI DI/DO data #6 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**d3_byte**								**d2_byte**							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**d1_byte**								**d0_byte**							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B24 **SPI_DIDO_7** **SPI DI/DO data #7 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

10000B28 **<u>SPI_MASTER</u>** **SPI master mode register** **000D8880**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rs_slave_sel			clk_mode	rs_clk_sel											
Type	RW			RW	RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cs_dsel_cnt					full_duplex	int_en	spi_start_sel	spi_prefetch	bidir_mode	cpha	cpol	lsb_first	more_buf_mode	serial_mode	
Type	RW					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	
Reset	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:29	rs_slave_sel	**select SPI device** 0: select SPI device 0 (default is flash) 1: select SPI device 1 ... 7: select SPI device 7
28	clk_mode	**This register is used to specify that period of SCLK HIGH is longer or period of SCLK LOW is longer when clock divisor(clk_sel) is odd.** 0: period of SCLK LOW is longer. 1: period of SCLL HIGH is longer.
27:16	rs_clk_sel	**Register Space SPI clock frequency select.** 0: SPI clock frequency is hclk/2. (50% duty cycle, duty cycle is the ratio of the output high time to the total cycle time) 1: SPI clock frequency is hclk/3. (33.33% or 66.67% duty cycle) 2: SPI clock frequency is hclk/4. (50% duty cycle) 3: SPI clock frequency is hclk/5. (40% or 60% duty cycle) 4095: SPI clock frequency is hclk/4097.
15:11	cs_dsel_cnt	**De-select time of SPI chip select is configured to occupy the number of cycles of AHB clock**
10	full_duplex	**Full duplex or half duplex mode.** 0: half duplex mode.

Bit(s)	Name	Description
		1: full duplex mode. Full duplex timing diagram Note: The full_duplex is valid only when more_buf_mode = 1. The transmission is always as half duplex when more_buf_mode = 0;
9	int_en	**Interrupt enable.** 0: disable SPI interrupt. 1: enable SPI interrupt.
8	spi_start_sel	**The interval between spi_cs_n and spi_sclk.** 0: 3 clk 1: 6 clk
7	spi_prefetch	**SPI pre-fetch buffer enable** 0: disable pre-fetch buffer. 1: enable pre-fetch buffer.
6	bidir_mode	**Bi-direction mode. In this mode, the SPI uses only one serial data pin for interface with external devices. The MOSI pin becomes the serial data I/O pin for the SPI transaction and MISO pin is not used. Bi-direction mode is used for the application with only 1 bi-direction serial pin for SPI transaction.** 0: normal mode (both MOSI and MISO pins are used). 1: bi-direction mode (only MOSI pin is used). SPI host controller must operate in half duplex mode if bidir_mode = 1. Note: The bidir_mode is valid only when more_buf_mode = 1.
5	cpha	**(CPHA, clock phase). Initial SPI clock phase for SPI transaction.** There are four SPI modes used to latch data. These SPI modes latch data in one of four ways, and are defined by the logic state combinations of the CLK Polarity (CPOL) in relation to the CLK Phase (CPHA). The valid logic combinations identify and determine the SPI modes supported by the SPI device. SPI mode At CPOL=0 the base value of the clock is zero For CPHA=0 (mode 0), data is read on the clock's rising edge and data is changed on a falling edge. For CPHA=1 (mode 1), data is read on the clock's falling edge and data is changed on a rising edge. At CPOL=1 the base value of the clock is one (inversion of CPOL=0) For CPHA=0 (mode 2), data is read on clock's falling edge and data is changed on a rising edge. For CPHA=1 (mode 3), data is read on clock's rising edge and data is changed on a falling edge.
4	cpol	**cpol (CPOL, clock polarity). Initial SPI clock polarity for SPI transaction.**
3	lsb_first	**0: MSB(most significant bit) is transferred first for SPI transaction.** 1: LSB(least significant bit) is transferred first for SPI transaction.
2	more_buf_mode	**Select 2 words buffer or 8 words buffer for SPI transaction.** 0: SPI transfer data buffer size is only 2 words. In this mode, SPI DI/DO data #0 register and SPI opcode/address register are the data buffer for SPI transaction. And, SPI master follows mosi_byte_cnt and miso_byte_cnt to complete the transmission and reception, respectively. This kind of transaction must operate in half duplex mode. 1: SPI transfer data buffer size is 8 words. In this mode, SPI opcode/address register are the data buffer for SPI transaction and follows cmd_bit_cnt to complete the transaction. SPI DI/DO data #0~#7 register are the data buffer for SPI transaction and follows do_bit_cnt and di_bit_cnt to complete the transmission and reception, respectively. In half duplex mode, transmitted

Bit(s)	Name	Description
		data are loaded from SPI opcode/address register and SPI DI/DO data #0~#7 registers. And, the received data will overwrite the SPI DI/DO data #0~#7 registers. In full duplex mode, SPI DI/DO data #0~#3 registers are used for transmission and SPI DI/DO #4~#7 registers are used for receipt.
1:0	serial_mode	**This mode is designed for Winbond SPI flash W25Q80/16/32 and W25X10/20/40/80/16/32/64 series.** 0: standard serial. 1: dual serial. 2: quad serial. 3: reserved. Note: The serial_mode is valid only when more_buf_mode = 0. The transaction mode is always as standard serial when more_buf_mode = 1.

10000B2C **SPI_MORE_BUF** **SPI more buf control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved0		cmd_bit_cnt						Reserved1			miso_bit_cnt[8:4]				
Type	RO		RW						RO			RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	miso_bit_cnt[3:0]				Reserved2			mosi_bit_cnt								
Type	RW				RO			RW								
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
29:24	cmd_bit_cnt	**SPI command phase MOSI (tx) bit count. Determines the number of command bits transmitted from the SPI opcode/address register to the SPI device. Values of 0 ~ 32 are valid, but other values are illegal.** Note: The cmd_bit_cnt is valid only when more_buf_mode = 1 and the SPI opcode/address register is treated as a command register.
20:12	miso_bit_cnt	**SPI data phase MISO (rx) bit count. Determines the number of bits received from the SPI device into the SPI DI/DO data #0~#7 register. Values of 0 ~ 256 are valid, but other values are illegal. Maximum value is 256 for half duplex mode and 128 for full duplex mode. Please note that do_bit_cnt must be equal to di_bit_cnt in full duplex mode.** Note: The miso_bit_cnt is valid only when more_buf_mode = 1.
8:0	mosi_bit_cnt	**SPI data phase MOSI (tx) bit count. Determines the number of data bits transmitted from the SPI DI/DO data #0~#7 register to the SPI device. Values of 0 ~ 256 are valid, but other values are illegal. Maximum value is 256 for half duplex mode and 128 for full duplex mode.** Note: The mosi_bit_cnt is valid only when more_buf_mode = 1.

10000B30 **SPI_QUEUE_CTL** **SPI flash queue control register** **00000E40**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	fs_page_sel						Reserved0[12:3]									
Type	RW						RO									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	Reserved0[2:0]			fs_	fs_addr_si		fs_addr_si		fs_di_ph_byc				Res	fast_spi_sel		

				bus y	ze_r		ze						erv ed1			
Type	RO			RO	RO		RW		RW				RO	RW		
Reset	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0	0

Bit(s)	Name	Description
31:26	fs_page_sel	**Flash Space Page Selection.** 0: (Page 0 space) 0x0000_0000 - 0x03ff_ffff 1: (Page 1 space) 0x0400_0000 - 0x07ff_ffff ... 63: (Page 63 space) 0xffc0_0000 - 0xffff_ffff
12	fs_busy	**Transaction busy indication (Read-only) in flash space. Writes to this bit are ignored.** 0: No SPI flash space access is ongoing. Software may change the configuration related to flash space. 1: SPI flash space access presently is underway. Software may not alter the configuration related to flash space.
11:10	fs_addr_size_r	**Latched fs_addr_size indication from internal spimc logic**
9:8	fs_addr_size	**SPI address. This field specifies the 24-bits/16-bits address to transmit to the SPI device for SPI Flash Space Read operation only.** 0: 25-bit SPI address size 1: 16-bit SPI address size Reserved. 2: 24-bit SPI address size (default for 3B SPI flash) 3: 26-bit SPI address size (default for 4B SPI flash) If the change of the fs_addr_size is needed, the sequence below must be followed. Otherwise, the new fs_addr_size configuration will not be updated to the internal spimc logic . Step 1: Set new fs_addr_size. Step 2: Transmit mode change command (ex. En4B or Ex4B of MX25L25635E) Note: 1. The value fs_addr_size is not valid in Register Space. 2. The Spimc now only supports 3-Byte mode (24 bits) and 4-Byte mode (25 or 26 bits) switch.
7:4	fs_di_ph_byc	**Determines the number of data bytes transmitted from the SPI master controller to the SPI device for SPI Flash Space Read operation. This field is similar to mosi_byte_cnt in STCSR but is used for setting of flash space access control path.** Note: this field should (if fs_addr_size_r = 2, 24-bit fs_addr_size) = 4 (OP + ADDR) if fast_spi_sel = 0 (0x03) = 5 (OP + ADDR + dummy) if fast_spi_sel = 1 (0x0b) = 5 (OP + ADDR + dummy) if fast_spi_sel = 2 (0x3b) = 5 (OP + ADDR + M7-0) if fast_spi_sel = 3 (0xbb) = 5 (OP + ADDR + dummy) if fast_spi_sel = 4 (0x6b) = 7 (OP + ADDR + M7-0 + dummy) if fast_spi_sel = 5 (0xeb) = 5 (OP + ADDR + M7-0) if fast_spi_sel = 6 (0xe3) (if fs_addr_size_r = 0 or 3, 25 or 26-bit fs_addr_size) = 5 (OP + ADDR) if fast_spi_sel = 0 (0x03) = 6 (OP + ADDR + dummy) if fast_spi_sel = 1 (0x0b) = 6 (OP + ADDR + dummy) if fast_spi_sel = 2 (0x3b) = 6 (OP + ADDR + M7-0) if fast_spi_sel = 3 (0xbb) = 6 (OP + ADDR + dummy) if fast_spi_sel = 4 (0x6b) = 8 (OP + ADDR + M7-0 + dummy) if fast_spi_sel = 5 (0xeb) = 6 (OP + ADDR + M7-0) if fast_spi_sel = 6 (0xe3)
2:0	fast_spi_sel	**Select SPI flash read instruction for Flash Space** 0: standard read data instruction (0x03).

Bit(s)	Name	Description
		1: standard fast read data instruction (0x0b). 2: fast read dual output instruction defined in Winbond W25Qxx series SPI flash (0x03b). 3: fast read dual I/O instruction defined in Winbond W25Qxx series SPI flash (0xbb). 4: fast read quad output instruction defined in Winbond W25Qxx series SPI flash (0x6b). 5: fast read quad I/O instruction defined in Winbond W25Qxx series SPI flash (0xeb). 6: burst read quad I/O instruction defined in Winbond W25Qxx series SPI flash (0xe3). Note: serial_mode and more_buf_mode are don't care for this flash space access control path.

10000B34 SPI_STATUS SPI controller status register 00000030

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved0[25:10]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved0[9:0]										spi_flash_mode		Reserved1			spi_ok
Type	RO										RO		RO			RC
Reset	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

Bit(s)	Name	Description
5:4	spi_flash_mode	**0: no SPI flash.** 1: standard SPI flash. 2: specific SPI flash with dual interface capability. 3: specific SPI flash with quad interface capability.
0	spi_ok	**When SPI transaction complete, SPI master controller will set this bit and assert SPI interrupt to notify software. Reading this register will clear this bit and de-assert SPI interrupt.**

10000B38 SPI_CS_POLAR SPI chip select polarity 00000000

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

Bit(s)	Name	Description
7:0	cs_polar	**Chip select default polarity** set cs_polar[n]=1'b0 for cs[n] low active (SPI Flash) set cs_polar[n]=1'b1 for cs[n] high active

10000B3C **SPI_SPACE** **SPI flash space control register** **00000030**

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

Bit(s)	Name	Description
14:12	fs_slave_sel	**(Flash Space Slave Select)** 0: select SPI device #0. (default is flash) 1: select SPI device #1. ... 7: select SPI device #7.
11:0	fs_clk_sel	**Flash Space SPI clock frequency select.** 0: SPI clock frequency is hclk/2. (50% duty cycle, duty cycle is the ratio of the output high time to the total cycle time) 1: SPI clock frequency is hclk/3. (33.33% or 66.67% duty cycle) 2: SPI clock frequency is hclk/4. (50% duty cycle) 3: SPI clock frequency is hclk/5. (40% or 60% duty cycle) 4095: SPI clock frequency is hclk/4097.

4.13 UART Lite

4.13.1 Features

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

4.13.2 Registers

n = 1; for uart1 only.

UARTn+0000h RX Buffer Register — UARTn_RBR

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

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

UARTn+0000h TX Holding Register — UARTn_THR

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

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

UARTn+0004h Interrupt Enable Register — UARTn_IER

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

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

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

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

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

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

ELSI When set ("1"), an interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.
- **0** No interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.
- **1** An interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.

ETBEI When set ("1"), an interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO
have been reduced to its Trigger Level.
- **0** No interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO have been reduced to its Trigger Level.
- **1** An interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO have been reduced to its Trigger Level

ERBFI When set ("1"), an interrupt is generated if the RX Buffer contains data.
- **0** No interrupt is generated if the RX Buffer contains data.
- **1** An interrupt is generated if the RX Buffer contains data.

UARTn+0008h Interrupt Identification Register UARTn_IIR

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

IIR Identify if there are pending interrupts; ID4 and ID3 are presented only when EFR[4] = 1.
The following table gives the IIR[5:0] codes associated with the possible interrupts:

IIR[5:0]	Priority Level	Interrupt	Source
000001	-	No interrupt pending	
000110	1	Line Status Interrupt	BI, FE, PE or OE set in LSR

000100	2	RX Data Received	RX Data received or RX Trigger Level reached.
001100	2	RX Data Timeout	Timeout on character in RX FIFO.
000010	3	TX Holding Register Empty	TX Holding Register empty or TX FIFO Trigger Level reached.
000000	4	Modem Status change	DDCD, TERI, DDSR or DCTS set in MSR
010000	5	Software Flow Control	XOFF Character received
100000	6	Hardware Flow Control	CTS or RTS Rising Edge

Table 4-1 The IIR[5:0] codes associated with the possible interrupts

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

1. FIFO contains at least one character;
2. The most recent character was received longer than four character periods ago (including all start, parity and stop bits);
3. The most recent CPU read of the FIFO was longer than four character periods ago.

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

1. FIFO is empty;
2. The most recent character was received longer than four character periods ago (including all start, parity and stop bits);
3. The most recent CPU read of the FIFO was longer than four character periods ago.

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

UARTn+0008h FIFO Control Register UARTn_FCR

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

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

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

FCR[5:4] TX FIFO trigger threshold
- **0** 1
- **1** 4
- **2** 8
- **3** 14 (FIFOSIZE - 2)

DMA1 This bit determines the DMA mode, which the TXRDY and RXRDY pins support. TXRDY and RXRDY act to support single-byte transfers between the UART and memory (DMA mode 0) or multiple byte transfers (DMA mode1). Note that this bit has no effect unless the FIFOE bit is set as well
- **0** The device operates in DMA Mode 0.
- **1** The device operates in DMA Mode 1.

TXRDY – mode0: Goes active (low) when the TX FIFO or the TX Holding Register is empty. Becomes inactive when a byte is written to the Transmit channel.
TXRDY – mode1: Goes active (low) when there are no characters in the TX FIFO. Becomes inactive when the TX FIFO is full.
RXRDY – mode0: Becomes active (low) when at least one character is in the RX FIFO or the RX Buffer Register is full. Becomes inactive when there are no more characters in the RX FIFO or RX Buffer register.
RXRDY – mode1: Becomes active (low) when the RX FIFO Trigger Level is reached or an RX FIFO Character Timeout occurs. Goes inactive when the RX FIFO is empty.

CLRT Clear Transmit FIFO. This bit is self-clearing.
- **0** Leave TX FIFO intact.
- **1** Clear all the bytes in the TX FIFO.

CLRR Clear Receive FIFO. This bit is self-clearing.
- **0** Leave RX FIFO intact.
- **1** Clear all the bytes in the RX FIFO.

FIFOE FIFO Enabled. This bit must be set to 1 for any of the other bits in the registers to have any effect.
- **0** Disable both the RX and TX FIFOs.
- **1** Enable both the RX and TX FIFOs.

UARTn+000Ch Line Control Register — UARTn_LCR

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

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

DLAB Divisor Latch Access Bit.
- **0** The RX and TX Registers are read/written at Address 0 and the IER register is read/written at Address 4.
- **1** The Divisor Latch LS is read/written at Address 0 and the Divisor Latch MS is read/written at Address 4.

SB Set Break
- **0** No effect
- **1** SOUT signal is forced into the “0” state.

SP Stick Parity
- **0** No effect.
- **1** The Parity bit is forced into a defined state, depending on the states of EPS and PEN:
  If EPS=1 & PEN=1, the Parity bit is set and checked = 0.
  If EPS=0 & PEN=1, the Parity bit is set and checked = 1.

EPS Even Parity Select
- **0** When EPS=0, an odd number of ones is sent and checked.
- **1** When EPS=1, an even number of ones is sent and checked.

PEN Parity Enable
- **0** The Parity is neither transmitted nor checked.
- **1** The Parity is transmitted and checked.

STB Number of STOP bits
- **0** One STOP bit is always added.
- **1** Two STOP bits are added after each character is sent; unless the character length is 5 when 1 STOP bit is added.

WLS1, 0 Word Length Select.
- **0** 5 bits
- **1** 6 bits
- **2** 7 bits
- **3** 8 bits

UARTn+0010h Modem Control Register — UARTn_MCR

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

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

XOFF Status This is a read-only bit.
- **0** When an XON character is received.
- **1** When an XOFF character is received.

DCM_EN UART DCM function enable bit
- **0** UART DCM is disabled.
- **1** UART DCM is enabled.

OUT2 Controls the state of the output NOUT2, even in loop mode.
- **0** NOUT2=1.
- **1** NOUT2=0.

OUT1 Controls the state of the output NOUT1, even in loop mode.
- **0** NOUT1=1.
- **1** NOUT1=0.

RTS Controls the state of the output NRTS, even in loop mode.
- **0** NRTS=1.
- **1** NRTS=0.

DTR Control the state of the output NDTR, even in loop mode.
- **0** NDTR=1.
- **1** NDTR=0.

UARTn+0014h Line Status Register UARTn_LSR

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

LSR Line Status Register.
Modified when LCR[7] = 0.

FIFOERR RX FIFO Error Indicator.
- **0** No PE, FE, BI set in the RX FIFO.
- **1** Set to 1 when there is at least one PE, FE or BI in the RX FIFO.

TEMT TX Holding Register (or TX FIFO) and the TX Shift Register are empty.
- **0** Empty conditions below are not met.
- **1** If FIFOs are enabled, the bit is set whenever the TX FIFO and the TX Shift Register are empty. If FIFOs are disabled, the bit is set whenever TX Holding Register and TX Shift Register are empty.

THRE Indicates if there is room for TX Holding Register or TX FIFO is reduced to its Trigger Level.
- **0** **Reset whenever the contents of the TX FIFO are more than its Trigger Level (FIFOs are enabled), or whenever TX Holding Register is not empty(FIFOs are disabled).**
- **1** Set whenever the contents of the TX FIFO are reduced to its Trigger Level (FIFOs are enabled), or whenever TX Holding Register is empty and ready to accept new data (FIFOs are disabled).

BI Break Interrupt.

0 Reset by the CPU reading this register

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

FE Framing Error.

0 Reset by the CPU reading this register

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

PE Parity Error

0 Reset by the CPU reading this register

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

OE Overrun Error.

0 Reset by the CPU reading this register.

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

DR Data Ready.

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

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

UARTn+0018h Modem Status Register UARTn_MSR

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

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

MSR Modem Status Register

DCD Data Carry Detect.

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

RI Ring Indicator.

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

DSR Data Set Ready

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

CTS Clear To Send.

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

DDCD Delta Data Carry Detect.

- **0** The state of DCD has not changed since the Modem Status Register was last read
- **1** Set if the state of DCD has changed since the Modem Status Register was last read.

TERI Trailing Edge Ring Indicator

- **0** The NRI input does not change since this register was last read.
- **1** Set if the NRI input changes from "0" to "1" since this register was last read.

DDSR Delta Data Set Ready

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

DCTS Delta Clear To Send

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

UARTn+001Ch Scratch Register UARTn_SCR

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

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

UARTn+0000h Divisor Latch (LS) UARTn_DLL

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

UARTn+0004h Divisor Latch (MS) UARTn_DLM

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

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

BAUD	13MHz	26MHz	52MHz
110	7386	14773	29545
300	2708	5417	10833
1200	677	1354	2708

2400	338	677	1354
4800	169	339	677
9600	85	169	339
19200	42	85	169
38400	21	42	85
57600	14	28	56
115200	6	14	28

Table 4-2 Divisor needed to generate a given baud rate

UARTn+0008h Enhanced Feature Register UARTn_EFR

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

*NOTE: Only when LCR=BF'h

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

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

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

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

UARTn+0010h XON1 UARTn_XON1

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

UARTn+0014h XON2 — UARTn_XON2

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

UARTn+0018h XOFF1 — UARTn_XOFF1

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

UARTn+001Ch XOFF2 — UARTn_XOFF2

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

UARTn+0024h HIGH SPEED UART — UARTn_HIGHSPEED

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

SPEED UART sample counter base

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

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

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	7386	14773	29545	7386 * 16
300	2708	7386	14773	2708 * 16
1200	677	2708	7386	677 * 16
2400	338	677	2708	338 * 16
4800	169	338	677	169 * 16
9600	85	169	338	85 * 16
19200	42	85	169	9 * 75

38400	21	42	85	13 * 26
57600	14	21	42	8 * 28
115200	7	14	21	4 * 28
230400	*	7	14	2 * 28
460800	*	*	7	1 * 28
921600	*	*	*	1 * 14

Table 4-3 Divisor needed to generate a given baud rate from 13MHz based on different HIGHSPEED value

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

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	14773	29545	59091	7386 * 32
300	5417	14773	29545	2708 * 32
1200	1354	5417	14773	677 * 32
2400	677	1354	5417	338 * 32
4800	339	677	1354	169 * 32
9600	169	339	667	85 * 32
19200	85	169	339	18 * 75
38400	42	85	169	26 * 26
57600	28	42	85	16 * 28
115200	14	28	42	8 * 28
230400	7	14	28	4 * 28
460800	*	7	14	2 * 28
921600	*	*	7	1 * 28

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

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

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	29545	59091	118182	14773 * 32
300	10833	29545	59091	5417 * 32
1200	2708	10833	29545	1354 * 32
2400	1354	2708	10833	667 * 32
4800	677	1354	2708	339 * 32
9600	339	677	1354	169 * 32
19200	169	339	677	36 * 75
38400	85	169	339	52 * 26

57600	56	85	169	32 * 28
115200	28	56	85	16 * 28
230400	14	28	56	8 * 28
460800	7	14	28	4 * 28
921600	*	7	14	2 * 28

Table 4-5 Divisor needed to generate a given baud rate from 52 MHz based on different HIGHSPEED value

UARTn+0028h SAMPLE_COUNT — UARTn_SAMPLE_COUNT

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

When HIGHSPEED=3, the sample_count is the threshold value for UART sample counter (sample_num).
Count from 0 to sample_count.

UARTn+002Ch SAMPLE_POINT — UARTn_SAMPLE_POINT

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

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

UARTn+0034h Rate Fix Address — UARTn_RATEFIX_AD

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

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

UARTn+003Ch Guard time added register — UARTn_GUARD

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

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

GUARD_EN Guard interval add enable signal.

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

UARTn+0040h Escape character register — UARTn_ESCAPE_DAT

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

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

UARTn+0044h Escape enable register — UARTn_ESCAPE_EN

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

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

- **0** Do not deal with the escape character.
- **1** Add escape character in transmitter and remove escape character in receiver.

UARTn+0048h Sleep enable register — UARTn_SLEEP_EN

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

SLEEP_EN For sleep mode issue

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

UARTn+004Ch Virtual FIFO enable register — UARTn_VFIFO_EN

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

VFIFO_EN Virtual FIFO mechanism enable signal.

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

UARTn+0050h Rx Trigger Address — UARTn_RXTRI_AD

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

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

UARTn+0054h Fractional Divider LSB Address — UARTn_FRACDIV_L

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

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

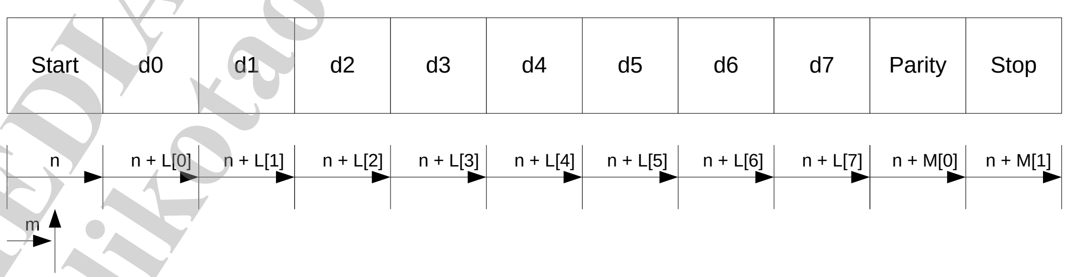
UARTn+0058h Fractional Divider MSB Address — UARTn_FRACDIV_M

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

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

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

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



UARTn+005Ch FIFO Control Register — UARTn_FCR_RD

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

Type									RO			RO

Read out UARTn_FCR register.

UARTn+0060h TX Active Enable Address — UARTn_TX_ACTIVE_EN

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

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

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

4.14 PCM Controller

4.14.1 Features

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

4.14.2 Block Diagram

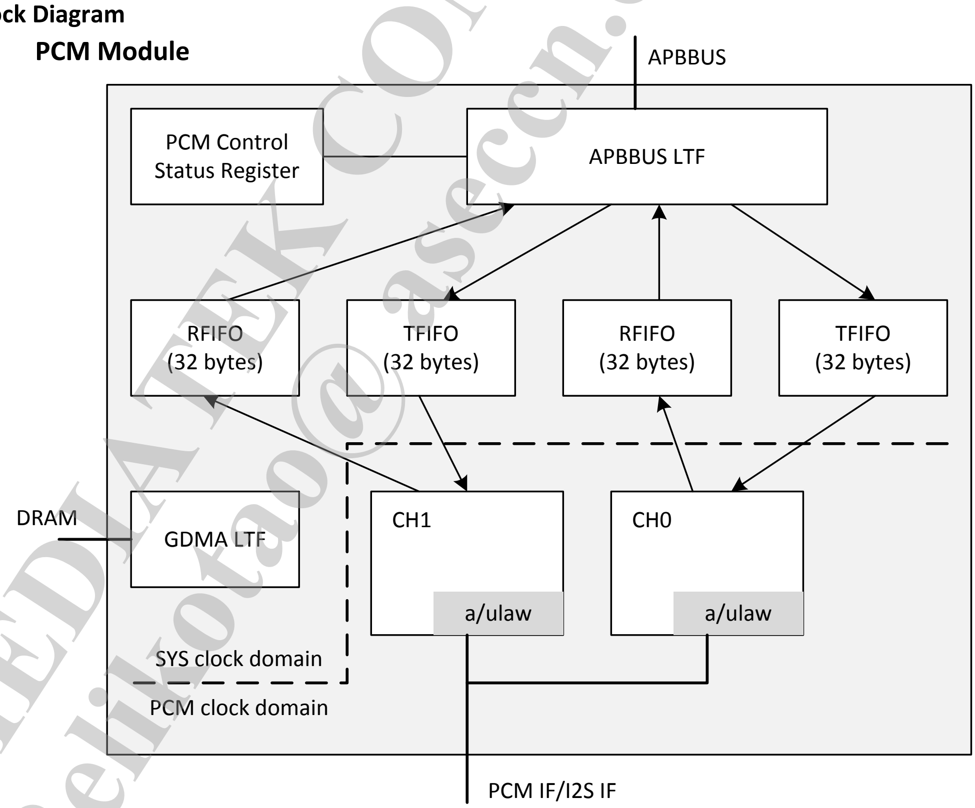


Figure 4-8 PCM Controller Block Diagram

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

The interrupt sources include:

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

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

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

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

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

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

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

4.14.3 List of Registers

4.14.4 PCM Configuration

PCM Initialization Flow

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

PCM Configuration Examples

Below are some examples of PCM configuration.

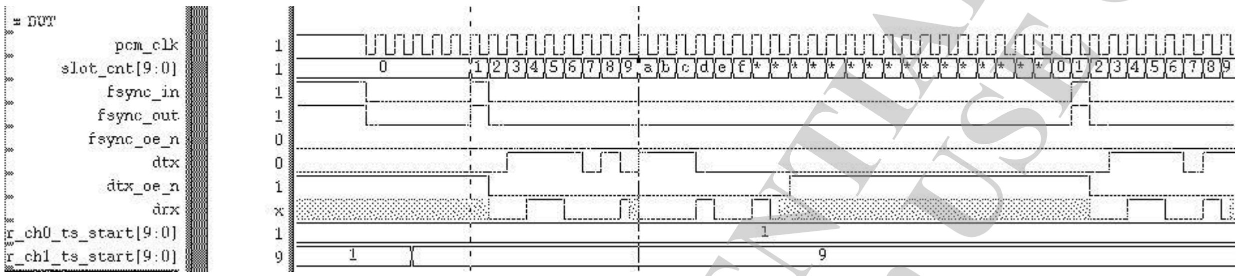
Case 1:

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

CH0_CFG Register: TS_START=1

CH1_CFG Register: TS_START=9

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



Case 2:

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

CH0_CFG Register: TS_START=1

CH1_CFG Register: TS_START=17

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

Case 3:

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

CH0_CFG Register: TS_START=1 (disable)

CH1_CFG Register: TS_START=0x1A

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

4.14.5 Register

PCM Changes LOG

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

Module name: PCM Base address: (+10002000h)

Address	Name	Width	Register Function
10002000	GLB_CFG	32	Global Config
10002004	PCM_CFG	32	PCM configuration
10002008	INT_STATUS	32	Interrupt status
1000200C	INT_EN	32	Interrupt enable
10002010	CHA0_FF_STATUS	32	Channel A0(represents channel 0) FIFO status
10002014	CHB0_FF_STATUS	32	Channel B0(represents channel 1) FIFO status
10002020	CHA0_CFG	32	Channel A0(represents channel 0) Config
10002024	CHB0_CFG	32	Channel B0(represents channel 1) Config
10002030	FSYNC_CFG	32	FSYNC config
10002034	CHA0_CFG2	32	Channel A0(represents channel 0) Config
10002038	CHB0_CFG2	32	Channel B0(represents channel 1) Config
10002040	IP_INFO	32	IP version info
10002044	RSV_REG16	32	SPARE REG 16 bits
10002050	DIVCOMP_CFG	32	Dividor Compensation part config
10002054	DIVINT_CFG	32	Dividor Integer part config
10002060	DIGDELAY_CFG	32	Digital delay config
10002080	CH0_FIFO	32	Channel 0 FIFO access point
10002084	CH1_FIFO	32	Channel 1 FIFO access point
10002088	CH2_FIFO	32	Channel 2 FIFO access point
1000208C	CH3_FIFO	32	Channel 3 FIFO access point

10002110	CHA1_FF_STATUS	32	**Channel A1(represents channel 3) FIFO status**
10002114	CHB1_FF_STATUS	32	**Channel B1(represents channel 4) FIFO status**
10002120	CHA1_CFG	32	**Channel A1(represents channel 3) Config**
10002124	CHB1_CFG	32	**Channel B1(represents channel 1) Config**
10002134	CHA1_CFG2	32	**Channel A1(represents channel 3) Config**
10002138	CHB1_CFG2	32	**Channel B1(represents channel 4) Config**

10002000 GLB_CFG Global Config 00440000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PCM_EN	DMA_EN	LBK_EN	EXT_LBK_EN	RSV0					RFF_THRES			RSV1	TFF_THRES		
Type	RW	RW	RW	RW	RO					RW			RO	RW		
Reset	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV2													CH_EN		
Type	RO													RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

Bit(s)	Name	Description
		It should be >2 and <6. When data in FIFO is over the threshold, an interrupt and DMA are triggered. (unit: word)
15:4	RSV2	**Reserved**
3:0	CH_EN	**Channels 3 to 0 Tx and Rx Enable** 0: disable 1: enable

10002004 **PCM_CFG** **PCM configuration** **03000000**

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

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

Bit(s)	Name	Description
		3: 32 slots, PCM clock out/in should be 2.048 MHz. 4: 64 slots, PCM clock out/in should be 4.096 MHz. 5:128 slots, PCM clock out/in should be 8.192 MHz. Other: Reserved. NOTE: When using the external clock, the frequency clock should be equal to PCM_clock out. Otherwise, the PCM_CLKin should be 8.192 MHz. 0: _4_SLOT 1: _8_SLOT 2: _16_SLOT 3: _32_SLOT 4: _64_SLOT 5: _128_SLOT

10002008 **INT_STATUS** **Interrupt status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RSV0[23:8]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RSV0[7:0]**								**CHT_DMA_FAULT**	**CHT_OVRUN**	**CHT_UNRUN**	**CHT_THRES**	**CHR_DMA_FAULT**	**CHR_OVRUN**	**CHR_UNRUN**	**CHR_THRES**
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000200C **INT_EN** **Interrupt enable** **00000000**

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

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

10002010	<u>CHA0_FF_STATUS</u>	Channel A0(represents channel 0) FIFO status	00100008

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

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A0 Tx DMA signal.**

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

10002014 **CHB0_FF_STATUS** **Channel B0(represents channel 1) FIFO status** **00100008**

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

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B0 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel B0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel B0 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel B0 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAULT	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B0 Rx DMA signal.**
18	CHRX_OVRUN	**Rx FIFO Overrun Interrupt,Asserts when the Channel B0 Rx FIFO is overrun.**

Bit(s)	Name	Description
17	CHRX_UNRUN	**Rx FIFO Underrun Interrupt,Asserts when the Channel B0 Rx FIFO is underrun.**
16	CHRX_THRES	**Rx FIFO Below Threshold Interrupt,Asserts when the Channel B0 FIFO is lower than the defined threshold.**
15:8	RSV1	**Reserved**
7:4	CHRFF_AVCNT	**Channel B0 RXFIFO Available Space Count,Counts the available space for reads in channel B0 RXFIFO.(unit: word)**
3:0	CHTFF_EPCNT	**Channel B0 TXFIFO Available Space Count,Counts the available space for writes in channel B0 TXFIFO.(unit: word)**

10002020 CHA0_CFG Channel A0(represents channel 0) Config 00000001

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

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

10002024 CHB0_CFG Channel B0(represents channel 1) Config 00000001

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

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

10002030 FSYNC_CFG FSYNC config 28000000

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

Bit(s)	Name	Description
31	CFG_FSYNC_EN	**Enables configurable FSYNC.**
30	POS_CAP_DT	**Positive Edge Capture Data, Sets the PCM controller to capture data on the negative or positive edge of the PCM clock. NOTE: This**

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

10002034 **CHA0_CFG2** **Channel A0(represents channel 0) Config** **00000000**

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

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

10002038 **CHB0_CFG2** **Channel B0(represents channel 1) Config** **00000000**

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

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

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

10002040 **IP_INFO** **IP version info** **00000401**

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

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

10002044 **RSV_REG16** **SPARE REG 16 bits** **00000000**

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

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

10002050 **DIVCOMP_CFG** **Dividor Compensation part config** **00000000**

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

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

10002054 **DIVINT_CFG** **Dividor Integer part config** **00000000**

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

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

10002060 **DIGDELAY_CFG** **Digital delay config** **00000002**

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

	D_DIGDLY_EN								EN_DIGDLY_EN							
Type	RW	RO		RW					RW	RO		RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

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

Bit(s)	Name	Description
		1: Enable
6:5	RSV5	**Reserved**
4:0	CHEN_DLYVAL	**Delay Count Value** The delay error = CLK_PERIOD * (SYNC_DELAY + SYNC_DELTA + (DLYCNT_CFG) + 1) For example, DLYCNT_CFG = 4, (SYNC_DELAY is always fixed to 4) Final Delay = CLK_PERIOD * (2 + (-1/0/+1) + (4) + 1) = CLK_PERIOD * (6/7/8)= CLK_PERIOD * (6 to 8) = 25 ns to 33.3 ns NOTE: Period is 1/240 MHz = 4.1667 ns in MT7620.

10002080 **CH0_FIFO** **Channel 0 FIFO access point** **00000000**

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

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

10002084 **CH1_FIFO** **Channel 1 FIFO access point** **00000000**

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

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

10002088 **CH2_FIFO** **Channel 2 FIFO access point** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH2_FIFO[31:16]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CH2_FIFO[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000208C **CH3_FIFO** **Channel 3 FIFO access point** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**CH3_FIFO[31:16]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CH3_FIFO[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

10002110 **CHA1_FF_STATUS** **Channel A1(represents channel 3) FIFO status** **00100008**

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

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A1 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel A0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel A1 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel A0 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAUL	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a**

Bit(s)	Name	Description
	T	Channel A1 Rx DMA signal.
18	CHRX_OVRUN	Rx FIFO Overrun Interrupt,Asserts when the Channel A1 Rx FIFO is overrun.
17	CHRX_UNRUN	Rx FIFO Underrun Interrupt,Asserts when the Channel A1 Rx FIFO is underrun.
16	CHRX_THRES	Rx FIFO Below Threshold Interrupt,Asserts when the Channel A1 FIFO is lower than the defined threshold.
15:8	RSV1	Reserved
7:4	CHRFF_AVCNT	Channel A1 RXFIFO Available Space Count,Counts the available space for reads in channel A1 RXFIFO.(unit: word)
3:0	CHTFF_EPCNT	Channel A1 TXFIFO Available Space Count,Counts the available space for writes in channel A1 TXFIFO.(unit: word)

10002114	CHB1_FF_STATUS	Channel B1(represents channel 4) FIFO status	00100008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0								CHTX_DMA_FAULT	CHTX_OVRUN	CHTX_UNRUN	CHTX_THRES	CHRX_DMA_FAULT	CHRX_OVRUN	CHRX_UNRUN	CHRX_THRES
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1								CHRFF_AVCNT				CHTFF_EPCNT			
Type	RO								RO				RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

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

Bit(s)	Name	Description
3:0	CHTFF_EPCNT	**Channel B1 TXFIFO Available Space Count,Counts the available space for writes in channel B1 TXFIFO.(unit: word)**

10002120 CHA1_CFG Channel A1(represents channel 3) Config 00000001

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

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

10002124 CHB1_CFG Channel B1(represents channel 1) Config 00000001

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

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

10002134 **CHA1_CFG2** **Channel A1(represents channel 3) Config** **00000000**

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

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

10002138 **CHB1_CFG2** **Channel B1(represents channel 4) Config** **00000000**

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

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

4.15 **Generic DMA Controller**

4.15.1 Features

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

4.15.2 Block Diagram

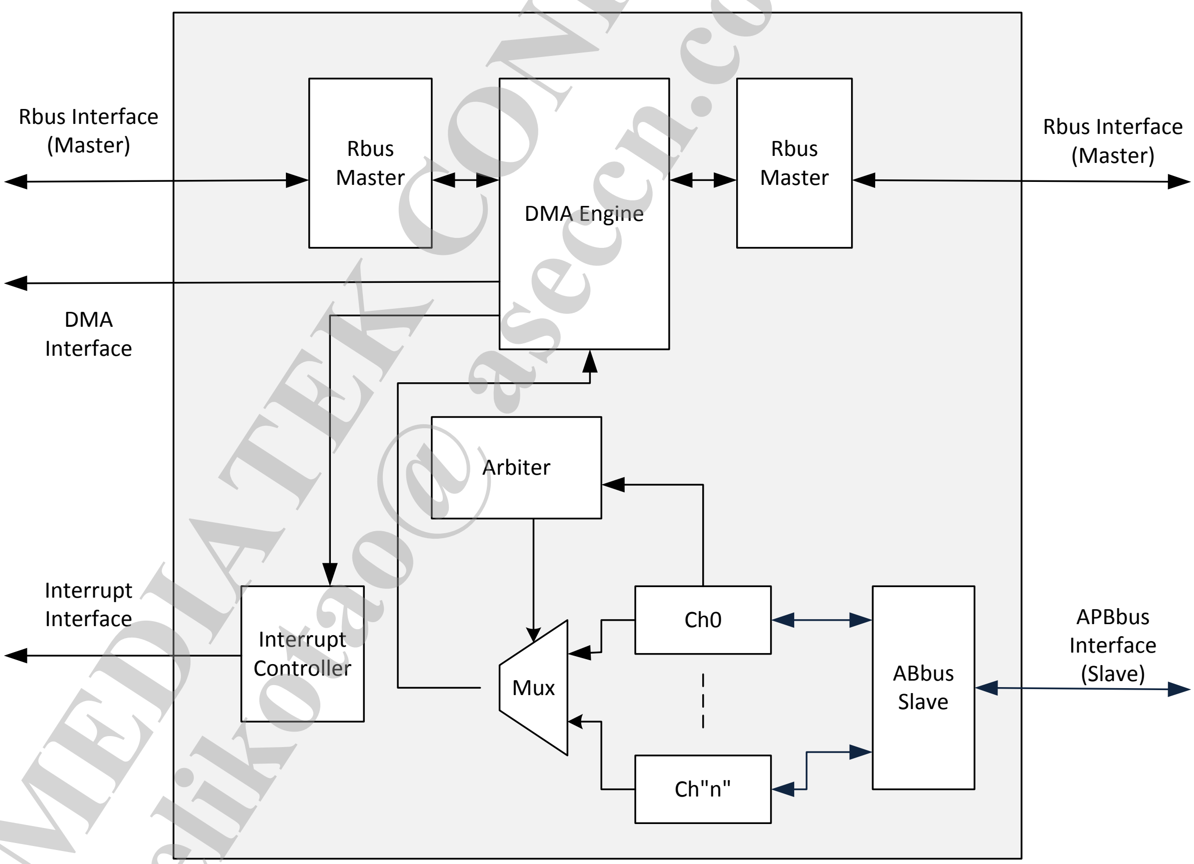


Figure 4-9 Generic DMA Controller Block Diagram

4.15.3 Peripheral Channel Connection

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

4.15.4 Registers

GDMA Changes LOG

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

Module name: GDMA Base address: (+10002800h)

Address	Name	Width	Register Function
10002800	GDMA_SA_0	32	Source Address of GDMA Channel 0
10002804	GDMA_DA_0	32	Destination Address of GDMA Channel 0
10002808	GDMA_CT0_0	32	Control Register 0 of GDMA Channel 0
1000280C	GDMA_CT1_0	32	Control Register 1 of GDMA Channel 0
10002810	GDMA_SA_1	32	Source Address of GDMA Channel 1
10002814	GDMA_DA_1	32	Destination Address of GDMA Channel 1
10002818	GDMA_CT0_1	32	Control Register 0 of GDMA Channel 1
1000281C	GDMA_CT1_1	32	Control Register 1 of GDMA Channel 1
10002820	GDMA_SA_2	32	Source Address of GDMA Channel 2
10002824	GDMA_DA_2	32	Destination Address of GDMA Channel 2
10002828	GDMA_CT0_2	32	Control Register 0 of GDMA Channel 2
1000282C	GDMA_CT1_2	32	Control Register 1 of GDMA Channel 2
10002830	GDMA_SA_3	32	Source Address of GDMA Channel 3
10002834	GDMA_DA_3	32	Destination Address of GDMA Channel 3
10002838	GDMA_CT0_3	32	Control Register 0 of GDMA Channel 3

1000283C	GDMA_CT1_3	32	Control Register 1 of GDMA Channel 3
10002840	GDMA_SA_4	32	Source Address of GDMA Channel 4
10002844	GDMA_DA_4	32	Destination Address of GDMA Channel 4
10002848	GDMA_CT0_4	32	Control Register 0 of GDMA Channel 4
1000284C	GDMA_CT1_4	32	Control Register 1 of GDMA Channel 4
10002850	GDMA_SA_5	32	Source Address of GDMA Channel 5
10002854	GDMA_DA_5	32	Destination Address of GDMA Channel 5
10002858	GDMA_CT0_5	32	Control Register 0 of GDMA Channel 5
1000285C	GDMA_CT1_5	32	Control Register 1 of GDMA Channel 5
10002860	GDMA_SA_6	32	Source Address of GDMA Channel 6
10002864	GDMA_DA_6	32	Destination Address of GDMA Channel 6
10002868	GDMA_CT0_6	32	Control Register 0 of GDMA Channel 6
1000286C	GDMA_CT1_6	32	Control Register 1 of GDMA Channel 6
10002870	GDMA_SA_7	32	Source Address of GDMA Channel 7
10002874	GDMA_DA_7	32	Destination Address of GDMA Channel 7
10002878	GDMA_CT0_7	32	Control Register 0 of GDMA Channel 7
1000287C	GDMA_CT1_7	32	Control Register 1 of GDMA Channel 7
10002880	GDMA_SA_8	32	Source Address of GDMA Channel 8
10002884	GDMA_DA_8	32	Destination Address of GDMA Channel 8
10002888	GDMA_CT0_8	32	Control Register 0 of GDMA Channel 8
1000288C	GDMA_CT1_8	32	Control Register 1 of GDMA Channel 8
10002890	GDMA_SA_9	32	Source Address of GDMA Channel 9
10002894	GDMA_DA_9	32	Destination Address of GDMA Channel 9
10002898	GDMA_CT0_9	32	Control Register 0 of GDMA Channel 9
1000289C	GDMA_CT1_9	32	Control Register 1 of GDMA Channel 9
100028A0	GDMA_SA_10	32	Source Address of GDMA Channel 10
100028A4	GDMA_DA_10	32	Destination Address of GDMA Channel 10
100028A8	GDMA_CT0_10	32	Control Register 0 of GDMA Channel 10
100028AC	GDMA_CT1_10	32	Control Register 1 of GDMA Channel 10
100028B0	GDMA_SA_11	32	Source Address of GDMA Channel 11
100028B4	GDMA_DA_11	32	Destination Address of GDMA Channel 11
100028B8	GDMA_CT0_11	32	Control Register 0 of GDMA Channel 11
100028BC	GDMA_CT1_11	32	Control Register 1 of GDMA Channel 11
100028C0	GDMA_SA_12	32	Source Address of GDMA Channel 12
100028C4	GDMA_DA_12	32	Destination Address of GDMA Channel 12
100028C8	GDMA_CT0_12	32	Control Register 0 of GDMA Channel 12
100028CC	GDMA_CT1_12	32	Control Register 1 of GDMA Channel 12
100028D0	GDMA_SA_13	32	Source Address of GDMA Channel 13
100028D4	GDMA_DA_13	32	Destination Address of GDMA Channel 13
100028D8	GDMA_CT0_13	32	Control Register 0 of GDMA Channel 13
100028DC	GDMA_CT1_13	32	Control Register 1 of GDMA Channel 13
100028E0	GDMA_SA_14	32	Source Address of GDMA Channel 14
100028E4	GDMA_DA_14	32	Destination Address of GDMA Channel 14
100028E8	GDMA_CT0_14	32	Control Register 0 of GDMA Channel 14
100028EC	GDMA_CT1_14	32	Control Register 1 of GDMA Channel 14

100028F0	GDMA_SA_15	32	**Source Address of GDMA Channel 15**
100028F4	GDMA_DA_15	32	**Destination Address of GDMA Channel 15**
100028F8	GDMA_CT0_15	32	**Control Register 0 of GDMA Channel 15**
100028FC	GDMA_CT1_15	32	**Control Register 1 of GDMA Channel 15**
10002A00	GDMA_UNMASK_INTSTS	32	**Unmask Fail Interrupt Status**
10002A04	GDMA_DONE_INTSTS	32	**Segment Done Interrupt Status**
10002A20	GDMA_GCT	32	**Global Control**
10002A30	GDMA_PERI_ADDR_START_0	32	**Peripheral Region 0 Starting Address**
10002A34	GDMA_PERI_ADDR_END_0	32	**Peripheral Region 0 End Address**
10002A38	GDMA_PERI_ADDR_START_1	32	**Peripheral Region 1 Starting Address**
10002A3C	GDMA_PERI_ADDR_END_1	32	**Peripheral Region 1 End Address**
10002A40	GDMA_PERI_ADDR_START_2	32	**Peripheral Region 2 Starting Address**
10002A44	GDMA_PERI_ADDR_END_2	32	**Peripheral Region 2 End Address**
10002A48	GDMA_PERI_ADDR_START_3	32	**Peripheral Region 3 Starting Address**
10002A4C	GDMA_PERI_ADDR_END_3	32	**Peripheral Region 3 End Address**

10002800 GDMA_SA_0 Source Address of GDMA Channel 0 00000000

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

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

10002804 GDMA_DA_0 Destination Address of GDMA Channel 0 00000000

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

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

10002808 **GDMA_CT0_0** **Control Register 0 of GDMA Channel 0** **00000000**

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

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

Bit(s)	Name	Description
		1: Software mode

1000280C **GDMA_CT1_0** **Control Register 1 of GDMA Channel 0** **00000000**

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

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

Bit(s)	Name	Description
		0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002810 GDMA_SA_1 Source Address of GDMA Channel 1 00000000

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

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

10002814 GDMA_DA_1 Destination Address of GDMA Channel 1 00000000

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

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

10002818 GDMA_CT0_1 Control Register 0 of GDMA Channel 1 00000000

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

Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000281C **GDMA_CT1_1** **Control Register 1 of GDMA Channel 1** **00000000**

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

	ED	DE_EN												NT_INT_EN	ASK_FAIL_INT_EN	K
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

10002820 **GDMA_SA_2** **Source Address of GDMA Channel 2** **0000000**

0

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

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

10002824 GDMA_DA_2 Destination Address of GDMA Channel 2 00000000

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

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

10002828 GDMA_CT0_2 Control Register 0 of GDMA Channel 2 00000000

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

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode

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

1000282C **GDMA_CT1_2** **Control Register 1 of GDMA Channel 2** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ						NEXT_CH2UNMASK					COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

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

10002830 **GDMA_SA_3** **Source Address of GDMA Channel 3** **00000000**

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

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

10002834 **GDMA_DA_3** **Destination Address of GDMA Channel 3** **00000000**

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

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

10002838 **GDMA_CT0_3** **Control Register 0 of GDMA Channel 3** **00000000**

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

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined

Bit(s)	Name	Description
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1000283C **GDMA_CT1_3** **Control Register 1 of GDMA Channel 3** **00000000**

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

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)

Bit(s)	Name	Description
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002840 GDMA_SA_4 Source Address of GDMA Channel 4 00000000

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

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

10002844 GDMA_DA_4 Destination Address of GDMA Channel 4 00000000

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

Bit(s)	Name	Description

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

10002848 **GDMA_CT0_4** **Control Register 0 of GDMA Channel 4** **00000000**

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

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

1000284C **GDMA_CT1_4** **Control Register 1 of GDMA Channel 4** **00000000**

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

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

Bit(s)	Name	Description
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002850 GDMA_SA_5 Source Address of GDMA Channel 5 00000000

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

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

10002854 GDMA_DA_5 Destination Address of GDMA Channel 5 00000000

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

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

10002858 GDMA_CT0_5 Control Register 0 of GDMA Channel 5 00000000

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

									ADDR_MODE	DR_MODE				T_DONE_INT_EN		E_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000285C __GDMA_CT1_5__ **Control Register 1 of GDMA Channel 5** **00000000**

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

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

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

10002860 **GDMA_SA_6** **Source Address of GDMA Channel 6** **00000000**

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

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

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

10002864 GDMA_DA_6 Destination Address of GDMA Channel 6 00000000

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

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

10002868 GDMA_CT0_6 Control Register 0 of GDMA Channel 6 00000000

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

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MOD	**Sets the destination address mode**

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

1000286C **GDMA_CT1_6** **Control Register 1 of GDMA Channel 6** **00000000**

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

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

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

10002870 **GDMA_SA_7** **Source Address of GDMA Channel 7** **00000000**

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

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

10002874 **GDMA_DA_7** **Destination Address of GDMA Channel 7** **00000000**

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

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

10002878 **GDMA_CT0_7** **Control Register 0 of GDMA Channel 7** **00000000**

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

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

Bit(s)	Name	Description
		0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1000287C **GDMA_CT1_7** **Control Register 1 of GDMA Channel 7** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ						NEXT_CH2UNMASK					COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will**

Bit(s)	Name	Description
		clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself. 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002880 GDMA_SA_8 Source Address of GDMA Channel 8 00000000

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

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

10002884 GDMA_DA_8 Destination Address of GDMA Channel 8 00000000

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

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

10002888 **GDMA_CT0_8** **Control Register 0 of GDMA Channel 8** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000288C **GDMA_CT1_8** **Control Register 1 of GDMA Channel 8** **00000000**

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

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

Bit(s)	Name	Description
		0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002890 GDMA_SA_9 Source Address of GDMA Channel 9 00000000

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

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

10002894 GDMA_DA_9 Destination Address of GDMA Channel 9 00000000

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

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

10002898 GDMA_CT0_9 Control Register 0 of GDMA Channel 9 00000000

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

									ODE	E				NT_EN		
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

1000289C **GDMA_CT1_9** **Control Register 1 of GDMA Channel 9** **00000000**

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

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

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

100028A0 **GDMA_SA_10** **Source Address of GDMA Channel 10** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

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

100028A4 GDMA_DA_10 Destination Address of GDMA Channel 10 00000000

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

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

100028A8 GDMA_CT0_10 Control Register 0 of GDMA Channel 10 00000000

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

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction**

Bit(s)	Name	Description
		0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

100028AC **GDMA_CT1_10** **Control Register 1 of GDMA Channel 10** **00000000**

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

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear**

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

100028B0 **GDMA_SA_11** **Source Address of GDMA Channel 11** **00000000**

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

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

100028B4 **GDMA_DA_11** **Destination Address of GDMA Channel 11** **00000000**

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

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

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

100028B8	GDMA_CT0_11	Control Register 0 of GDMA Channel 11	00000000

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

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

Bit(s)	Name	Description
		number of bytes transfferred reaches the TARGET_BYTE_CNT 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

100028BC **GDMA_CT1_11** **Control Register 1 of GDMA Channel 11** **00000000**

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

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.**

Bit(s)	Name	Description
		0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

100028C0 GDMA_SA_12 Source Address of GDMA Channel 12 00000000

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

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

100028C4 GDMA_DA_12 Destination Address of GDMA Channel 12 00000000

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

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

100028C8	GDMA_CT0_12	Control Register 0 of GDMA Channel 12	00000000

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

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

100028CC	GDMA_CT1_12	Control Register 1 of GDMA Channel 12	0000000

2 0

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

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

Bit(s)	Name	Description
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

100028D0 **GDMA_SA_13** **Source Address of GDMA Channel 13** **00000000**

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

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

100028D4 **GDMA_DA_13** **Destination Address of GDMA Channel 13** **00000000**

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

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

100028D8 **GDMA_CT0_13** **Control Register 0 of GDMA Channel 13** **00000000**

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

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

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

100028DC **GDMA_CT1_13** **Control Register 1 of GDMA Channel 13** **00000000**

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

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

100028E0 **GDMA_SA_14** **Source Address of GDMA Channel 14** **00000000**

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

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

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

100028E4 GDMA_DA_14 Destination Address of GDMA Channel 14 00000000

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

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

100028E8 GDMA_CT0_14 Control Register 0 of GDMA Channel 14 00000000

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

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs

Bit(s)	Name	Description
		3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

100028EC **GDMA_CT1_14** **Control Register 1 of GDMA Channel 14** **00000000**

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

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled

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

100028F0 GDMA_SA_15 Source Address of GDMA Channel 15 00000000

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

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

100028F4 GDMA_DA_15 Destination Address of GDMA Channel 15 00000000

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

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

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

100028F8 **GDMA_CT0_15** **Control Register 0 of GDMA Channel 15** **00000000**

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

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

Bit(s)	Name	Description
		starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted. 0: Hardware mode 1: Software mode

100028FC	GDMA_CT1_15	Control Register 1 of GDMA Channel 15	00000000

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

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

Bit(s)	Name	Description
	N	**destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

10002A00 **GDMA_UNMASK_INTSTS** **Unmask Fail Interrupt Status** **00000000**

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

Bit(s)	Name	Description
31:0	UNMASK_FAIL_INTSTS	**This field is the bit-map of unmask fail interrupt status of each channel. The unmask fail interrupt will assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.**

10002A04 **GDMA_DONE_INTSTS** **Segment Done Interrupt Status** **00000000**

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

Bit(s)	Name	Description
31:0	SEGMENT_DONE_INTSTS	**This field is the bit-map of segment done interrupt status of each channel. The segment done interrupt will assert when each segment is transferred completely.**

10002A20 **GDMA_GCT** **Global Control** **0000000E**

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

Bit(s)	Name	Description
4:3	TOTAL_CH_NUM	**Total channel number supported** 0: 8 channels 1: 16 channels 2: 32 channels 3: Undefined
2:1	IP_VER	**GDMA core version**
0	ARB_MODE	**Arbitration mode selection** 0: channel 0 has highest priority and others are round-robin 1: All channel are round-robin

10002A30 **GDMA_PERI_ADDR_START_0** **Peripheral Region 0 Starting Address** **10000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_0[31:16]															
Type	RW															
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_0	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A34 **GDMA_PERI_ADDR_END_0** **Peripheral Region 0 End Address** **20000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_0[31:16]															
Type	RW															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_0	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A38 **GDMA_PERI_ADDR_START_1** **Peripheral Region 1 Starting Address** **20000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_1[31:16]															
Type	RW															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_1	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A3C **GDMA_PERI_ADDR_END_1** **Peripheral Region 1 End Address** **30000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_1[31:16]															
Type	RW															
Reset	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_1	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A40 **GDMA_PERI_ADDR_START_2** **Peripheral Region 2 Starting Address** **10000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_2[31:16]															
Type	RW															
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_2	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A44 **GDMA_PERI_ADDR_END_2** **Peripheral Region 2 End Address** **20000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_2[31:16]															
Type	RW															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_2	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A48 **GDMA_PERI_ADDR_START_3** **Peripheral Region 3 Starting Address** **60000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_3[31:16]															
Type	RW															
Reset	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_3	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

10002A4C **GDMA_PERI_ADDR_END_3** **Peripheral Region 3 End Address** **70000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_3[31:16]															
Type	RW															
Reset	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_	**GDMA request will direct to peripheral bus if the request address >=**

Bit(s)	Name	Description
	3	**PERI_ADDR_START_x & < PERI_ADDR_END_x**

4.16 **AES Controller**

4.16.1 Registers

AES Changes LOG

Revision	Date	Author	Change Log
0.1	2013/4/30	Morrie Lin	Initialization
0.2	2013/6/5	Morrie Lin	Add desc_5dw_info_en register
0.3	2013/6/7	Morrie Lin	Update AES base address

Module name: AES Base address: (+10004000h)

Address	Name	Width	Register Function
10004000	TX_BASE_PTR0	32	**TX_BASE_PTR0** Used for DMA base address of TX ring0
10004004	TX_MAX_CNT0	32	**TX_MAX_CNT0** Used for DMA max number of TX ring0
10004008	TX_CTX_IDX0	32	**TX_CTX_IDX0** Used for CPU pointer of TX ring0
1000400C	TX_DTX_IDX0	32	**TX_DTX_IDX0** Used for DMA pointer of TX ring0
10004100	RX_BASE_PTR0	32	**RX_BASE_PTR0** Used for DMA base address of RX ring0
10004104	RX_MAX_CNT0	32	**RX_MAX_CNT0** Used for DMA max number of RX ring0
10004108	RX_CALC_IDX0	32	**RX_CALC_IDX0** Used for CPU pointer of RX ring0
1000410C	FS_DRX_IDX0	32	**FS_DRX_IDX0** Used for DMA pointer of RX ring0
10004200	PDMA_INFO	32	**PDMA_INFO** used for PDMA information
10004204	PDMA_GLO_CFG	32	**PDMA_GLO_CFG** used for PDMA setting
10004208	PDMA_RST_IDX	32	**PDMA_RST_IDX** used for PDMA setting
1000420C	DELAY_INT_CFG	32	**DELAY_INT_CFG** used for PDMA setting
10004210	PDMA_Q_CFG	32	**PDMA_Q_CFG** used for PDMA setting
10004220	PDMA_INT_STA	32	**PDMA_INT_STA** used for PDMA setting
10004228	PDMA_INT_MSK	32	**PDMA_INT_MSK** used for PDMA setting

10004000	**TX_BASE_PTR0**	**TX_BASE_PTR0**	**00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_BASE_PTR0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	TX_BASE_PTR0	**Tx Base Pointer 0** Points to the base address of TX_Ring 0 (If enable desc_5dw_info_en 8-DWORD aligned address, else 4-DWORD aligned address).

10004004 **TX_MAX_CNT0** **TX_MAX_CNT0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_MAX_CNT0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	TX_MAX_CNT0	**Tx Maximum TXD Count 0** The maximum TXD count in TXD_Ring 0.

10004008 **TX_CTX_IDX0** **TX_CTX_IDX0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_MAX_CNT0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	TX_MAX_CNT0	**Tx CPU TXD Index n** Points to the next TXD to be used by the CPU. (If enable desc_5dw_info_en, 8-DWORD aligned address, else 4-DWORD aligned address).

1000400C **TX_DTX_IDX0** **TX_DTX_IDX0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV[7:0]								TX_DTX_IDX0							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RESV	**Reserved**
7:0	TX_DTX_IDX0	**Tx DMA TXD Index n** Points to the next TXD to be used by the DMA. (If enable desc_5dw_info_en, 8-DWORD aligned address, else 4-DWORD aligned address).

10004100 **RX_BASE_PTR0** **RX_BASE_PTR0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	RX_BASE_PTR0	**Rx Base Pointer 0** Points to the base address of RXD Ring 0 (If enable desc_5dw_info_en, 8-DWORD aligned address, else 4-DWORD aligned address).

10004104 **RX_MAX_CNT0** **RX_MAX_CNT0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_MAX_CNT0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	RX_MAX_CNT0	**Rx Maximum Count 0** The maximum RXD count in RXD Ring 0.

10004108 **RX_CALC_IDX0** **RX_CALC_IDX0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RX_CALI_IDX0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	**Reserved**
15:0	RX_CALI_IDX0	**Rx CPU RXD Index 0** Points to the next RXD the CPU will allocate to RXD Ring 0. (If enable desc_5dw_info_en, 8-DWORD aligned address, else 4-DWORD aligned address).

1000410C **FS_DRX_IDX0** **FS_DRX_IDX0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	RESV[7:0]								RX_DRX_IDX0							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RESV	**Reserved**
7:0	RX_DRX_IDX0	**Rx DMA RXD Index n** Points to the next RXD that the DMA will use in FDS Ring 0. (If enable desc_5dw_info_en, 8-DWORD aligned address, else 4-DWORD aligned address).

10004200 **PDMA_INFO** **PDMA_INFO** **4C000101**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VERSION				INDEX_WIDTH				BASE_PTR_WIDTH							
Type	RO				RO				RO							
Reset	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0

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

Bit(s)	Name	Description
31:28	VERSION	**PDMA controller version.**
27:24	INDEX_WIDTH	**RX Ring index width**
23:16	BASE_PTR_WIDTH	**Base Pointer Width**
15:8	RX_RING_NUM	**Rx ring number**
7:0	TX_RING_NUM	**Tx ring number**

10004204 **PDMA_GLO_CFG** **PDMA_GLO_CFG** **00000450**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_2B_OFFSET	CLKGATE_BYP	BYTE_SWAP	RESV[16:4]												
Type	RW	RO	RO	RO												
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV[3:0]				desc_5dw_info_en	multi_dma_en	share_fifo_en	desc_32b_en	BIG_ENDIAN	TX_WB_DDONE	WPDMA_BT_SIZE		RX_DMA_BUSY	RX_DMA_EN	TX_DMA_BUSY	TX_DMA_EN
Type	RO				RW	RW	RW	RW	RW	RW	RW		RO	RW	RO	RW
Reset	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0

Bit(s)	Name	Description
31	RX_2B_OFFSET	**Rx 2 Byte Offset** Sets the byte size of the Rx buffer offset. 0: 4 bytes 1: 2 bytes. 0
30	CLKGATE_BYP	**Clock Gating Control Status Register** Controls gating of the PDMA clock. 0: PDMA clock operates in freerun mode. 1: PDMA clock is gated when idle.
29	BYTE_SWAP	**Byte Swap** The DMA applies the endian rule to convert the descriptor. 0: Byte swap not applied. 1: Apply byte swap.
28:12	RESV	**Reserved**
11	desc_5dw_info_en	**Support extension tx_info/rx_info to to 20 byte and the total length of descriptor is 32 byte.** 0: Disable 1: Enable
10	multi_dma_en	
9	share_fifo_en	

Bit(s)	Name	Description
8	desc_32b_en	**Support 32 Byte alignment descriptor** Enables support for 32 Byte alignment PDMA descriptors. 0: Disable 1: Enable
7	BIG_ENDIAN	**Selects the Endian mode for the SoC platform section.** DMA applies the endian rule to convert payload and Tx/Rx information. DMA does not apply the endian rule to registers or descriptors. 0: Little endian 1: Big endian
6	TX_WB_DDONE	**Tx Write Back DDONE** Enables TX_DMA writing back DDONE into TXD. 0: Disable 1: Enable
5:4	WPDMA_BT_SIZE	**PDMA Burst Size** Defines the burst size of PDMA. 0 : 4 DWORD (16bytes). 1 : 8 DWORD (32 bytes). 2 : 16 DWORD (64 bytes). 3 : 32 DWORD (128 bytes)
3	RX_DMA_BUSY	**1 : RX_DMA is busy. 0 : RX_DMA is not busy**
2	RX_DMA_EN	**Rx DMA Enable** Enables Rx DMA. When disabled, Rx DMA finishes the current receiving packet, and then stops. 0: Disable 1: Enable
1	TX_DMA_BUSY	**Indicates whether Tx DMA is busy.** 0: Not busy 1: Busy
0	TX_DMA_EN	**Tx DMA Enable** Enables Tx DMA. When disabled, Tx DMA finishes the current sending packet, and then stops. 0: Disable 1: Enable

10004208 **PDMA_RST_IDX** **PDMA_RST_IDX** **00000000**

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

Bit(s)	Name	Description
31:0	RESV	**Reserved**

1000420C **DELAY_INT_CFG** **DELAY_INT_CFG** **00000000**

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

Bit(s)	Name	Description
31	TXDLY_INT_EN	**Tx Delay Interrupt Enable** Enables the Tx delayed interrupt mechanism. 0: Disable 1: Enable
30:24	TXMAX_PINT	**Tx Maximum Pending Interrupts** Specifies the maximum number of pending interrupts. When the number of pending interrupts is equal to or greater than the value specified here or the interrupt pending time has reached the limit (see below), a final TX_DLY_INT is generated. 0: Disable this feature.
23:16	TXMAX_PTIME	**Tx Maximum Pending Time** Specifies the maximum pending time for the internal TX_DONE_INT0 and TX_DONE_INT1. When the pending time is equal to or greater than TXMAX_PTIME x 20us or the number of pended TX_DONE_INT0 and TX_DONE_INT1 is equal to or greater than TXMAX_PINT (see above), a final TX_DLY_INT is generated 0: Disable this feature.
15	RXDLY_INT_EN	**Rx Delay Interrupt Enable** Enables the Rx delayed interrupt mechanism. 0: Disable 1: Enable
14:8	RXMAX_PINT	**Rx Maximum Pending Interrupts** Specifies the maximum number of pending interrupts. When the number of pended interrupts is equal to or greater than the value specified here or the interrupt pending time has reached the limit (see below), a final RX_DLY_INT is generated. 0: Disable this feature.
7:0	RXMAX_PTIME	**Rx Maximum Pending Time** Specifies the maximum pending time for the internal RX_DONE_INT. When the pending time is equal to or greater than RXMAX_PTIME x 20 us, or the number of pended RX_DONE_INT is equal to or greater than RXMAX_PCNT (see above), a final RX_DLY_INT is generated. 0: Disable this feature.

10004210 **PDMA_Q_CFG** **PDMA_Q_CFG** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[27:12]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV[11:0]												RST_DRX_IDX1			
Type	RO												RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RESV	
3:0	RST_DRX_IDX1	**Will stop to block interface as RX-descriptors reach this threshold**

10004220 **PDMA_INT_STA** **PDMA_INT_STA** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	TX_COHERENT	TX_DLY_INT	RESV1											RX_DONE_INT
Type	RW	RW	RW	RW	RO											RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV															TX_DONE_INT
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**Rx Coherent Interrupt** Asserts when the Rx DMA is ready to handle a queue, but cannot access the queue because the driver is not ready.
30	RX_DLY_INT	**Rx Delay Interrupt** Asserts when the number of pended Rx interrupts has reached a specified level, or when the pending time is reached. Configure this interrupt using the DELAY_INT_CFG register.
29	TX_COHERENT	**Tx Coherent Interrupt** Asserts when the Tx DMA is ready to handle a queue, but cannot access the queue because the driver is not ready.
28	TX_DLY_INT	**Tx Delay Interrupt** Asserts when the number of pended Tx interrupts has reached a specified level, or when the pending time is reached. Configure this interrupt using the DELAY_INT_CFG register.
27:17	RESV1	
16	RX_DONE_INT	**Rx Queue 0 Done Interrupt** Asserts when an Rx packet is received on Queue 0.

Bit(s)	Name	Description
15:1	RESV	
0	TX_DONE_INT	**Tx Queue 0 Done Interrupt** Asserts when a Tx Queue 0 packet is transmitted.

10004228 **PDMA_INT_MSK** **PDMA_INT_MSK** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RX_COHERENT_EN**	**RX_DLY_INT_EN**	**TX_COHERENT_INT_EN**	**TX_DLY_INT_EN**	**RESV1**											**RX_DONE_INT_EN**
Type	RW	RW	RW	RW	RO											RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RESV**															**TX_DONE_INT_EN**
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT_EN	**Masks the Rx Coherent interrupt. This interrupt asserts when the Rx DMA is ready to handle a queue, but cannot access the queue because the driver is not ready.**
30	RX_DLY_INT_EN	**Masks the Rx Delay interrupt. This interrupt asserts when the number of pending Rx interrupts has reached a specified level, or when the pending time is reached.**
29	TX_COHERENT_INT_EN	**Masks the Tx Coherent interrupt. This interrupt asserts when the Tx DMA is ready to handle a queue, but cannot access the queue because the driver is not ready.**
28	TX_DLY_INT_EN	**Masks the Tx Delay interrupt. This interrupt asserts when the number of pending Tx interrupts has reached a specified level, or when the pending time is reached.**
27:17	RESV1	
16	RX_DONE_INT_EN	**Masks the Rx Queue 0 Done interrupt. This interrupt asserts when an Rx packet is received on Queue 0.**
15:1	RESV	
0	TX_DONE_INT_EN	**Masks the Tx Queue 0 Done interrupt. This interrupt asserts when a Tx packet is transmitted on Queue 0.**

4.17 PWM (Pulse Width Modulation)

4.17.1 Registers

PWM Changes LOG

Revision	Date	Author	Change Log
1	2013/11/26	Rick Ho	Initial Version

Module name: PWM Base address: (+10005000h)

Address	Name	Width	Register Function
10005000	PWM_ENABLE	32	PWM Enable register
10005010	PWM0_CON	32	PWM0 Control register
10005014	PWM0_HDURATION	32	PWM0 High Duration register
10005018	PWM0_LDURATION	32	PWM0 Low Duration register
1000501C	PWM0_GDURATION	32	PWM0 Guard Duration register
10005030	PWM0_SEND_DATA0	32	PWM0 Send Data0 register
10005034	PWM0_SEND_DATA1	32	PWM0 Send Data1 register
10005038	PWM0_WAVE_NUM	32	PWM0 Wave Number register
1000503C	PWM0_DATA_WIDTH	32	PWM0 Data Width register
10005040	PWM0_THRESH	32	PWM0 Thresh register
10005044	PWM0_SEND_WAVENUM	32	PWM0 Send Wave Number register
10005050	PWM1_CON	32	PWM1 Control register
10005054	PWM1_HDURATION	32	PWM1 High Duration register
10005058	PWM1_LDURATION	32	PWM1 Low Duration register
1000505C	PWM1_GDURATION	32	PWM1 Guard Duration register
10005070	PWM1_SEND_DATA0	32	PWM1 Send Data0 register
10005074	PWM1_SEND_DATA1	32	PWM1 Send Data1 register
10005078	PWM1_WAVE_NUM	32	PWM1 Wave Number register
1000507C	PWM1_DATA_WIDTH	32	PWM1 Data Width register
10005080	PWM1_THRESH	32	PWM1 Thresh register
10005084	PWM1_SEND_WAVENUM	32	PWM1 Send Wave Number register

10005090	PWM2_CON	32	PWM2 Control register
10005094	PWM2_HDURATION	32	PWM2 High Duration register
10005098	PWM2_LDURATION	32	PWM2 Low Duration register
1000509C	PWM2_GDURATION	32	PWM2 Guard Duration register
100050B0	PWM2_SEND_DATA0	32	PWM2 Send Data0 register
100050B4	PWM2_SEND_DATA1	32	PWM2 Send Data1 register
100050B8	PWM2_WAVE_NUM	32	PWM2 Wave Number register
100050BC	PWM2_DATA_WIDTH	32	PWM2 Data Width register
100050C0	PWM2_THRESH	32	PWM2 Thresh register
100050C4	PWM2_SEND_WAVENUM	32	PWM2 Send Wave Number register
100050D0	PWM3_CON	32	PWM3 Control register
100050D4	PWM3_HDURATION	32	PWM3 High Duration register
100050D8	PWM3_LDURATION	32	PWM3 Low Duration register
100050DC	PWM3_GDURATION	32	PWM3 Guard Duration register
100050F0	PWM3_SEND_DATA0	32	PWM3 Send Data0 register
100050F4	PWM3_SEND_DATA1	32	PWM3 Send Data1 register
100050F8	PWM3_WAVE_NUM	32	PWM3 Wave Number register
100050FC	PWM3_DATA_WIDTH	32	PWM3 Data Width register
10005100	PWM3_THRESH	32	PWM3 Thresh register
10005104	PWM3_SEND_WAVENUM	32	PWM3 Send Wave Number register
1000520C	PWM_EN_STATUS	32	PWM Enable Status register

10005000 **PWM_ENABLE** **PWM Enable register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[27:12]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV[11:0]												PWM3_EN	PWM2_EN	PWM1_EN	PWM0_EN
Type	RO												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RESV	**RESERVED**
3	PWM3_EN	0: disable PWM3 1: enable PWM3
2	PWM2_EN	0: disable PWM2 1: enable PWM2
1	PWM1_EN	0: disable PWM1 1: enable PWM1
0	PWM0_EN	0: disable PWM0 1: enable PWM0

10005010 **PWM0_CON** **PWM0 Control register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	RESV1			CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RO			RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**RESERVED**
15	OLD_PWM_MODE	0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**The stop bit position for source data in periodical mode. In FIFO mode, it's used to indicate the stop bit position in total 64 bits. In Memory mode, it's for the stop bit position in the last 32 bits.**
8	GUARD_VALUE	**PWM0 output value when guard time.**
7	IDLE_VALUE	**PWM0 output value when idle state.**
6:4	RESV1	**RESERVED**
3	CLKSEL	**Select PWM0 clock** 0: CLK= 100KHz 1: CLK= 40MHz
2:0	CLKDIV	**Select PWM0 clock scale.** 000: CLK Hz 001: CLK/2 Hz 010: CLK/4 Hz 011: CLK/8 Hz 100: CLK/16 Hz 101: CLK/32 Hz 110: CLK/64 Hz 111: CLK/128 Hz

10005014 **PWM0_HDURATION** **PWM0 High Duration register** **00000001**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	HDURATION	**PWM0 pulse duration based on the current clock when PWM output is high. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

10005018 **PWM0_LDURATION** **PWM0 Low Duration register** **00000001**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	LDURATION	**PWM0 pulse duration based on the current clock when PWM output is low. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

1000501C **PWM0_GDURATION** **PWM0 Guard Duration register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	GUARD_DURATIO	

Bit(s)	Name	Description
	N	

10005030 **PWM0_SEND_DATA0** **PWM0 Send Data0 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM0 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

10005034 **PWM0_SEND_DATA1** **PWM0 Send Data1 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM0 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

10005038 **PWM0_WAVE_NUM** **PWM0 Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM0 will generate from the pulse data repeatedly.** Note: If WAVE_NUM=0, the waveform generation will not stop until it is disabled.

1000503C	PWM0_DATA_WIDTH	PWM0 Data Width register	00000000

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	DATA_WIDTH	**The PWM0 pulse data width in the old PWM mode.**

10005040	PWM0_THRESH	PWM0 Thresh register	00000000

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	THRESH	**The PWM0 pulse data high/low switching threshold in the old PWM mode.**

10005044	PWM0_SEND_WAVENUM	PWM0 Send Wave Number register	00000000

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	SEND_WAVENUM	**The number by which PWM0 has already generated from the specified data source in the periodical mode.**

10005050 **PWM1_CON** **PWM1 Control register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	RESV1			CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RO			RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**RESERVED**
15	OLD_PWM_MODE	**Use old PWM mode** Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode). 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode).**
8	GUARD_VALUE	**PWM1 output value when guard time.**
7	IDLE_VALUE	**PWM1 output value when idle state.**
6:4	RESV1	**Select Random Generator mode**
3	CLKSEL	**Select PWM1 clock** 0: CLK= 100KHz 1: CLK= 40MHz
2:0	CLKDIV	**Select PWM1 clock scale.** 000: CLK Hz 001: CLK/2 Hz 010: CLK/4 Hz 011: CLK/8 Hz 100: CLK/16 Hz 101: CLK/32 Hz 110: CLK/64 Hz 111: CLK/128 Hz

10005054 **PWM1_HDURATION** **PWM1 High Duration register** **00000001**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	HDURATION	**PWM1 pulse duration based on the current clock when PWM output is high. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

10005058 PWM1_LDURATION PWM1 Low Duration register 00000001

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	LDURATION	**PWM1 pulse duration based on the current clock when PWM output is low. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

1000505C PWM1_GDURATION PWM1 Guard Duration register 00000000

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	GUARD_DURATION	

10005070 **PWM1_SEND_DATA0** **PWM1 Send Data0 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM1 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

10005074 **PWM1_SEND_DATA1** **PWM1 Send Data1 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM1 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

10005078 **PWM1_WAVE_NUM** **PWM1 Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	WAVE_NUM	**The number by which PWM1 will generate from the pulse data repeatedly.** Note: If WAVE_NUM=0, the waveform generation will not stop until it is disabled.

1000507C PWM1_DATA_WIDTH PWM1 Data Width register 00000000

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	DATA_WIDTH	**The PWM1 pulse data width in the old PWM mode.**

10005080 PWM1_THRESH PWM1 Thresh register 00000000

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	THRESH	**The PWM1 pulse data high/low switching threshold in the old PWM mode.**

10005084 PWM1_SEND_WAVENUM PWM1 Send Wave Number register 00000000

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	SEND_WAVENUM	**The number by which PWM1 has already generated from the specified data source in the periodical mode.**

10005090 **PWM2_CON** **PWM2 Control register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RESV0**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**OLD_PWM_MODE**	**STOP_BITPOS**						**GUARD_VALUE**	**IDLE_VALUE**	**RESV1**			**CLKSEL**	**CLKDIV**		
Type	RW	RW						RW	RW	RO			RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**RESERVED**
15	OLD_PWM_MODE	**Use old PWM mode** Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode). 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode).**
8	GUARD_VALUE	**PWM2 output value when guard time.**
7	IDLE_VALUE	**PWM2 output value when idle state.**
6:4	RESV1	**Select Random Generator mode**
3	CLKSEL	**Select PWM2 clock** 0: CLK= 100KHz 1: CLK= 40MHz
2:0	CLKDIV	**Select PWM2 clock scale.** 000: CLK Hz 001: CLK/2 Hz 010: CLK/4 Hz 011: CLK/8 Hz 100: CLK/16 Hz 101: CLK/32 Hz 110: CLK/64 Hz 111: CLK/128 Hz

10005094 **PWM2_HDURATION** **PWM2 High Duration register** **00000001**

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

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	HDURATION	**PWM2 pulse duration based on the current clock when PWM output is high. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

10005098 PWM2_LDURATION PWM2 Low Duration register 00000001

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	LDURATION	**PWM2 pulse duration based on the current clock when PWM output is low. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

1000509C PWM2_GDURATION PWM2 Guard Duration register 00000000

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	GUARD_DURATION	

100050B0 PWM2_SEND_DATA0 PWM2 Send Data0 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**SEND_DATA0[31:16]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM2 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

100050B4 **PWM2_SEND_DATA1** **PWM2 Send Data1 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM2 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

100050B8 **PWM2_WAVE_NUM** **PWM2 Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	WAVE_NUM	**The number by which PWM2 will generate from the pulse data repeatedly.** Note: If WAVE_NUM=0, the waveform generation will not stop until it is disabled.

100050BC **PWM2_DATA_WIDTH** **PWM2 Data Width register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[18:3]															

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	DATA_WIDTH	**The PWM2 pulse data width in the old PWM mode.**

100050C0 **PWM2_THRESH** **PWM2 Thresh register** **00000000**

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	THRESH	**The PWM2 pulse data high/low switching threshold in the old PWM mode.**

100050C4 **PWM2_SEND_WAVENUM** **PWM2 Send Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	SEND_WAVENUM	**The number by which PWM2 has already generated from the specified data source in the periodical mode.**

100050D0 **PWM3_CON** **PWM3 Control register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RESV0**															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	RESV1			CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RO			RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**RESERVED**
15	OLD_PWM_MODE	**Use old PWM mode** Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode). 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Note: Using old PWM mode also means periodical mode. So SRCSEL and MODE is ignored in this situation. Only old PWM mode with 32 KHz clock source (however could not work in the system sleep-mode).**
8	GUARD_VALUE	**PWM3 output value when guard time.**
7	IDLE_VALUE	**PWM3 output value when idle state.**
6:4	RESV1	**Select Random Generator mode**
3	CLKSEL	**Select PWM3 clock** 0: CLK= 100KHz 1: CLK= 40MHz
2:0	CLKDIV	**Select PWM3 clock scale.** 000: CLK Hz 001: CLK/2 Hz 010: CLK/4 Hz 011: CLK/8 Hz 100: CLK/16 Hz 101: CLK/32 Hz 110: CLK/64 Hz 111: CLK/128 Hz

100050D4 **PWM3_HDURATION** **PWM3 High Duration register** **00000001**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	HDURATION	**PWM3 pulse duration based on the current clock when PWM output is high. If duration =N, need to program N-1 in this register.**

Bit(s)	Name	Description
		Note: The duration of PWM must not be 0.

100050D8 **PWM3_LDURATION** **PWM3 Low Duration register** **00000001**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	LDURATION	**PWM3 pulse duration based on the current clock when PWM output is low. If duration =N, need to program N-1 in this register.** Note: The duration of PWM must not be 0.

100050DC **PWM3_GDURATION** **PWM3 Guard Duration register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	GUARD_DURATION	

100050F0 **PWM3_SEND_DATA0** **PWM3 Send Data0 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM3 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

100050F4 **PWM3_SEND_DATA1** **PWM3 Send Data1 register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM3 local buffer0 of pulse sequence data to be generated.** Note: This value should be written only in periodically FIFO mode. In other mode, this buffer is for internal memory access.

100050F8 **PWM3_WAVE_NUM** **PWM3 Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	WAVE_NUM	**The number by which PWM3 will generate from the pulse data repeatedly.** Note: If WAVE_NUM=0, the waveform generation will not stop until it is disabled.

100050FC **PWM3_DATA_WIDTH** **PWM3 Data Width register** **00000000**

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

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	DATA_WIDTH	**The PWM3 pulse data width in the old PWM mode.**

10005100 **PWM3_THRESH** **PWM3 Thresh register** **00000000**

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

Bit(s)	Name	Description
31:13	RESV	**RESERVED**
12:0	THRESH	**The PWM3 pulse data high/low switching threshold in the old PWM mode.**

10005104 **PWM3_SEND_WAVENUM** **PWM3 Send Wave Number register** **00000000**

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

Bit(s)	Name	Description
31:16	RESV	**RESERVED**
15:0	SEND_WAVENUM	**The number by which PWM3 has already generated from the specified data source in the periodical mode.**

1000520C **PWM_EN_STATUS** **PWM Enable Status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RESV[27:12]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RESV[11:0]**												**PWM3_EN_**	**PWM2_EN_**	**PWM1_EN_**	**PWM0_EN_**

													ST	**ST**	**ST**	**ST**
Type	RO												RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RESV	**RESERVED**
3	PWM3_EN_ST	**PWM3 enable status**
2	PWM2_EN_ST	**PWM2 enable status**
1	PWM1_EN_ST	**PWM1 enable status**
0	PWM0_EN_ST	**PWM0 enable status**

4.18 **Frame Engine**

4.18.1 Registers

SDM Changes LOG

Revision	Date	Author	Change Log
0.1	2013/5/27	PeterCT WU	Initialization

Module name: SDM Base address: (+10100000h)

Address	Name	Width	Register Function
10100800	TX_BASE_PTR_0	32	TX Ring #0 Base Pointer
10100804	TX_MAX_CNT_0	32	TX Ring #0 Maximum Count
10100808	TX_CTX_IDX_0	32	TX Ring #0 CPU pointer
1010080C	TX_DTX_IDX_0	32	TX Ring #0 DMA poitner
10100810	TX_BASE_PTR_1	32	TX Ring #1 Base Pointer
10100814	TX_MAX_CNT_1	32	TX Ring #1 Maximum Count
10100818	TX_CTX_IDX_1	32	TX Ring #1 CPU pointer
1010081C	TX_DTX_IDX_1	32	TX Ring #1 DMA poitner
10100820	TX_BASE_PTR_2	32	TX Ring #2 Base Pointer
10100824	TX_MAX_CNT_2	32	TX Ring #2 Maximum Count
10100828	TX_CTX_IDX_2	32	TX Ring #2 CPU pointer
1010082C	TX_DTX_IDX_2	32	TX Ring #2 DMA poitner
10100830	TX_BASE_PTR_3	32	TX Ring #3 Base Pointer
10100834	TX_MAX_CNT_3	32	TX Ring #3 Maximum Count
10100838	TX_CTX_IDX_3	32	TX Ring #3 CPU pointer
1010083C	TX_DTX_IDX_3	32	TX Ring #3 DMA poitner
10100900	RX_BASE_PTR_0	32	RX Ring #0 Base Pointer
10100904	RX_MAX_CNT_0	32	RX Ring #0 Maximum Count
10100908	RX_CRX_IDX_0	32	RX Ring #0 CPU pointer
1010090C	RX_DRX_IDX_0	32	RX Ring #0 DMA poitner
10100910	RX_BASE_PTR_1	32	RX Ring #1 Base Pointer
10100914	RX_MAX_CNT_1	32	RX Ring #1 Maximum Count
10100918	RX_CRX_IDX_1	32	RX Ring #1 CPU pointer
1010091C	RX_DRX_IDX_1	32	RX Ring #1 DMA poitner
10100A00	PDMA_INFO	32	PDMA Information

10100A04	PDMA_GLO_CFG	32	**PDMA Global Configuration**
10100A0C	DELAY_INT_CFG	32	**Delay Interrupt Configuration**
10100A10	FREEQ_THRES	32	**Free Queue Threshold**
10100A20	INT_STATUS	32	**Interrupt Status**
10100A28	INT_MASK	32	**Interrupt Mask**
10100A80	PDMA_SCH	32	**Scheduler Configuration for Q0&Q1**
10100A84	PDMA_WRR	32	**Scheduler Configuration for Q2&Q3**
10100C00	SDM_CON	32	**Switch DMA Control**
10100C04	SDM_RING	32	**Switch DMA Rx Ring**
10100C08	SDM_TRING	32	**Switch DMA TX Ring**
10100C0C	SDM_MAC_ADRL	32	**Switch MAC Address LSB**
10100C10	SDM_MAC_ADRH	32	**Switch MAC Address MSB**
10100D00	SDM_TPCNT	32	**Switch DMA Tx Packet Count**
10100D04	SDM_TBCNT	32	**Switch DMA TX Byte Count**
10100D08	SDM_RPCNT	32	**Switch DMA RX Packet Count**
10100D0C	SDM_RBCNT	32	**Switch DMA RX Byte Count**
10100D10	SDM_CS_ERR	32	**Switch DMA RX Checksum Error**

10100800 **TX_BASE_PTR_0** **TX Ring #0 Base Pointer** **00000000**

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

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

10100804 **TX_MAX_CNT_0** **TX Ring #0 Maximum Count** **00000000**

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

Bit(s)	Name	Description

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

10100808 **TX_CTX_IDX_0** **TX Ring #0 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1010080C **TX_DTX_IDX_0** **TX Ring #0 DMA poitner** **00000000**

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

Bit(s)	Name	Description
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

10100810 **TX_BASE_PTR_1** **TX Ring #1 Base Pointer** **00000000**

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

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

10100814 **TX_MAX_CNT** **TX Ring #1 Maximum Count** **0000000**

	1															0

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

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

10100818 **TX_CTX_IDX_1** **TX Ring #1 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1010081C **TX_DTX_IDX_1** **TX Ring #1 DMA poitner** **00000000**

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

Bit(s)	Name	Description
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

10100820 **TX_BASE_PTR_2** **TX Ring #2 Base Pointer** **00000000**

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

Name	**TX_BASE_PTR[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

10100824 **TX_MAX_CNT_2** **TX Ring #2 Maximum Count** **00000000**

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

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

10100828 **TX_CTX_IDX_2** **TX Ring #2 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1010082C **TX_DTX_IDX_2** **TX Ring #2 DMA poitner** **00000000**

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

Bit(s)	Name	Description
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

10100830 **TX_BASE_PTR_3** **TX Ring #3 Base Pointer** **00000000**

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

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

10100834 **TX_MAX_CNT_3** **TX Ring #3 Maximum Count** **00000000**

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

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

10100838 **TX_CTX_IDX_3** **TX Ring #3 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1010083C **TX_DTX_IDX_3** **TX Ring #3 DMA poitner** **00000000**

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

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

Bit(s)	Name	Description
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

10100900 **RX_BASE_PTR_0** **RX Ring #0 Base Pointer** **00000000**

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

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #0 (4-DW aligned address)**

10100904 **RX_MAX_CNT_0** **RX Ring #0 Maximum Count** **00000000**

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

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

10100908 **RX_CRX_IDX_0** **RX Ring #0 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next RXD CPU wants to use**

1010090C **RX_DRX_IDX_0** **RX Ring #0 DMA poitner** **00000000**

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

Bit(s)	Name	Description
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

10100910 **RX_BASE_PTR_1** **RX Ring #1 Base Pointer** **00000000**

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

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #0 (4-DW aligned address)**

10100914 **RX_MAX_CNT_1** **RX Ring #1 Maximum Count** **00000000**

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

Bit(s)	Name	Description
11:0	TX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

10100918 **RX_CRX_IDX_1** **RX Ring #1 CPU pointer** **00000000**

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

Bit(s)	Name	Description
11:0	TX_CTX_IDX	**Point to the next RXD CPU wants to use**

1010091C **RX_DRX_IDX_1** **RX Ring #1 DMA poitner** **00000000**

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

Bit(s)	Name	Description
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

10100A00 **PDMA_INFO** **PDMA Information** **1C000204**

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

Bit(s)	Name	Description
27:24	INDEX_WIDTH	**Point to the next RXD CPU wants to use**
23:16	BASE_PTR_WIDTH	**Base pointer width, x** Base_addr[31:32-x] is shared with all ring base adderss. Only ring #0 base address[31:32-x] field Is writabl. [note]: "0" means no bit of base_address is shared.
15:8	RX_RING_NUM	**Rx ring number**
7:0	TX_RING_NUM	**Tx ring number**

10100A04 __PDMA_GLO_CFG__ **PDMA Global Configuration** **00000050**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			HDR_SEG_LEN													
Type			RW													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									BIG_ENDIAN	TX_WB_DDONE	PDMA_BT_SIZE		RX_DMA_BUSY	RX_DMA_EN	TX_DMA_BUSY	TX_DMA_EN
Type									RW	RW	RW		RO	RW	RO	RW
Reset									0	1	0	1	0	0	0	0

Bit(s)	Name	Description
29:16	HDR_SEG_LEN	**Header Segment Length** Specify the header segment size in byte to supoprt RX header/payload scattering fucntion, when set to a non-zero value. When set to zero, the header/payload scattering feature is disabled.
7	BIG_ENDIAN	**Big endian** 0: PDMA will not do byte swapping for TX/RX packet header and payload 1: PDMA will do byte swaping for TX/RX packet header and payload
6	TX_WB_DDONE	0: Disable TX_DMA writing back DDONE into TXD 1: Enable TX_DMA writing back DDONE into TXD
5:4	PDMA_BT_SIZE	**The burst size of PDMA** 0: 4 DWORDs (16-bytes) 1: 8 DWORDs (32-bytes) 2: 16 DWORDs (64-bytes) 3: Reserved
3	RX_DMA_BUSY	0: RX_DMA is not busy 1: RX_DMA is busy
2	RX_DMA_EN	0: Diable RX_DMA (when disabled, RX_DMA will finish the current receiving packet, then stop) 1: Enable RX_DMA
1	TX_DMA_BUSY	0: TX_DMA is not busy 1: TX_DMA is busy
0	TX_DMA_EN	0: Disable TX_DMA (when disabled, TX_DMA will finish the current sending packet, then stop) 1: Enable TX_DMA

10100A0C __DELAY_INT_CFG__ **Delay Interrupt Configuration** **00000000**

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

	DLY_INT_EN															
Type	RW	RW							RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TXDLY_INT_EN	**Delay interrupt mechanism** 0: Disable TX delayed interrupt mechanism 1: Enable Tx delayed interrupt mechanism
30:24	TXMAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final TX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
23:16	TXMAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than TXMAX_PTIME x 20us or the number of pended TX_DONE is equal or greater than TXMAX_PINT 9see above), an final TX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.
15	RXDLY_INT_EN	0: Disable Rx delayed interrupt mechanism 1: Enable Rx delayed interrupt mechanism
14:8	RXMAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final RX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
7:0	RXMAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than RXMAX_PTIME x 20us or the number of pended RX_DONE is equal or greater than RXMAX_PINT 9see above), an finalRX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.

10100A10 **FREEQ_THRES** **Free Queue Threshold** **00000002**

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

Bit(s)	Name	Description
3:0	FREEQ_THRES	**Rx free queue threshold** PDMA will stop DMA interface when left RX descriptors reach this threshold

10100A20 **INT_STATUS** **Interrupt Status** **0000000**

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	TX_COHERENT	TX_DLY_INT											RX_DONE_INT1	RX_DONE_INT0
Type	W1C	W1C	W1C	W1C											W1C	W1C
Reset	0	0	0	0											0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													TX_DONE_INT3	TX_DONE_INT2	TX_DONE_INT1	TX_DONE_INT0
Type													W1C	W1C	W1C	W1C
Reset													0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**RX_DMA finds data coherent event while checking ddone bit.**
30	RX_DLY_INT	**Summary of the whole PDMA Rx related interrupts.**
29	TX_COHERENT	**TX_DMA finds data coherent event while checking ddone bit.**
28	TX_DLY_INT	**Summary of the whole PDMA Tx related interrupts.**
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt**
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt**
3	TX_DONE_INT3	**Tx ring #3 packet transmit interrupt**
2	TX_DONE_INT2	**Tx ring #2 packet transmit interrupt**
1	TX_DONE_INT1	**Tx ring #1 packet transmit interrupt**
0	TX_DONE_INT0	**Tx ring #0 packet transmit interrupt**

10100A28 **INT_MASK** **Interrupt Mask** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	TX_COHERENT	TX_DLY_INT											RX_DONE_INT1	RX_DONE_INT0
Type	RW	RW	RW	RW											RW	RW
Reset	0	0	0	0											0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													TX_DONE_INT3	TX_DONE_INT2	TX_DONE_INT1	TX_DONE_INT0
Type													RW	RW	RW	RW
Reset													0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**Interrupt enable for RX_DMA data coherent vent** 0: Disable interrupt

Bit(s)	Name	Description
		1: Enable interrupt
30	RX_DLY_INT	**Summary of the whole PDMA Rx related interrupts.** 0: Disable interrupt 1: Enable interrupt
29	TX_COHERENT	**Interrupt enable for TX_DMA data coherent vent** 0: Disable interrupt 1: Enable interrupt
28	TX_DLY_INT	**Summary of the whole PDMA Tx related interrupts.** 0: Disable interrupt 1: Enable interrupt
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
3	TX_DONE_INT3	**Tx ring #3 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
2	TX_DONE_INT2	**Tx ring #2 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
1	TX_DONE_INT1	**Tx ring #1 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
0	TX_DONE_INT0	**Tx ring #0 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt

10100A80 **PDMA_SCH** **Scheduler Configuration for Q0&Q1** **00000000**

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

Bit(s)	Name	Description
25:24	SCH_MODE	**Scheduling Mode** 00: WRR 01: Strict priority, Q3>Q2,Q1>Q0 10: Mixed mode, Q3>WRR(Q2,Q1,Q0) 11: Mixed mode, Q3>Q2>WRR(Q1,Q0)

10100A84 **PDMA_WRR** **Scheduler Configuration for Q2&Q3** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		SCH_WT_Q3				SCH_WT_Q2				SCH_WT_Q1				SCH_WT_Q0		
Type		RW				RW				RW				RW		
Reset		0	0	0		0	0	0		0	0	0		0	0	0

Bit(s)	Name	Description
14:12	SCH_WT_Q3	**Scheduling Weight of TX Q3**
10:8	SCH_WT_Q2	**Scheduling Weight of TX Q2**
6:4	SCH_WT_Q1	**Scheduling Weight of TX Q1**
2:0	SCH_WT_Q0	**Scheduling Weight of TX Q0**

10100C00 **SDM_CON** **Switch DMA Control** **00078100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								PDMA_FC	PORT_MAP	LOOP_EN	TCO_81xx	UN_DROP_EN	UDPCS	TCPCS	IPCS
Type	RO								RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXT_VLAN															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23	PDMA_FC	**TX PDMA Flow Cotnrol Enable** When this bit is set, the downstream fow control is enabled on PDMA 4 TX Ring (SDM_TRGING) 0: Disable 1: Enable
22	PORT_MAP	**RX Ring Selection** The received frame will be collected into the corresponding PDMA RX Ring based on the source port priority tag. 0: Priority Tag (SDMRRING[7:0]) 1: Source Port (SDM_RRING[12:8])
21	LOOP_EN	**Frame Engine Loop-back Mode Enable**
20	TCO_81xx	**Special tag Recongization Enable** When this bit is set, PDI(0x81xx) is recognized by the first byte (0x81) only. The second byte could be used for the specilqa purpose like the incoming source port.
19	UN_DROP_EN	**Drop Unknonwn MAC Address** 0: Disable

Bit(s)	Name	Description
		1: Enable
18	UDPCS	**UDP Packet Checksum RX Offload Enable** 0: disable, checksum result is showed on RX descriptor 1: enable, drop checksum error packet
17	TCPCS	**TCP Packet Checksum RX Offload Enable** 0: disable, checksum result is showed on RX descriptor 1: enable, drop checksum error packet
16	IPCS	**IP Header Checksum RX Offload Enable** 0: disable, checksum result is showed on RX descriptor 1: enable, drop checksum error packet
15:0	EXT_VLAN	**Outer VLAN Protocol ID** The specific vlaue is used to recognize the outer VLAN protocol ID only. Per inner VLAN or the general VLAN-tagged frame, the value PID=0x8100 is the uniqe protocol ID.

10100C04 **SDM_RING** **Switch DMA Rx Ring** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**REV0**												**QUE3_RING_FC**	**QUE2_RING_FC**	**QUE1_RING_FC**	**QUE0_RING_FC**
Type	RO												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				**PORT4_RING**	**PORT3_RING**	**PORT2_RING**	**PORT1_RING**	**PORT0_RING**	**PRI7_RING**	**PRI6_RING**	**PRI5_RING**	**PRI4_RING**	**PRI3_RING**	**PRI2_RING**	**PRI1_RING**	**PRI10_RING**
Type				RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:20	REV0	**Reserved**
19	QUE3_RING_FC	**Pause Switch Queue 3 by RX Ring##** When RX Ring# reaches the reserved free threshold(FREEQ_THRES), the queue 3 to CPU will be paused. 1: RX Ring #0 0: RX Ring #1
18	QUE2_RING_FC	**Pause Switch Queue 2 by RX Ring##** When RX Ring# reaches the reserved free threshold(FREEQ_THRES), the queue 3 to CPU will be paused. 1: RX Ring #0 0: RX Ring #1
17	QUE1_RING_FC	**Pause Switch Queue 1 by RX Ring##** When RX Ring# reaches the reserved free threshold(FREEQ_THRES), the queue 3 to CPU will be paused. 1: RX Ring #0 0: RX Ring #1
16	QUE0_RING_FC	**Pause Switch Queue 0 by RX Ring##**

Bit(s)	Name	Description
		When RX Ring# reaches the reserved free threshold(FREEQ_THRES), the queue 3 to CPU will be paused. 1: RX Ring #0 0: RX Ring #1
12	PORT4_RING	**Source Port 4 to RX Ring##** The received frames from the source port 4 will be sent to RX Ring# [Note] To use the source port, the special tag between FE and SW should be enabled. 1: RX Ring #0 0: RX Ring #1
11	PORT3_RING	**Source Port 3 to RX Ring##** The received frames from the source port 4 will be sent to RX Ring# [Note] To use the source port, the special tag between FE and SW should be enabled. 1: RX Ring #0 0: RX Ring #1
10	PORT2_RING	**Source Port 2 to RX Ring##** The received frames from the source port 4 will be sent to RX Ring# [Note] To use the source port, the special tag between FE and SW should be enabled. 1: RX Ring #0 0: RX Ring #1
9	PORT1_RING	**Source Port 1 to RX Ring##** The received frames from the source port 4 will be sent to RX Ring# [Note] To use the source port, the special tag between FE and SW should be enabled. 1: RX Ring #0 0: RX Ring #1
8	PORT0_RING	**Source Port 0 to RX Ring##** The received frames from the source port 4 will be sent to RX Ring# [Note] To use the source port, the special tag between FE and SW should be enabled. 1: RX Ring #0 0: RX Ring #1
7	PRI7_RING	**Priority 7 to RX Ring##** The received frames with priority tag 7 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
6	PRI6_RING	**Priority 6 to RX Ring##** The received frames with priority tag 6 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
5	PRI5_RING	**Priority 5 to RX Ring##** The received frames with priority tag 5 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
4	PRI4_RING	**Priority 4 to RX Ring##** The received frames with priority tag 4 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
3	PRI3_RING	**Priority 3 to RX Ring##** The received frames with priority tag 3 will be sent to RX Ring# 1: RX Ring #0

Bit(s)	Name	Description
		0: RX Ring #1
2	PRI2_RING	**Priority 2 to RX Ring##** The received frames with priority tag 2 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
1	PRI1_RING	**Priority 1to RX Ring##** The received frames with priority tag 1 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1
0	PRI10_RING	**Priority 0 to RX Ring##** The received frames with priority tag 0 will be sent to RX Ring# 1: RX Ring #0 0: RX Ring #1

10100C08 **SDM_TRING** **Switch DMA TX Ring** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RING3_WAN_FC				RING2_WAN_FC				RING1_WAN_FC				RING0_WAN_FC			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RING3_LAN_FC				RING2_LAN_FC				RING1_LAN_FC				RING0_LAN_FC			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	RING3_WAN_FC	**Pause TX Ring 3 by WAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: WAN port Queue#3 Bit.2: WAN port Queue#2 Bit.1: WAN port Queue#1 Bit.0: WAN port Queue#0
27:24	RING2_WAN_FC	**Pause TX Ring 2 by WAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: WAN port Queue#3 Bit.2: WAN port Queue#2 Bit.1: WAN port Queue#1 Bit.0: WAN port Queue#0
23:20	RING1_WAN_FC	**Pause TX Ring 1 by WAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: WAN port Queue#3 Bit.2: WAN port Queue#2 Bit.1: WAN port Queue#1 Bit.0: WAN port Queue#0
19:16	RING0_WAN_FC	**Pause TX Ring 0 by WAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: WAN port Queue#3 Bit.2: WAN port Queue#2

Bit(s)	Name	Description
		Bit.1: WAN port Queue#1 Bit.0: WAN port Queue#0
15:12	RING3_LAN_FC	**Pause TX Ring 3 by LAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: LAN port Queue#3 Bit.2: LAN port Queue#2 Bit.1: LAN port Queue#1 Bit.0: LAN port Queue#0
11:8	RING2_LAN_FC	**Pause TX Ring 2 by LAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: LAN port Queue#3 Bit.2: LAN port Queue#2 Bit.1: LAN port Queue#1 Bit.0: LAN port Queue#0
7:4	RING1_LAN_FC	**Pause TX Ring 1 by LAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: LAN port Queue#3 Bit.2: LAN port Queue#2 Bit.1: LAN port Queue#1 Bit.0: LAN port Queue#0
3:0	RING0_LAN_FC	**Pause TX Ring 0 by LAN Port** TX Ring# will be paused when the corresponding switch egress queue on WAN port is congested. Bit.3: LAN port Queue#3 Bit.2: LAN port Queue#2 Bit.1: LAN port Queue#1 Bit.0: LAN port Queue#0

10100C0C **SDM_MAC_ADRL** **Switch MAC Address LSB** **00000000**

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

Bit(s)	Name	Description
31:0	MAC_ADDR_LSB	**MAC Address bit[31:0]**

10100C10 **SDM_MAC_ADRH** **Switch MAC Address MSB** **00000000**

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

Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**MAC_ADDR_MSB**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	MAC_ADDR_MSB	**MAC Address bit[47:32]**

10100D00 SDM_TPCNT Switch DMA Tx Packet Count 00000000

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

Bit(s)	Name	Description
31:0	TX_PCNT	**Transmit Packet Count**

10100D04 SDM_TBCNT Switch DMA TX Byte Count 00000000

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

Bit(s)	Name	Description
31:0	TX_BCNT	**Transmit Byte Count**

10100D08 SDM_RPCNT Switch DMA RX Packet Count 00000000

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

Bit(s)	Name	Description
31:0	RX_PCNT	**Receive Packet Count**

10100D0C **SDM_RBCNT** **Switch DMA RX Byte Count** **00000000**

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

Bit(s)	Name	Description
31:0	RX_BCNT	**Receive Byte Count**

10100D10 **SDM_CS_ERR** **Switch DMA RX Checksum Error** **00000000**

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

Bit(s)	Name	Description
31:0	CS_ERR_CNT	**Receive Checksum Error Count**

4.19 **Switch Controller**

4.19.1 Registers

ESW Changes LOG

Revision	Date	Author	Change Log
0.1	2013/5/29	PeterCT WU	Initialization

Module name: ESW Base address: (+10110000h)

Address	Name	Width	Register Function
10110000	ISR	32	Interrupt Status
10110004	IMR	32	Interrupt Mask
10110008	FCT0	32	Flow Control Threshold 0
1011000C	FCT1	32	Flow Control Threshold 1
10110010	PFC0	32	Priority Flow Control 0
10110014	PFC1	32	Priority Flow Control 1
10110018	PFC2	32	Priority Flow Control 2
1011001C	GQS0	32	Global Queue Status 0
10110020	GQS1	32	Global Queue Status 1
10110024	ATS	32	Address Table Search
10110028	ATS0	32	Address Table Status 0
1011002C	ATS1	32	Address Table Status 1
10110030	ATS2	32	Address Table Status 2
10110034	WMAD0	32	WT_MAC_AD0
10110038	WMAD1	32	WT_MAC_AD1
1011003C	WMAD2	32	WT_MAC_AD2
10110040	PVIDC0	32	PVID Configuration 0
10110044	PVIDC1	32	PVID Configuration 1
10110048	PVIDC2	32	PVID Configuration 2
1011004C	PVIDC3	32	PVID Configuration 3
10110050	VLANI0	32	VLAN Identifier 0
10110054	VLANI1	32	VLAN Identifier 1
10110058	VLANI2	32	VLAN Identifier 2
1011005C	VLANI3	32	VLAN Identifier 3
10110060	VLANI4	32	VLAN Identifier 4
10110064	VLANI5	32	VLAN Identifier 5
10110068	VLANI6	32	VLAN Identifier 6
1011006C	VLANI7	32	VLAN Identifier 7
10110070	VMSC0	32	VLAN Member Port Configuration 0
10110074	VMSC1	32	VLAN Member Port Configuration 1
10110078	VMSC2	32	VLAN Member Port Configuration 2
1011007C	VMSC3	32	VLAN Member Port Configuration 3
10110080	POA	32	Port Ability Offset

10110084	FPA	32	Force Port4 - Port0 Ability
10110088	PTS	32	Port Status
1011008C	SOCPC	32	SoC Port Control
10110090	POC0	32	Port Control 0
10110094	POC1	32	Port Control 1
10110098	POC2	32	Port Control 2
1011009C	SGC	32	Switch Global Control
101100A0	STRT	32	Switch Reset
101100A4	LEDP0	32	LED Port0
101100A8	LEDP1	32	LED Port1
101100AC	LEDP2	32	LED Port2
101100B0	LEDP3	32	LED Port3
101100B4	LEDP4	32	LED Port4
101100B8	WDTR	32	Watch Dog Trigger Reset
101100BC	DES	32	Debug Signal
101100C0	PCR0	32	PHY Control Register 0
101100C4	PCR1	32	PHY Control Register 1
101100C8	FPA1	32	Force P5P6 Ability
101100CC	FCT2	32	Flow Control Threshold 2
101100D0	QSS0	32	Queue Status 0
101100D4	QSS1	32	Queue Status 1
101100D8	DEC	32	Debug Control
101100DC	MTI	32	Memory Test Information
101100E0	PPC	32	Packet Counter
101100E4	SGC2	32	Switch Global Control 2
101100E8	P0PC	32	Port 0 Packet Counter
101100EC	P1PC	32	Port 1 Packet Counter
101100F0	P2PC	32	Port 2 Packet Counter
101100F4	P3PC	32	Port 3 Packet Counter
101100F8	P4PC	32	Port 4 Packet Counter
101100FC	P5PC	32	Port 5 Packet Counter
10110100	VUB0	32	VLAN Untag Block 0
10110104	VUB1	32	VLAN Untag Block 1
10110108	VUB2	32	VLAN Untag Block 2
1011010C	VUB3	32	VLAN Untag Block 3
10110110	BMU_CTRL	32	BC/MC/UN Rate Limit Control
10110114	BMU_LMT_NUM1	32	BC/MC/UN Rate Limit Frame Number
10110118	BMU_LMT_NUM2	32	BC/MC/UN Rate Limit Frame Number
1011011C	P01_ING_CTRL	32	Port 0&1 Ingress Rate Limit Control
10110120	P23_ING_CTRL	32	Port 2&3 Ingress Rate Limit Control
10110124	P45_ING_CTRL	32	Port 4&5 Ingress Rate Limit Control
10110128	P0_ING_THRES	32	Port 0 Ingress Rate Limit Threshold
1011012C	P1_ING_THRE	32	Port 1 Ingress Rate Limit Threshold

	S		
10110130	P2_ING_THRES	32	**Port 2 Ingress Rate Limit Threshold**
10110134	P3_ING_THRES	32	**Port 3 Ingress Rate Limit Threshold**
10110138	P4_ING_THRES	32	**Port 4 Ingress Rate Limit Threshold**
1011013C	P5_ING_THRES	32	**Port 5 Ingress Rate Limit Threshold**
10110140	P01_EG_CTRL	32	**Port 0/1 Egress Rate Limit Control**
10110144	P23_EG_CTRL	32	**Port 2/3 Egress Rate Limit Control**
10110148	P45_EG_CTRL	32	**Port 4/5 Egress Rate Limit Control**
1011014C	PCRI	32	**Packet Counter Recycle Indication**
10110150	P0TPC	32	**Port 0 TX Packet Counter**
10110154	P1TPC	32	**Port 1 TX Packet Counter**
10110158	P2TPC	32	**Port 2 TX Packet Counter**
1011015C	P3TPC	32	**Port 3 TX Packet Counter**
10110160	P4TPC	32	**Port 4 TX Packet Counter**
10110164	P5TPC	32	**Port 5 TX Packet Counter**
10110168	LEDC	32	**LED Control**

10110000 ISR Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0		WATCHDOG1_TMR_EXPIRED	WATCHDOG0_TMR_EXPIRED	HAS_INTRUDER	PORT_ST_CHG	BC_STORM	MUST_DROP_LAN	GLOBAL_QUE_FULL			LAN_QUE_FULL_6	LAN_QUE_FULL_5	LAN_QUE_FULL_4	LAN_QUE_FULL_3	LAN_QUE_FULL_2
Type	RO		W1C	W1C	W1C	W1C	W1C	W1C	W1C			W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0			0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LAN_QUE_FULL_1	LAN_QUE_FULL_0														
Type	W1C	W1C														
Reset	0	0														

Bit(s)	Name	Description
31:30	REV0	**Reserved**
29	WATCHDOG1_TMR_EXPIRED	**P5 no transmit packet alert.** This bit indicating that P5 don't transmit packet for 3 seconds when P5 need to transmit packet. Write one clear. [Note] This feature is only valid when port 5 Giga MAC is implemented.

Bit(s)	Name	Description
28	WATCHDOG0_TMR_EXPIRED	**Abnormal Alert** This bit indicating that global queue block counts is less than buf_starvation_th for 3 seconds. Write one clear.
27	HAS_INTRUDER	**Intruder Alert** This bit indicating that an unsecured packet is coming into a secured port. Write one clear.
26	PORT_ST_CHG	**Port status change** Any port from link status change. Write one clear.
25	BC_STORM	**BC storm** The device is undergoing broadcast storm. Write one clear.
24	MUST_DROP_LAN	**Queue exhausted** The global queue is used up and all packets are dropped. Write one clear.
23	GLOBAL_QUE_FULL	**Global Queue Full.** Write one clear.
20	LAN_QUE_FULL_6	**Port 6 out queue full. Write one clear.** [Note]: This feature is only valid when port 5 Giga MAC is implemented.
19	LAN_QUE_FULL_5	**Port 5 out queue full. Write one clear.**
18	LAN_QUE_FULL_4	**Port 4 out queue full. Write one clear.**
17	LAN_QUE_FULL_3	**Port 3 out queue full. Write one clear.**
16	LAN_QUE_FULL_2	**Port 2 out queue full. Write one clear.**
15	LAN_QUE_FULL_1	**Port 1 out queue full. Write one clear.**
14	LAN_QUE_FULL_0	**Port 0 out queue full. Write one clear.**

10110004 **IMR** **Interrupt Mask** **FFFFFFFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			WATCHDOG1_TMR_EXPIRED	WATCHDOG0_TMR_EXPIRED	HAS_INTRUDER	PORT_ST_CHG	BC_STORM	MUST_DROP_LAN	GLOBAL_QUE_FULL	REV1		LAN_QUE_FULL_6	LAN_QUE_FULL_5	LAN_QUE_FULL_4	LAN_QUE_FULL_3	LAN_QUE_FULL_2
Type			RW	RW	RW	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW
Reset			1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LAN_QUE_FULL_1	LAN_QUE_FULL_0														
Type	RW	RW														
Reset	1	1														

Bit(s)	Name	Description
29	WATCHDOG1_TMR_EXPIRED	**P5 no transmit packet alert.** This bit indicating that P5 don't transmit packet for 3 seconds when P5 need

Bit(s)	Name	Description
		to transmit packet. Write one clear. [Note]: This feature is only valid when port 5 Giga MAC is implemented.
28	WATCHDOG0_TMR_EXPIRED	**Abnormal Alert** This bit indicating that global queue block counts is less than buf_starvation_th for 3 seconds. Write one clear.
27	HAS_INTRUDER	**Intruder Alert** This bit indicating that an unsecured packet is coming into a secured port. Write one clear.
26	PORT_ST_CHG	**Port status change** Any port from link status change. Write one clear.
25	BC_STORM	**BC storm** The device is undergoing broadcast storm. Write one clear.
24	MUST_DROP_LAN	**Queue exhausted** The global queue is used up and all packets are dropped. Write one clear.
23	GLOBAL_QUE_FULL	**Global Queue Full.** Write one clear.
22:21	REV1	**Port 6 out queue full. Write one clear.** [Note]: This feature is only valid when port 5 Giga MAC is implemented.
20	LAN_QUE_FULL_6	**Port 6 out queue full. Write one clear.** [Note]: This feature is only valid when port 5 Giga MAC is implemented.
19	LAN_QUE_FULL_5	**Port 5 out queue full. Write one clear.**
18	LAN_QUE_FULL_4	**Port 4 out queue full. Write one clear.**
17	LAN_QUE_FULL_3	**Port 3 out queue full. Write one clear.**
16	LAN_QUE_FULL_2	**Port 2 out queue full. Write one clear.**
15	LAN_QUE_FULL_1	**Port 1 out queue full. Write one clear.**
14	LAN_QUE_FULL_0	**Port 0 out queue full. Write one clear.**

10110008 FCT0 Flow Control Threshold 0 FFC86E5A

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

Bit(s)	Name	Description
31:24	FC_RLS_TH	**Flow Control Release Threshold** Flow control will be disabled when the global queue block counts is greater than the release threshold
23:16	FC_SET_TH	**Flow Control Set Threshold** Flow control will be enabled when the global queue block counts is less than the set threshold
15:8	DRO_RLS_TH	**Drop Release Threshold** Switch will stop dropping packets when the global queue block counts is

Bit(s)	Name	Description
		greater than the drop-release threshold
7:0	DROP_SET_TH	**Drop Set Threshold** Switch will start dropping packets when the global queue block counts is less than the drop-set threshold.

1011000C **FCT1** **Flow Control Threshold 1** **00000014**

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

Bit(s)	Name	Description
7:0	PORT_TH	**Per Port Output Threshold** When the global queue reaches the flow control or drop threshold on register FCT0, per port output threshold will be checked to enable flow-control or packet-dop depending on per queue minimum reserved blocks of the register PFC2.

10110010 **PFC0** **Priority Flow Control 0** **0F000000**

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

Bit(s)	Name	Description
27:24	MTCC_LMT	**MTCC LIMIT** The maximum Back-off count limit to drop excessive collision packets.
22:16	TURN_OFF_FC	**Turn off FC When Receiving High Packet** Auto-turn-off FC when the programmed ports receive one of the highest priority packet. 0: DIsable 1: Enable
15:12	VO_NUM	**The proportional number of WRR for Voice Queue** After transmit exactly the number of packets then proceed to next queue. If equal to 0, only this queue is forced to the strict priority mode
11:8	CL_NUM	**The proportional number of WRR for Control-Load Queue** After transmit exactly the number of packet then proceed to next queue.
7:4	BE_NUM	**The proportional number of WRR for Best-Effort Queue**

Bit(s)	Name	Description
3:0	BK_NUM	After transmit exactly the number of packet then proceed to next queue. **The proportional number of WRR for Background Queue After transmit exactly the number of packet then proceed to next queue.**

10110014 **PFC1** **Priority Flow Control 1** **00001555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**CPU_USE_Q1_EN**	**EN_TOS**							**IGMP_TO_CPU**	**EN_VLAN**						
Type	RW	RW							RW	RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**PRIORITY_OPTION**		**PORT_PRI 6**		**PORT_PRI 5**		**PORT_PRI 4**		**PORT_PRI 3**		**PORT_PRI 2**		**PORT_PRI 1**		**PORT_PRI 0**	
Type	RW		RW		RW		RW		RW		RW		RW		RW	
Reset	0		0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31	CPU_USE_Q1_EN	**CPU Port only use q1 enable** 0: default priority resolution 1: packets forwarded to CPU port uses Best-Effor Queue
30:24	EN_TOS	**Port6 ~ port0 TOS_en.** Check TOS field of IP packets for priority resolution. [Note] Port 5 function is only vlaid when port 5 Giga MAC is implemented 0: Disable 1: Enable
23	IGMP_TO_CPU	**IGMP forward to CPU enable** 0: IGMP message will be floode to all ports 1: IGMP message will be forwarded to CPU port only.
22:16	EN_VLAN	**Enable per port VLAN-tag VID membership and priority tag check.** [Note] Port 5 function is only vlaid when port 5 Giga MAC is implemented 0: disable. 1: enable
15	PRIORITY_OPTION	**Priority Resolution Option** 0: 802.1p -> TOS -> Per port 1: TOS -> 802.1p -> Per port
13:12	PORT_PRI6	**Port priority** By setting this register to assign per port's default priority queue.
11:10	PORT_PRI5	**Port priority** By setting this register to assign per port's default priority queue. [Note] This feature is only valid when port 6 Giga MAC is implemented
9:8	PORT_PRI4	**Port priority** By setting this register to assign per port's default priority queue.

Bit(s)	Name	Description
7:6	PORT_PRI3	**Port priority** By setting this register to assign per port's default priority queue.
5:4	PORT_PRI2	**Port priority** By setting this register to assign per port's default priority queue.
3:2	PORT_PRI1	**Port priority** By setting this register to assign per port's default priority queue.
1:0	PORT_PRI0	**Port priority** By setting this register to assign per port's default priority queue.

10110018 **PFC2** **Priority Flow Control 2** **03030303**

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

Bit(s)	Name	Description
31:24	PRI_TH_VO	**Voice Threshold (Highest Priority)** The minimum reserved packet block count which outout queue can store when the flow-control /drop threshold of registers FTC0 and FCT1 is reached. If the queued blocks exceed the threshold, the incoming packet will be paused or dropped.
23:16	PRI_TH_CL	**Control Load Threshold** The minimum reserved packet block count which outout queue can store when the flow-control/drop threshold of registers FTC0 and FCT1 is reached. If the queued blocks exceed the threshold, the incoming packet will be paused or dropped.
15:8	PRI_TH_BE	**Best Effort threshold** The minimum reserved packet block count which outout queue can store when the flow-control/drop threshold of registers FTC0 and FCT1 is reached. If the queued blocks exceed the threshold, the incoming packet will be paused or dropped.
7:0	PRI_TH_BK	**Background Threshold (Lowest Priority)** The minimum reserved packet block count which outout queue can store when the flow-control/drop threshold of registers FTC0 and FCT1 is reached. If the queued blocks exceed the threshold, the incoming packet will be paused or dropped.

1011001C **GQS0** **Global Queue Status 0** **FA41016E**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PRI7_QUE		PRI6_QUE		PRI5_QUE		PRI4_QUE		PRI3_QUE		PRI2_QUE		PRI1_QUE		PRI0_QUE	
Type	RW		RW		RW		RW		RW		RW		RW		RW	
Reset	1	1	1	1	1	0	1	0	0	1	0	0	0	0	0	1

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

Bit(s)	Name	Description
31:30	PRI7_QUE	**Queue mapping for Priority Tag #7**
29:28	PRI6_QUE	**Queue mapping for Priority Tag #6**
27:26	PRI5_QUE	**Queue mapping for Priority Tag #5**
25:24	PRI4_QUE	**Queue mapping for Priority Tag #4**
23:22	PRI3_QUE	**Queue mapping for Priority Tag #3**
21:20	PRI2_QUE	**Queue mapping for Priority Tag #2**
19:18	PRI1_QUE	**Queue mapping for Priority Tag #1**
17:16	PRI0_QUE	**Queue mapping for Priority Tag #0**
8:0	EMPTY_CNT	**Global Queue Block Counts** This field indicates the number of block count left in the global free queue.

10110020 GQS1 Global Queue Status 1 00000000

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

Bit(s)	Name	Description
31:24	OUTQUE_FULL_VO	**Congested Voice Queue** The corresponding queue is congested
23:16	OUTQUE_FULL_CL	**Congested Control Load Queue** The corresponding queue is congested
15:8	OUTQUE_FULL_BE	**Congested Best Effort Queue** The corresponding queue is congested
7:0	OUTQUE_FULL_BK	**Congested Background Queue** The corresponding queue is congested

10110024 ATS Address Table Search 00000004

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

														IDLE	_NXT_ADDR	ARCH_ADDR
Type														RO	W1C	W1C
Reset														1	0	0

Bit(s)	Name	Description
2	AT_LKUP_IDLE	**Address Lookup Idle** This field indicates that Adress Table engine is in IDLE state.
1	SEARCH_NXT_ADDR	**Search For The Next Address (Self_Clear)**
0	BEGIN_SEARCH_ADDR	**Start Searching The Address Table (Self_Clear)**

10110028 ATS0 Address Table Status 0 00000000

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

Bit(s)	Name	Description
31:22	HASH_ADD_LU	**Address table lookup address**
18:12	R_PORT_MAP	**Port map** The MAC existing in the bit =1.
10:7	R_VLD	**VLAN index**
6:4	R_AGE_FIELD	**Aging field**
2	R_MC_INGRESS	**MC Ingress**
1	AT_TABLE_END	**Search to the end of address table**
0	SEARCH_RDY	**Data is ready (read clear)**

1011002C ATS1 Address Table Status 1 00000000

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

Name	MAC_AD_SER0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	MAC_AD_SER0	**Read MAC Address [15:0]**

10110030 **ATS2** **Address Table Status 2** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAC_AD_SER0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAC_AD_SER0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAC_AD_SER0	**Read MAC Address [31:16]**

10110034 **WMAD0** **WT_MAC_AD0** **00080000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	HASH_ADD_LU												AT_CFG_IDLE	W_PORT_MAP[6:4]		
Type	RO												RO	RW		
Reset	0	0	0	0	0	0	0	0	0	0			1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W_PORT_MAP[3:0]					W_INDEX				W_AGE_FIELD			SA_FILTER	W_MC_INGRESS	W_MAC_DONE	W_MAC_CMD
Type	RW					RW				RW			RW	RW	RO	WO
Reset	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:22	HASH_ADD_LU	**Address table configuration address**
19	AT_CFG_IDLE	**Address Table Configuration SM IDLE**
18:12	W_PORT_MAP	**Write Port map**
10:7	W_INDEX	**VLAN index** 0: VLAN 0 1-14: ... 15: VLAN 15
6:4	W_AGE_FIELD	**Write Aging field** 3'b111: static address,

Bit(s)	Name	Description
		3'b001 - 3'b110: the entry is valid and will be aged out 2'b000: default, entry is invalid
3	SA_FILTER	**SA_FILTER** 0: default 1: The corresponding packet will be dropped when the SA is matched
2	W_MC_INGRESS	**Write MC Ingress**
1	W_MAC_DONE	**MAC Write Done** 0: default 1: MAC address write OK (read clear)
0	W_MAC_CMD	**MAC Address write Command** 0: default 1: the MAC write data is ready and write to MAC table now(self_clear)

10110038 **WMAD1** **WT_MAC_AD1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W_MAC_15_0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	W_MAC_15_0	**Write MAC Address[15:0]**

1011003C **WMAD2** **WT_MAC_AD2** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W_MAC_47_16[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W_MAC_47_16[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W_MAC_47_16	**Write MAC Address[47:16]**

10110040 **PVIDC0** **PVID Configuration 0** **00001001**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									P1_PVID[11:4]							
Type									RW							

Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**P1_PVID[3:0]**				**P0_PVID**											
Type	RW				RW											
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
23:12	P1_PVID	**Port1 PVID Setting**
11:0	P0_PVID	**Port0 PVID Setting**

10110044 PVIDC1 PVID Configuration 1 00001001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									**P3_PVID[11:4]**							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**P3_PVID[3:0]**				**P2_PVID**											
Type	RW				RW											
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
23:12	P3_PVID	**Port3 PVID Setting**
11:0	P2_PVID	**Port2 PVID Setting**

10110048 PVIDC2 PVID Configuration 2 00001001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									**P5_PVID[11:4]**							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**P5_PVID[3:0]**				**P4_PVID**											
Type	RW				RW											
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
23:12	P5_PVID	**Port5 PVID Setting** [Note] This feature is only valid when port 5 Giga MAC is implemented.
11:0	P4_PVID	**Port4 PVID Setting**

1011004C PVIDC3 PVID Configuration 3 75020001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		**QUE3_PRIT**				**QUE2_PRIT**				**QUE1_PRIT**				**QUE0_PRIT**		
Type		RW				RW				RW				RW		
Reset		1	1	1		1	0	1		0	0	0		0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name					P6_PVID											
Type					RW											
Reset					0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
30:28	QUE3_PRIT	**Priority Tag Egress Mapping for Voice Queue#3**
26:24	QUE2_PRIT	**Priority Tag Egress Mapping for Control Load Queue#2**
22:20	QUE1_PRIT	**Priority Tag Egress Mapping for Best Effort Queue#1**
18:16	QUE0_PRIT	**Priority Tag Egress Mapping for Back Ground Queue#0**
11:0	P6_PVID	**Port6 PVID Setting**

10110050 **VLANI0** **VLAN Identifier 0** **0000200 1**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID1[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID1[3:0]				VID0											
Type	RW				RW											
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
23:12	VID1	**VLAN Field Identifier for VLAN 1**
11:0	VID0	**VLAN Field Identifier for VLAN 0**

10110054 **VLANI1** **VLAN Identifier 1** **0000400 3**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID3[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID3[3:0]				VID2											
Type	RW				RW											
Reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Name	Description
23:12	VID3	**VLAN Field Identifier for VLAN 3**
11:0	VID2	**VLAN Field Identifier for VLAN 2**

10110058 **VLANI2** **VLAN Identifier 2** **0000600 5**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID5[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID5[3:0]				VID4											
Type	RW				RW											
Reset	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	1

Bit(s)	Name	Description
23:12	VID5	**VLAN Field Identifier for VLAN 5**
11:0	VID4	**VLAN Field Identifier for VLAN 4**

1011005C **VLANI3** **VLAN Identifier 3** **00008007**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID7[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID7[3:0]				VID6											
Type	RW				RW											
Reset	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

Bit(s)	Name	Description
23:12	VID7	**VLAN Field Identifier for VLAN 7**
11:0	VID6	**VLAN Field Identifier for VLAN 6**

10110060 **VLANI4** **VLAN Identifier 4** **0000A009**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID9[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID9[3:0]				VID8											
Type	RW				RW											
Reset	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Name	Description
23:12	VID9	**VLAN Field Identifier for VLAN 9**
11:0	VID8	**VLAN Field Identifier for VLAN 8**

10110064 **VLANI5** **VLAN Identifier 5** **0000C00B**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID11[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID11[3:0]				VID10											
Type	RW				RW											

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

Bit(s)	Name	Description
23:12	VID11	**VLAN Field Identifier for VLAN 11**
11:0	VID10	**VLAN Field Identifier for VLAN 10**

10110068 VLANI6 VLAN Identifier 6 0000E00D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID13[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID13[3:0]				VID12											
Type	RW				RW											
Reset	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1

Bit(s)	Name	Description
23:12	VID13	**VLAN Field Identifier for VLAN 13**
11:0	VID12	**VLAN Field Identifier for VLAN 12**

1011006C VLANI7 VLAN Identifier 7 0001000F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									VID15[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VID15[3:0]				VID14											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Bit(s)	Name	Description
23:12	VID15	**VLAN Field Identifier for VLAN 15**
11:0	VID14	**VLAN Field Identifier for VLAN 14**

10110070 VMSC0 VLAN Member Port Configuration 0 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VLAN_MEMSET_3								VLAN_MEMSET_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_MEMSET_1								VLAN_MEMSET_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	VLAN_MEMSET_3	**VLAN 3 Member Port**
23:16	VLAN_MEMSET_2	**VLAN 2 Member Port**
15:8	VLAN_MEMSET_1	**VLAN 1 Member Port**
7:0	VLAN_MEMSET_0	**VLAN 0 Member Port**

10110074 VMSC1 VLAN Member Port Configuration 1 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VLAN_MEMSET_7								VLAN_MEMSET_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_MEMSET_5								VLAN_MEMSET_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	VLAN_MEMSET_7	**VLAN 7 Member Port**
23:16	VLAN_MEMSET_6	**VLAN 6 Member Port**
15:8	VLAN_MEMSET_5	**VLAN 5 Member Port**
7:0	VLAN_MEMSET_4	**VLAN 4 Member Port**

10110078 VMSC2 VLAN Member Port Configuration 2 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VLAN_MEMSET_11								VLAN_MEMSET_10							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_MEMSET_9								VLAN_MEMSET_8							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	VLAN_MEMSET_11	**VLAN 11 Member Port**
23:16	VLAN_MEMSET_10	**VLAN 10 Member Port**
15:8	VLAN_MEMSET_9	**VLAN 9 Member Port**
7:0	VLAN_MEMSET_8	**VLAN 8 Member Port**

1011007C VMSC3 VLAN Member Port Configuration 3 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VLAN_MEMSET_15								VLAN_MEMSET_14							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_MEMSET_13								VLAN_MEMSET_12							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	VLAN_MEMSET_15	**VLAN 15 Member Port**
23:16	VLAN_MEMSET_14	**VLAN 14 Member Port**
15:8	VLAN_MEMSET_13	**VLAN 13 Member Port**
7:0	VLAN_MEMSET_12	**VLAN 12 Member Port**

10110080 **POA** **Port Ability Offset** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	G1_LINK	G0_LINK	LINK					G1_TXC		G0_TXC		XFC				
Type	RO	RO	RO					RO		RO		RO				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DUPLEX							G1_SPD		G0_SPD		SPEED				
Type	RO							RO		RO		RO				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	G1_LINK	**Port 6 Link Status** 1: Link up 0: Link down
30	G0_LINK	**Port 5 Link Status** [Note] This feature is only valid when port 5 giga MAC is implemented. 1: Link up 0: Link down
29:25	LINK	**Port 4 ~ port0 Link Status** 1: Link up 0: Link down
24:23	G1_TXC	**Flow Control Status fo Port6** The flow control capability status bit after Auto-negotiation or force mode. 1xb: full duplex and tx flow control ON x1b: full duplex and rx flow control ON 00b: flow control off
22:21	G0_TXC	**Flow Control Status fo Port5** The flow control capability status bit after Auto-negotiation or force mode. [Note] This feature is only valid when port 5 giga MAC is implemented. 1xb: full duplex and tx flow control ON x1b: full duplex and rx flow control ON 00b: flow control off
20:16	XFC	**Flow Control Status of port 0 ~ 4** The flow control capability status bit after Auto-negotiation or force mode. 0: flow control off 1: full duplex and 802.3x flow control ON (after AN or forced)
15:9	DUPLEX	**Port6 ~ port0 Duplex Mode**

Bit(s)	Name	Description
		[Note]: Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: half duplex 1: full duplex
8:7	G1_SPD	**MII port 6 Speed:Mode** 10: 1000M 01: 100M 00: 10M
6:5	G0_SPD	**MII port 5 Speed:Mode** [Note] This feature is only valid when port 5 Giga MAC is implemented 10: 1000M 01: 100M 00: 10M
4:0	SPEED	**Port4 ~ port0 Speed Mode** 0: 10M 1: 100M

10110084 **FPA** **Force Port4 - Port0 Ability** **0000000 0**

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

Bit(s)	Name	Description
31:27	FORCE_MODE	**Port4 ~ port 0 force mode** 0: default 1: force mode. Auto-negotiation status is ignored. All the port ability are forced according to the following fields of the register FPA.
26:22	FORCE_LINK	**Port 4 ~ port 0 PHY Link** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 1: Link up 0: Link down
20:16	FORCE_XFC	**Port 4 ~ port 0 Flow control of PHY port** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 0: default OFF 1: 802.3x flow control ON
12:8	FORCE_DPX	**Flow Control Status of port 0 ~ 4** The flow control capability status bit after Auto-negotiation or force mode. 0: flow control off 1: full duplex and 802.3x flow control ON (after AN or forced)
5	XTAL_COMP	**Crystal rate compensation** 0: Disable

Bit(s)	Name	Description
		1: When the switch has transmitted 20000 bytes, the switch will compensate for the loss of crystal rate.
4:0	FORCE_SPD	**Port4 ~ port0 Speed:** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 1: 100M 0: 10M

10110088 **PTS** **Port Status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name							G1_TXC_STATUS	G0_TXC_STATUS		SECURED_ST						
Type							RO	RO		RO						
Reset							0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
9	G1_TXC_STATUS	**Port 6 TXC status** 0: no alert 1: error, no TXC
8	G0_TXC_STATUS	**Port 5 TXC status** [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: no alert 1: error, no TXC
6:0	SECURED_ST	**Security Status** 0: no alert 1: has intruder coming if turn on the SA_secured mode, read clear

1011008C **SOCPC** **SoC Port Control** **027F7F7F**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							CRC_PADDING	CPU_SELECTION		DISBC2CPU						
Type							RW	RW		RW						
Reset							1	0	0	1	1	1	1	1	1	1
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name		DISMC2CPU								DISUN2CPU						
Type		RW								RW						
Reset		1	1	1	1	1	1	1		1	1	1	1	1	1	1

Bit(s)	Name	Description

Bit(s)	Name	Description
25	CRC_PADDING	**CRC padding from CPU** If this bit is set , all packets from CPU don't need to append CRC and the outgoing LAN/WAN port will calculate and append CRC. 0: packets from CPU need CRC appending 1: packets from CPU without CRC appending
24:23	CPU_SELECTION	**CPU Selection** 00b: Port 6 01b: Port 0 10b: Port 4 11b: Port 5
22:16	DISBC2CPU	**Disable BC to CPU** When this bit = 1, BC frames from the corresponding port will not be forward to CPU. [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: Includes CPU port. 1: Excludes CPU port
14:8	DISMC2CPU	**Disable MC to CPU** When this bit =1, MC frames from the corresponding port will not forward to CPU. [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: Includes CPU port. 1: Excludes CPU port
6:0	DISUN2CPU	**Disable UN to CPU** When this bit =1, Unkonwn frames from the corresponding port will not forward to CPU. [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: Includes CPU port. 1: Excludes CPU port

10110090 **POC0** **Port Control 0** **3F807F7F**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	HASH_ADDR_SHIFT		DIS_GMII_PORT_1	DIS_GMII_PORT_0	DIS_PORT					DISRMC2_CPU						
Type	RW		RW	RW	RW					RW						
Reset	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		EN_FC							MAC_FCP_OPTION	EN_BP						
Type		RW							RW	RW						
Reset		1	1	1	1	1	1	1	0	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:30	HASH_ADDR_SHIFT	**Address table hashing algorithm option for member set index**

Bit(s)	Name	Description
29	DIS_GMII_PORT_1	**Disable port 6** 0: port enable 1: port disable
28	DIS_GMII_PORT_0	**Disable port 5** [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: port enable 1: port disable
27:23	DIS_PORT	**Disable phy port** 0: port enable 1: port disable
22:16	DISRMC2_CPU	**Unknown Reserved Multicast Frame Excludes CPU** [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: Unknown Reserved Multicast Forward Rule (SGC.RMC_RULE) 1: Excludes CPU port
14:8	EN_FC	**Apply 802.3x status after Auto-negotiation** This field can individually control the 802.3x capability after Auto-negotiation is done. [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: ignore the AN stats for 802.3x capability 1: follow the AN status for 802.3x capability
7	MAC_FCP_OPTION	**Multicast Flow control/Backpressure option** 0: When all ports are fc/bp disable, the switch will use drop_threshold to drop frames only. If not, the switch will use fc_threshold and drop_threshold. 1: When only the destination TX port is fc/bp disable, the switch will use drop_threshold to drop frames only . If not, that TX port uses fc_threshold and drop_threshold.
6:0	EN_BP	**Apply back pressure capability** [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: ignore the back pressure mode (default OFF) 1: apply back pressure based on SGC.BP_MODE.

10110094 **POC1** **Port Control 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DISIPMC2CPU							BLOCKING_STATE						
Type			RW							RW						
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DIS_LRNING								SA_SECURED_PORT						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
29:23	DISIPMC2CPU	**Unknown IP Multicast Frame Excludes CPU** 0: Unknown IP Multicast Forward Rule (SGC.IP_MULT_RULE) 1: Excludes CPU port
22:16	BLOCKING_STATE	**Port State for Spanning Tree Protocol** [Note]: Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: normal state 1: blocking state, forwarding rmc packet to cpu(need programming address table

Bit(s)	Name	Description
14:8	DIS_LRNING	**Disable SA learning** [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: default enabled 1: disable Source MAC learning
6:0	SA_SECURED_PORT	**SA secured mode** [Note*1]: Must set dis_learn and sa_secured at the same time. [Note*2] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: don't care SA match, 1: the packets' SA needs match, otherwise discard the packets

10110098 POC2 Port Control 2 00007F00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		G1_TXC_CHECK	G0_TXC_CHECK				MLD2CPU_EN	IPV6_MULT_RULE		DIS_UC_PAUSE						
Type		RW	RW				RW	RW		RW						
Reset		0	0				0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PER_VLAN_UNTAG_EN	ENAGING_PORT								UNTAG_EN						
Type	RW	RW								RW						
Reset	0	1	1	1	1	1	1	1		0	0	0	0	0	0	0

Bit(s)	Name	Description
30	G1_TXC_CHECK	**Check the port 6 TXC** if no txc clock, then disable MII port 0: disable 1: enable, check TXC
29	G0_TXC_CHECK	**Check the port 5 TXC** if no txc clock, then disable MII port [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: disable 1: enable, check TXC
25	MLD2CPU_EN	**MLD Message Packets forward to CPU** 0: MLD message will be flooded 1: MLD message will be forward to CPU port only
24:23	IPV6_MULT_RULE	**Unknown IPV6 Multicast Frame Forward Rule** If no match in the address table, then folloing the rule 00: BC 01: to CPU 10: drop 11: Reserved
22:16	DIS_UC_PAUSE	**Disable Unicast Pause Frame**

Bit(s)	Name	Description
		[Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: switch will consider pause frame when DA!=0180c20001 but unicast to CPU, 1: switch will not consider pause frame when DA!= 0180c20001 and unicast to CPU
15	PER_VLAN_UNTAG_EN	**Per port per vlan untag enable** VLAN tag removal option. 0: Use per port UNTAG_EN 1: Use untag enable bitmap in VLAN table
14:8	ENAGING_PORT	**Port aging** [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: disable aging that the MAC address is belong to programmed port(s) 1: enable aging
6:0	UNTAG_EN	**Per Port VLAN Tag Temoval** [Note] Port5 funciton is only valid when port 5 Giga MAC is implemented. 0: disable 1: enable VLAN tag field removal.

1011009C **SGC** **Switch Global Control** **6008A041**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		BKOFF_ALG	LEN_ERR_CHK	IP_MULT_RULE		RMC_RULE		LED_FLASH_TIME		BISH_TH		BISH_DIS	BP_MODE		DISMIIPORT_WASTX	
Type		RW	RW	RW		RW		RW		RW		RW	RW		RW	
Reset		1	1	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BP_JAM_CNT				DISABLE_TX_BACKOFF	ADDRESS_HASH_ALG		DIS_PKT_TX_ABORT	PKT_MAX_LEN		BC_STORM_PROT		AGING_INTERNAL			
Type	RW				RW	RW		RW	RW		RW		RW			
Reset	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1

Bit(s)	Name	Description
30	BKOFF_ALG	**Backoff Algorithm Option** 0: default 1: comply with UNH test
29	LEN_ERR_CHK	**Length of Received Frame Check Enable** When the bit is set, the received packet length will be checked for length encapsulated frames. 0: default disabled 1: comply with UNH test
28:27	IP_MULT_RULE	**Unknown IP Multicase Frame Forward Rule** If no match in the address table, then following the rules. 00: BC 01: to cpu

Bit(s)	Name	Description
		10: drop 11: reserved
26:25	RMC_RULE	**Unknown Reserved Multicast Frame Forward Rule** If no match in the address table, then follow the rules. 00: to all port(not include blocking state port) 01: to cpu 10: drop 11: reserved
24:23	LED_FLASH_TIME	**The Frequency Of LED Flash** 00: 30ms 01: 60ms 10: 240ms 11: 480ms
22:21	BISH_TH	**The Threshold Of Memory Bisshop** 11: skip if fail 8 blocks, 0 00: skip if fail 16 (default, from pins) 01: skip if fail 48 10: skip if fail 64
20	BISH_DIS	**Build In Self Hop** 0: enable skip function 1: disable
19:18	BP_MODE	**Back Pressure Mode** 00: disable 01: BP jam, the jam number is set by bp_num 10: BP jamALL, jam packet until the BP condition is released(default), 11: BP carrier, use carrier insertion to do back pressure
17:16	DISMIIPORT_WASTX	**GMII Port Disable Was_Transmit** [Note] This feature is only valid when port 5 Giga MAC is implemented. 1: disable was_transmit (good for late CRS PHY, like HPNA2.0 or power-LAN), 0: enable
15:12	BP_JAM_CNT	**Back Pressure Jam Number** The consecutive jam count when back pressure is enabled, The default is 10 packet jam then one no-jam packet.
11	DISABLE_TX_BACKOFF	**Disable The Collision Back Off Timer** 0: default 1: re-transmit immediately after collision
10:9	ADDRESSS_HASH_ALG	**MAC Address Hashing Algorithm** 00: direct mode, using last 10-bit as hashing address 01: XOR48 mode 10: XOR32 mode 11: reserved
8	DIS_PKT_TX_ABORT	**Disable Packet TX Abort** 1: Disable collision 16 packet abort and late collision abort 0: enable both abort
7:6	PKT_MAX_LEN	**Maximum Packet Length** Untaged / VLAN-taged 00: 1536 Bytes / 1536 Bytes 01: 1518 Bytes / 1522 Bytes 10: 1522 Bytes / 1526 Bytes 11: Reserved / Reserved
5:4	BC_STORM_PROT	**Global Broadcast Storm Protection**

Bit(s)	Name	Description
		BC will be blocked, if the following number of BC blocks in in output queues 00: disable 01: 64 10: 96 11: 128
3:0	AGING_INTERNAL	**Aging Timer** 0000: disable age 0001: 300sec 0010 - 0111: 600 ~ 38400sec 1xxx: Fast Age (60sec)

101100A0 **STRT** **Switch Reset** **0000000**
0

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

Bit(s)	Name	Description
31:0	RESET_SW	**Reset switch engine, data, address, link memory , cpu port and ahb interface when writing data to the STRT register.**

101100A4 **LEDP0** **LED Port0** **0000000**
5

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													P0_LED			
Type													RW			
Reset													0	1	0	1

Bit(s)	Name	Description
3:0	P0_LED	**port0 LED state, default = link/activity** 0000: link 0001: 100M speed 0010: duplex 0011: activity 0100: collision 0101: link/activity 0110: duplex/collsion 0111: 10M speed/activity 1000: 100M speed/activity 1011: off 1100: on 1010: blink

101100A8 **LEDP1** **LED Port1** **00000005**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													P1_LED			
Type													RW			
Reset													0	1	0	1

Bit(s)	Name	Description
3:0	P1_LED	port1 LED state, default = link/activity

101100AC **LEDP2** **LED Port2** **00000005**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													P2_LED			
Type													RW			
Reset													0	1	0	1

Bit(s)	Name	Description
3:0	P2_LED	port2 LED state, default = link/activity

101100B0 **LEDP3** **LED Port3** **00000005**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													P3_LED			
Type													RW			
Reset													0	1	0	1

Bit(s)	Name	Description
3:0	P3_LED	port3 LED state, default = link/activity

101100B4 **LEDP4** **LED Port4** **00000005**

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

Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													P4_LED			
Type													RW			
Reset													0	1	0	1

Bit(s)	Name	Description
3:0	P4_LED	**port4 LED state, default = link/activity**

101100B8 WDTR Watch Dog Trigger Reset 0000001E

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

Bit(s)	Name	Description
7:0	BUF_STARV_TH	**Buffer starvation threshold** Switch will interrupt CPU when the global queue block counts is less than the threshold for 3 seconds.

101100BC DES Debug Signal 00000000

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

Bit(s)	Name	Description
15:0	DEBUG_SIGNAL	**Port 5 Debug Signal**

101100C0 PCR0 PHY Control Register 0 00000000

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

	SV0	_PHY_CMD	_PHY_CMD													
Type	RO	RW	RW	RW								RW				
Reset	0	0	0	0	0	0	0	0				0	0	0	0	0

Bit(s)	Name	Description
31:16	WT_NWAY_DATA	**The Data Be Written into PHY**
15	RESV0	**Reserved**
14	RD_PHY_CMD	**Read command** To enable read command on PHY, write 1 to this bit . After command is completed, this bit is self-cleared.
13	WT_PHY_CMD	**Write command** To enable write command on PHY, write 1 to this bit . After command is completed, this bit is self-cleared
12:8	CPU_PHY_REG	**PHY register address**
4:0	CPU_PHY_ADDR	**PHY address** (Note: The internal 5-ports PHY reserves the PHY address starting from 5'd0 ~ 5'd4. For the external PHY, the PHY address from 5'd5 to 5'd31 can be applied. The default PHY address of Port 5 is 5'd5 for auto-polling function.)

101100C4 **PCR1** **PHY Control Register 1** **00000000**

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

Bit(s)	Name	Description
31:16	RD_DATA	**The Read Data**
1	RD_RDY	**Read Operation is Done**
0	WT_DONE	**Write Operation is Done**

101100C8 **FPA1** **Force P5P6 Ability** **05500328**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			AP_EN	EXT_PHY_ADDR_BASE					G0_RXCLK_SKEW_SEL		G0_TXCLK_SKEW_SEL			TURBO_MII_CLK		
Type			RW	RW					RW		RW			RW		
Reset			0	0	0	1	0	1	0	1	0	1		0		

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			FORCE_RGMII_LINK1	FORCE_RGMII_LINK0	FORCE_RGMII_EN1	FORCE_RGMII_EN0	FORCE_RGMII_XFC1		FORCE_RGMII_XFC0		FORCE_RGMII_DPX1	FORCE_RGMII_DPX0	FORCE_RGMII_SPD1		FORCE_RGMII_SPD0	
Type			RW	RW	RW	RW	RW		RW		RW	RW	RW		RW	
Reset			0	0	0	0	1	1	0	0	1	0	1	0	0	0

Bit(s)	Name	Description
29	AP_EN	**Port 5 Auto Polling Enable** [Note] This feature is only valid when port 5 Giga MAC is implemented.
28:24	EXT_PHY_ADDR_BASE	**Port 5 External PHY Base Address** [Note] This feature is only valid when port 5 Giga MAC is implemented.
23:22	G0_RXCLK_SKEW_SEL	**Port 5 RXCLK Skew Selection** [Note] This feature is only valid when port 5 Giga MAC is implemented.
21:20	G0_TXCLK_SKEW_SEL	**Port 5 TXCLK Skew Selection** [Note] This feature is only valid when port 5 Giga MAC is implemented.
18	TURBO_MII_CLK	**Port 5 revMII Mode Clock Selection** [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: 25MHz output clock 1: 31.25MHz output clock
13	FORCE_RGMII_LINK1	**Force Port 6 Link** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 0: link down 1: link up
12	FORCE_RGMII_LINK0	**Force Port 5 Link** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: link down 1: link up
11	FORCE_RGMII_EN1	**Force Port 6 Enable** 0: reserved 1: force mode. Auto-negotiation status is ignored. Port 5 ability is forced according to the following fields of the register FPA1.
10	FORCE_RGMII_EN0	**Force Port 5 Enable** [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: default 1: force mode. Auto-negotiation status is ignored. Port 5 ability is forced according to the following fields of the register FPA1.
9:8	FORCE_RGMII_XFC1	**Force port 6 flow control ability** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 1x: for tx x1: for rx
7:6	FORCE_RGMII_XFC0	**Force port 5 flow control ability** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. [Note] This feature is only valid when port 5 Giga MAC is implemented. 1x: for tx

Bit(s)	Name	Description
		x1: for rx
5	FORCE_RGMII_DPX1	**Force port 6 duplex** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 0: half duplex 1: full duplex
4	FORCE_RGMII_DPX0	**Force port 5 duplex** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. [Note] This feature is only valid when port 5 Giga MAC is implemented. 0: half duplex 1: full duplex
3:2	FORCE_RGMII_SPD1	**Force port 6 speed** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. 1x: 1GbpsMhz 01: 100MbpsMHz 00: 10MbpsMHz
1:0	FORCE_RGMII_SPD0	**Force port 5 speed** This field is valid only FORCE_MDOE is set. The final resolution is reported to POA register. [Note] This feature is only valid when port 5 Giga MAC is implemented. 1x: 1GbpsMhz 01: 100MbpsMHz 00: 10MbpsMHz

101100CC **FCT2** **Flow Control Threshold 2** **0000A30C**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								DIS_IPV6MC2CPU							MUST_DROP_RLS_TH[4:3]	
Type								RW							RW	
Reset								0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MUST_DROP_RLS_TH[2:0]			MUST_DROP_SET_TH							MC_PER_PORT_TH					
Type	RW			RW							RW					
Reset	1	0	1	0	0	0	1	1			0	0	1	1	0	0

Bit(s)	Name	Description
24:18	DIS_IPV6MC2CPU	**Unknown IPv6 Multicast Frame Excludes CPU** 0: Unknown IPv6 Multicast Forward Rule (POC2.IPV6_MULT_RULE) 1: Exclude CPU port
17:13	MUST_DROP_RLS_TH	**If the global queue pointer higher than the threshold. The must drop condition will be released.**
12:8	MUST_DROP_SET_TH	**If the global queue pointer reach msut drop threshold. All incoming packets have to be dropped.**
5:0	MC_PER_PORT_TH	**MC packets per port threshold.** When the global queue reaches the flow control threshold on register FCT0, per port output threshold for MC packet will be checked to enable flow-control or packet-drop on imncoming MC packets.

101100D0 **QSS0** **Queue Status 0** **00000000**

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

Bit(s)	Name	Description
23:15	BE_CNT_R	**Link control best effort queue block counter monitor.**
14:5	BK_CNT_R	**Link control background queue block counter monitor.**
4:0	SEE_CNT_PORT_SEL	**Link control bock couonter port selection**

101100D4 **QSS1** **Queue Status 1** **00000000**

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

Bit(s)	Name	Description
17:9	VO_CNT_R	**Link control voice queue block counter monitor.**
8:0	CL_CNT_R	**Link control control queue block counter monitor.**

101100D8 **DEC** **Debug Control** **40400100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SW2FE_IPG								FE2SW_IPG							
Type	RW								RW							
Reset	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name								BRIDGE_EN								
Type								RW								

Reset								1								

Bit(s)	Name	Description
31:24	SW2FE_IPG	**SW2FE Bridge IPG Byte Count** Inter-Frame Byte Count between the consecutive frames flowing from Switch to Frame Engine
23:16	FE2SW_IPG	**FE2SW Bridge IPG byte count** Inter-Frame Byte Count between the consecurtive frames flowing from Frame Engine to Switch
8	BRIDGE_EN	**Enable FE2SW Bridge IPG Prevention** 1'b0: Disable 1'b1: Enable IPG Prevention when FE2SW_BRIDGE_IPG is too short (8'd16) to receive the next frame.

101100DC **MTI** **Memory Test Information** **0000006A**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name										SW_RAM_TEST_DONE	LK_RAM_TEST_DONE	LK_RAM_TEST_FAIL	AT_RAM_TEST_DONE	AT_RAM_TEST_FAIL	DT_RAM_TEST_DONE	DT_RAM_TEST_FAIL
Type										RO	RO	RO	RO	RO	RO	RO
Reset										1	1	0	1	0	1	0

Bit(s)	Name	Description
6	SW_RAM_TEST_DONE	**Switch Memory Ram Test Done**
5	LK_RAM_TEST_DONE	**Link Ram Test Done**
4	LK_RAM_TEST_FAIL	**Link Ram Test Fail**
3	AT_RAM_TEST_DONE	**Address Table Ram Test Done**
2	AT_RAM_TEST_FAIL	**Address Table Ram Test Fail**
1	DT_RAM_TEST_DONE	**Data Buffer Ram Test Done**
0	DT_RAM_TEST_FAIL	**Data Buffer Ram Test Fail**

101100E0 **PPC** **Packet Counter** **00000000**

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

Name	SW2FE_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FE2SW_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	SW2FE_CNT	**SW2FE_CNT Switch to frame engine packet counter**
15:0	FE2SW_CNT	**FE2SW_CNT Frame engine to switch packet counter**

101100E4 **SGC2** **Switch Global Control 2** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P6_RXFC_QUE_EN	P6_TXFC_WL_EN	LAN_PMAP						SPECIAL_TAG_EN	TX_CPU_TPID_BIT_MAP						
Type	RW	RW	RW						RW	RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				P6_TXFC_QUE_EN	ARBITER_LAN_EN	CPU_TPID_EN	ARBITER_GPT_EN	SLOT_4TO1		DOUBLE_TAG_EN						
Type				RW	RW	RW	RW	RW		RW						
Reset				0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
31	P6_RXFC_QUE_EN	**Port 6 RX flow control on per egress queue** 0: Port 6 RX flow control will pause all 4 egress queue 1: Port 6 RX flow control will pause 4 egress queue independently according to the corresponding congestion signals.
30	P6_TXFC_WL_EN	**Port 6 TX flow controll by Switch WAN/LAN port** 0: Port 6 TX flow control is decided by any port and any queue of the Switch congestion 1: Port 6 TX flow control is decided by WAN/LAN port of the Switch congestion separately.
29:24	LAN_PMAP	**Lan port bit map** This field indicates per port attribute used for flow control. (Note: Port5 funciton is only valid when port 5 Giga MAC is implemented) 1: Lan port 0: Wan port
23	SPECIAL_TAG_EN	**Special Tag enable** 0: default; RX special tag is enabled according to the global control bit-CPU_TPID_EN. TX special tag is enabled according to the per-port TX_CPU_TPID_BIT_MAP 1: CPU_TPID_EN is not used Both TX and RX special tag feature are decided by the per-port TX_CPU_TPID_BIT_MAP

Bit(s)	Name	Description
22:16	TX_CPU_TPID_BIT_MAP	**Transmit CPU TPID(810x) port bit map** 0: default (TPID=0x8100) 1: TPID=0x810? depending on TX/RX usages (Note: Port5 funciton is only valid when port 5 Giga MAC is implemented)
12	P6_TXFC_QUE_EN	**Port 6 per queue TX flow control** This bit is only valid when P6_TXFC_WL_EN is enabled. 0: 4 congest signals to Frame Engine are decided by the wired-or result of all egress queues on Switch WAN/LAN ports. 1: 4 congest signals to Frame Engine are decided by the individual and the corresponding 4 egress queues on Switch WAN/LAN ports.
11	ARBITER_LAN_EN	**Memory arbiter only for P0~P4 enable** 0: default 1: memory arbiter only for P0~P4.
10	CPU_TPID_EN	**CPU TPID(81xx) enable** 0: disable. CPU TPID=8100 1: enable. CPU TPID=810x.
9	ARBITER_GPT_EN	**Memory Arbiter only for P5 and P6** 0: default 1: Enable
8	SLOT_4TO1	**Memory Arbiter Ratio Selection** 0: (P5,P6) : (P0-P4) = 3:2 1: (P5,P6) : (P0-P4) = 4:1
6:0	DOUBLE_TAG_EN	**Insert double tag field** When this bit is set , the incoming packet is allowed to insert outer or double tag. 1: enable double tag field 0: disable the double tag field. (Note: Port5 funciton is only valid when port 5 Giga MAC is implemented)

101100E8 **P0PC** **Port 0 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT0	**Port 0 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT0	**Port 0 Receive Good Packet Counter**

101100EC **P1PC** **Port 1 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT1															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT1															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT1	**Port 1 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT1	**Port 1 Receive Good Packet Counter**

101100F0 **P2PC** **Port 2 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT2															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT2															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT2	**Port 2 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT2	**Port 2 Receive Good Packet Counter**

101100F4 **P3PC** **Port 3 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT3															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT3															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT3	**Port 3 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT3	**Port 3 Receive Good Packet Counter**

101100F8 **P4PC** **Port 4 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT4															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	GOOD_PKT_CNT4															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT4	**Port 4 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT4	**Port 4 Receive Good Packet Counter**

101100FC **P5PC** **Port 5 Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT5															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT5															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT5	**Port 5 Receive Bad Packet Counter**
15:0	GOOD_PKT_CNT5	**Port 5 Receive Good Packet Counter**

10110100 **VUB0** **VLAN Untag Block 0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					VLAN_3_UNTAG_EN							VLAN_2_UNTAG_EN[6:2]				
Type					RW							RW				
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_2_UNTAG_EN[1:0]		VLAN_1_UNTAG_EN							VLAN_0_UNTAG_EN						
Type	RW		RW							RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:21	VLAN_3_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 3**
20:14	VLAN_2_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 2**
13:7	VLAN_1_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 1**
6:0	VLAN_0_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 0**

10110104 **VUB1** **VLAN Untag Block 1** **0000000**

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					VLAN_7_UNTAG_EN							VLAN_6_UNTAG_EN[6:2]				
Type					RW							RW				
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_6_UNTAG_EN[1:0]		VLAN_5_UNTAG_EN							VLAN_4_UNTAG_EN						
Type	RW		RW							RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:21	VLAN_7_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 7**
20:14	VLAN_6_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 6**
13:7	VLAN_5_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 5**
6:0	VLAN_4_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 4**

10110108 VUB2 VLAN Untag Block 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					VLAN_11_UNTAG_EN							VLAN_10_UNTAG_EN[6:2]				
Type					RW							RW				
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_10_UNTAG_EN[1:0]		VLAN_9_UNTAG_EN							VLAN_8_UNTAG_EN						
Type	RW		RW							RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:21	VLAN_11_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 11**
20:14	VLAN_10_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 10**
13:7	VLAN_9_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 9**
6:0	VLAN_8_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 8**

1011010C VUB3 VLAN Untag Block 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					VLAN_15_UNTAG_EN							VLAN_14_UNTAG_EN[6:2]				
Type					RW							RW				
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VLAN_14_UNTAG_EN[1:0]		VLAN_13_UNTAG_EN							VLAN_12_UNTAG_EN						
Type	RW		RW							RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:21	VLAN_15_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 15**
20:14	VLAN_14_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 14**
13:7	VLAN_13_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 13**
6:0	VLAN_12_UNTAG_EN	**Port 0 ~ 6 untag_en of VLAN 12**

10110110 BMU_CTRL BC/MC/UN Rate Limit Control 7C000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		ONE_US_CYCLE_NUM								P5_RATE_LIMIT_CTRL				P4_RATE_LIMIT_CTRL		
Type		RW								RW				RW		
Reset		1	1	1	1	1	0	0		0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		P3_RATE_LIMIT_CTRL				P2_RATE_LIMIT_CTRL				P1_RATE_LIMIT_CTRL				P0_RATE_LIMIT_CTRL		
Type		RW				RW				RW				RW		
Reset		0	0	0		0	0	0		0	0	0		0	0	0

Bit(s)	Name	Description
30:24	ONE_US_CYCLE_NUM	**One micro-second Cycle Number** This field is used to calculate 1us period
22:20	P5_RATE_LIMIT_CTRL	**Port 5 rate Limit Control** (Note: This feature is only alid when port 5 GMAC is implemented)
18:16	P4_RATE_LIMIT_CTRL	**Port 4 rate Limit Control**
14:12	P3_RATE_LIMIT_CTRL	**Port 3 rate Limit Control**
10:8	P2_RATE_LIMIT_CTRL	**Port 2 rate Limit Control**
6:4	P1_RATE_LIMIT_CTRL	**Port 1 rate Limit Control**
2:0	P0_RATE_LIMIT_CTRL	**Port 0 rate Limit Control** 2: Broadcast frame enable 1: Multicast frame enable 0: Unknown frame enable

10110114 BMU_LMT_N BC/MC/UN Rate Limit Frame Number FFFFFF

UM1 **FF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RATE_LIMIT_NUM_100M															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RATE_LIMIT_NUM_10M															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:16	RATE_LIMIT_NUM_100M	**Rate Limit Received BC/MC/UN frame number in 100M in 100ms duration**
15:0	RATE_LIMIT_NUM_10M	**Rate Limit Received BC/MC/UN frame number in 10M in 1s duration**

10110118 **BMU_LMT_NUM2** **BC/MC/UN Rate Limit Frame Number** **1818FFFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IG_RATE_BYTE_OPT	IG_RATE_BYTE_NUM							EG_RATE_BYTE_OPT	EG_RATE_BYTE_NUM						
Type	RW	RW							RW	RW						
Reset	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RATE_LIMIT_NUM_1000M															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31	IG_RATE_BYTE_OPT	**Ingress Rate Byte Option** 0: Add 1: Minus
30:24	IG_RATE_BYTE_NUM	**Ingress Rate Byte Number**
23	EG_RATE_BYTE_OPT	**Egress Rate Byte Option** 0: Add 1: Minus
22:16	EG_RATE_BYTE_NUM	**Egress Rate Byte Number**
15:0	RATE_LIMIT_NUM_1000M	**Rate Limit Received BC/MC/UN frame number in 1000M in 10ms duration** (note: This feature is only valid whe port 5 GMAC is implemented))

1011011C **P01_ING_CTRL** **Port 0&1 Ingress Rate Limit Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		P1_INGRESS_CTRL	P1_MNG_PKT_BYPASS	P1_INGRESS_FLOW_CTRL_ON	P1_TIMER_TICK		P1_TOKEN									
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		P0_INGRESS_CTRL	P0_MNG_PKT_BYPASS	P0_INGRESS_FLOW_CTRL_ON	P0_TIMER_TICK		P0_TOKEN									
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30	P1_INGRESS_CTRL	**Port1 Ingress Limit Control** 0: OFF 1: ON
29	P1_MNG_PKT_BYPASS	**Port1 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
28	P1_INGRESS_FLOW_CTRL_ON	**Port 1 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P1_ING_THRES. If the bucket is empty, then P1 will start to discard the received packets except those specific packet in P1_MNG_PKY_BYPASS mode. 0: OFF 1: ON
27:26	P1_TIMER_TICK	**Port 1 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
25:16	P1_TOKEN	**Port 1 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes
14	P0_INGRESS_CTRL	**Port 0 Ingress Limit Control** 0: OFF 1: ON
13	P0_MNG_PKT_BYPASS	**Port 0 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
12	P0_INGRESS_FLOW_CTRL_ON	**Port 0 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P0_ING_THRES. If the bucket is empty, then P0 will start to

Bit(s)	Name	Description
		discard the received packets except those specific packet in P0_MNG_PKY_BYPASS mode. 0: OFF 1: ON
11:10	P0_TIMER_TICK	**Port 0 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
9:0	P0_TOKEN	**Port 0 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

10110120 **P23_ING_CTRL** **Port 2&3 Ingress Rate Limit Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		P3_INGRESS_CTRL	P3_MNG_PKT_BYPASS	P3_INGRESS_FLOW_CTRL_ON	P3_TIMER_TICK		P3_TOKEN									
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		P2_INGRESS_CTRL	P2_MNG_PKT_BYPASS	P2_INGRESS_FLOW_CTRL_ON	P2_TIMER_TICK		P2_TOKEN									
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30	P3_INGRESS_CTRL	**Port 3 Ingress Limit Control** 0: OFF 1: ON
29	P3_MNG_PKT_BYPASS	**Port 3 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
28	P3_INGRESS_FLOW_CTRL_ON	**Port 3 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P3_ING_THRES. If the bucket is empty, then P3 will start to discard the received packets except those specific packet in P3 MNG_PKY_BYPASS mode. 0: OFF

Bit(s)	Name	Description
		1: ON
27:26	P3_TIMER_TICK	**Port 3 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
25:16	P3_TOKEN	**Port 3 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes
14	P2_INGRESS_CTRL	**Port 2 Ingress Limit Control** 0: OFF 1: ON
13	P2_MNG_PKT_BYPASS	**Port 2 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
12	P2_INGRESS_FLOW_CTRL_ON	**Port 2 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P2_ING_THRES. If the bucket is empty, then P2 will start to discard the received packets except those specific packet in P2_MNG_PKY_BYPASS mode. 0: OFF 1: ON
11:10	P2_TIMER_TICK	**Port 2 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
9:0	P2_TOKEN	**Port 2 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

10110124 **P45_ING_CTRL** **Port 4&5 Ingress Rate Limit Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		P5_INGRESS_CTRL	P5_MNG_PKT_BYPASS	P5_INGRESS_FLOW_CTRL_ON	P5_TIMER_TICK		P5_TOKEN									
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		P4_INGRESS_CT	P4_MNG_PKT_B	P4_INGRESS_FL	P4_TIMER_TICK		P4_TOKEN									

		RL	YPASS	OW_CTRL_ON												
Type		RW	RW	RW	RW		RW									
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30	P5_INGRESS_CTRL	**Port 5 Ingress Limit Control** 0: OFF 1: ON
29	P5_MNG_PKT_BYPASS	**Port 5 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
28	P5_INGRESS_FLOW_CTRL_ON	**Port 5 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P5_ING_THRES. If the bucket is empty, then P5 will start to discard the received packets except those specific packet in P5_MNG_PKY_BYPASS mode. 0: OFF 1: ON
27:26	P5_TIMER_TICK	**Port 5 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
25:16	P5_TOKEN	**Port 5 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes
14	P4_INGRESS_CTRL	**Port 4 Ingress Limit Control** 0: OFF 1: ON
13	P4_MNG_PKT_BYPASS	**Port 4 Management Packet ByPass** 0: All packet included 1: Mangement Frame Excluded
12	P4_INGRESS_FLOW_CTRL_ON	**Port 4 Ingress rate Flow Control** When the bit is set, the pause frame is used prior to packet dropped according to P4_ING_THRES. If the bucket is empty, then P4 will start to discard the received packets except those specific packet in P4_MNG_PKY_BYPASS mode. 0: OFF 1: ON
11:10	P4_TIMER_TICK	**Port 4 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
9:0	P4_TOKEN	**Port 4 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

10110128 **P0_ING_THRES** **Port 0 Ingress Rate Limit Threshold** **AAAA5555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P0_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P0_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P0_IN_FCOFF_THRES	**Port 0 ingress rate limit flow control off.** If P0_INGRESS_FLOW_CTRL_ON = 1 and P0 Flow control capability is on (XFC status in 0x80), then P0 will initiate PAUSE OFF frame or stop backpressure.
15:0	P0_IN_FCON_THRES	**Port 0 ingress rate limit flow control on.** If P0_INGRESS_FLOW_CTRL_ON = 1 and P0 Flow control capability is on (XFC status in 0x80), then P0 will initiate PAUSE ON frame or backpressure.

1011012C **P1_ING_THRES** **Port 1 Ingress Rate Limit Threshold** **AAAA5555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P1_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P1_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P1_IN_FCOFF_THRES	**Port 1 ingress rate limit flow control off.** If P1_INGRESS_FLOW_CTRL_ON = 1 and P1 Flow control capability is on (XFC status in 0x80), then P2 will initiate PAUSE OFF frame or stop backpressure.
15:0	P1_IN_FCON_THRES	**Port 1 ingress rate limit flow control on.** If P1_INGRESS_FLOW_CTRL_ON = 1 and P1 Flow control capability is on (XFC status in 0x80), then P1 will initiate PAUSE ON frame or backpressure.

10110130 **P2_ING_THRES** **Port 2 Ingress Rate Limit Threshold** **AAAA5555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P2_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P2_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P2_IN_FCOFF_THRES	**Port 2 ingress rate limit flow control off.** If P2_INGRESS_FLOW_CTRL_ON = 1 and P2 Flow control capability is on (XFC status in 0x80), then P2 will initiate PAUSE OFF frame or stop backpressure.
15:0	P2_IN_FCON_THRES	**Port 2 ingress rate limit flow control on.** If P2_INGRESS_FLOW_CTRL_ON = 1 and P2 Flow control capability is on (XFC status in 0x80), then P2 will initiate PAUSE ON frame or backpressure.

10110134 **P3_ING_THRES** **Port 3 Ingress Rate Limit Threshold** **AAAA5555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P3_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P3_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P3_IN_FCOFF_THRES	**Port 3 ingress rate limit flow control off.** If P3_INGRESS_FLOW_CTRL_ON = 1 and P3 Flow control capability is on (XFC status in 0x80), then P3 will initiate PAUSE OFF frame or stop backpressure.
15:0	P3_IN_FCON_THRES	**Port 3 ingress rate limit flow control on.** If P3_INGRESS_FLOW_CTRL_ON = 1 and P3 Flow control capability is on (XFC status in 0x80), then P3 will initiate PAUSE ON frame or backpressure.

10110138 **P4_ING_THRES** **Port 4 Ingress Rate Limit Threshold** **AAAA5555**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P4_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P4_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P4_IN_FCOFF_THRES	**Port 4 ingress rate limit flow control off.** If P4_INGRESS_FLOW_CTRL_ON = 1 and P4 Flow control capability is on (XFC status in 0x80), then P4 will initiate PAUSE OFF frame or stop backpressure.
15:0	P4_IN_FCON_THRES	**Port 4 ingress rate limit flow control on.** If P4_INGRESS_FLOW_CTRL_ON = 1 and P4 Flow control capability is on (XFC status in 0x80), then P4 will initiate PAUSE ON frame or backpressure.

1011013C	P5_ING_THRES	Port 5 Ingress Rate Limit Threshold	AAAA5555

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	P5_IN_FCOFF_THRES															
Type	RW															
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P5_IN_FCON_THRES															
Type	RW															
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Name	Description
31:16	P5_IN_FCOFF_THRES	**Port 5 ingress rate limit flow control off.** If P5_INGRESS_FLOW_CTRL_ON = 1 and P5 Flow control capability is on (XFC status in 0x80), then P5 will initiate PAUSE OFF frame or stop backpressure. (note: This feature is only valid when port 5 Giga MAC is implemented)
15:0	P5_IN_FCON_THRES	**Port 5 ingress rate limit flow control on.** If P5_INGRESS_FLOW_CTRL_ON = 1 and P0 Flow control capability is on (XFC status in 0x80), then P5 will initiate PAUSE ON frame or backpressure. (note: This feature is only valid when port 5 Giga MAC is implemented)

10110140	P01_EG_CTRL	Port 0/1 Egress Rate Limit Control	00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				P1_EGRESS_CTRL	P1_TIMER_TICK		P1_TOKEN									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				P0_EGRESS_CTRL	P0_TIMER_TICK		P0_TOKEN									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	P1_EGRESS_CTRL	**Port 1 Egress Control** 1: ON 0: OFF
27:26	P1_TIMER_TICK	**Port 1 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us

Bit(s)	Name	Description
25:16	P1_TOKEN	**Port 1 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes 1: ON 0: OFF
12	P0_EGRESS_CTRL	**Port 0 Egress Control** 0: 512us 1: 128us 2: 32us 3: 8us
11:10	P0_TIMER_TICK	**Port 0 Timer Tick** 1: ON 0: OFF
9:0	P0_TOKEN	**Port 0 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

10110144 **P23_EG_CTRL** **Port 2/3 Egress Rate Limit Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				P3_EGRESS_CTRL	P3_TIMER_TICK		P3_TOKEN									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				P2_EGRESS_CTRL	P2_TIMER_TICK		P2_TOKEN									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	P3_EGRESS_CTRL	**Port 3 Egress Control** 1: ON 0: OFF
27:26	P3_TIMER_TICK	**Port 3 Timer Tick** 0: 512us 1: 128us 2: 32us 3: 8us
25:16	P3_TOKEN	**Port 3 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

Bit(s)	Name	Description
		1: ON 0: OFF
12	P2_EGRESS_CTRL	**Port 2 Egress Control** 0: 512us 1: 128us 2: 32us 3: 8us
11:10	P2_TIMER_TICK	**Port 2 Timer Tick** 1: ON 0: OFF
9:0	P2_TOKEN	**Port 2 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

10110148 **P45_EG_CTRL** **Port 4/5 Egress Rate Limit Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				**P5_EGRESS_CTRL**	**P5_TIMER_TICK**		**P5_TOKEN**									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				**P4_EGRESS_CTRL**	**P4_TIMER_TICK**		**P4_TOKEN**									
Type				RW	RW		RW									
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	P5_EGRESS_CTRL	**Port 5 Egress Control** (Note: This feature is only valid when port 5 Giga MAC is implemented) 1: ON 0: OFF
27:26	P5_TIMER_TICK	**Port 5 Timer Tick** (Note: This feature is only valid when port 5 Giga MAC is implemented) 0: 512us 1: 128us 2: 32us 3: 8us
25:16	P5_TOKEN	**Port 5 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes (Note: This feature is only valid when port 5 Giga MAC is implemented) 1: ON

Bit(s)	Name	Description
		0: OFF
12	P4_EGRESS_CTRL	**Port 4 Egress Control** 0: 512us 1: 128us 2: 32us 3: 8us
11:10	P4_TIMER_TICK	**Port 4 Timer Tick** 1: ON 0: OFF
9:0	P4_TOKEN	**Port 4 Token** Every timer tick, Token number bytes will be added into the bucket. (Unit : Byte) The maximum space of this bucket is 16'hFFFF bytes

1011014C **PCRI** **Packet Counter Recycle Indication** **00000000**

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

Bit(s)	Name	Description
31	PKT_CNT_CLR	**Tx/Rx Packet Counters Write One Clear** When this bit is set,all Tx/Rx packet counters will be clear.This bit can be self-clear automatically.
29:24	TCOL_PKT_REC	**Per Port Transmitted Collision Packet Counter Recycle** This bit indicates that the per port transmitted collision packet counter recycles the count. Write one clear.
22:16	TXOK_PKT_REC	**Per Port Transmitted Good Packet Counter Recycle** This bit indicates that the per port transmitted good packet counter recycles the count. Write one clear.
13:8	BAD_PKT_REC	**Per Port Received Bad Packet Counter Recycle** This bit indicates that the per port received badd packet counter recycles the count. Write one clear.
6:0	GOOD_PKT_REC	**Per Port Received Good Packet Counter Recycle** This bit indicates that the per port received good packet counter recycles the count. Write one clear.

10110150 **P0TPC** **Port 0 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT0	**Port 0 packet counte for transmitted packets with collisionautomatically.**
15:0	GOOD_PKT_CNT0	**Port 0 packet counter for transmitted packets successfully**

10110154 **P1TPC** **Port 1 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT1															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT1															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT1	**Port 1 packet counte for transmitted packets with collisionautomatically.**
15:0	GOOD_PKT_CNT1	**Port 1 packet counter for transmitted packets successfully**

10110158 **P2TPC** **Port 2 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT2															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT2															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT2	**Port 2 packet counte for transmitted packets with collisionautomatically.**
15:0	GOOD_PKT_CNT2	**Port 2 packet counter for transmitted packets successfully**

1011015C **P3TPC** **Port 3 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT3															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT3															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT3	**Port 3 packet counte for transmitted packets with collisionautomatically.**
15:0	GOOD_PKT_CNT3	**Port 3 packet counter for transmitted packets successfully**

10110160 **P4TPC** **Port 4 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT4															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT4															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT4	**Port 4 packet counte for transmitted packets with collisionautomatically.**
15:0	GOOD_PKT_CNT4	**Port 4 packet counter for transmitted packets successfully**

10110164 **P5TPC** **Port 5 TX Packet Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAD_PKT_CNT5															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOOD_PKT_CNT5															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	BAD_PKT_CNT5	**Port 5 packet counte for transmitted packets with collisionautomatically.** (Note: This feature is only valid when port 5 Giga MAC is implemented)
15:0	GOOD_PKT_CNT5	**Port 5 packet counter for transmitted packets successfully** (Note: This feature is only valid when port 5 Giga MAC is implemented)

10110168 **LEDC** **LED Control** **00E00000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**OLT_MODE**							**EPHY_GPIO_8_5**				**EPHY_GPIO_4_0**				
Type	RW							RW				RW				
Reset	0							0	1	1	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						**LED_SEL**						**LED_POLARITY**				
Type						RW						RW				
Reset						0	0	0				0	0	0	0	0

Bit(s)	Name	Description
31	OLT_MODE	**EPHY OLT Mode** 0: Disable 1: Enable
24:21	EPHY_GPIO_8_5	**EPHY GPIO[8:5]** EPHY_GPIO[8:5] is used to set EPHY initial state which is latched by EPHY SW reset.
20:16	EPHY_GPIO_4_0	**EPHY GPIO[4:0]** EPHY_GPIO[4:0] is used to set EPHY MDIO address which is latched by EPHY SW reset.
10:8	LED_SEL	**LED Source** 0: ESW LED Control 1: EPHY LED Control[0] 2: EPHY LED Control[1] 3: EPHY LED Control[2]
4:0	LED_POLARITY	**Per Port LED Polarity Control** 0: Low Active 1: High Active

5. Abbreviations

Abbrev.	Description
AC	Access Category
ACK	Acknowledge/ Acknowledgement
ACPR	Adjacent Channel Power Ratio
AD/DA	Analog to Digital/Digital to Analog converter
ADC	Analog-to-Digital Converter
AES	Advanced Encryption Standard
AGC	Auto Gain Control
AIFS	Arbitration Inter-Frame Space
AIFSN	Arbitration Inter-Frame Spacing Number
ALC	Asynchronous Layered Coding
A-MPDU	Aggregate MAC Protocol Data Unit
A-MSDU	Aggregation of MAC Service Data Units
AP	Access Point
ASIC	Application-Specific Integrated Circuit
ASME	American Society of Mechanical Engineers
ASYNC	Asynchronous
BA	Block Acknowledgement
BAC	Block Acknowledgement Control
BAR	Base Address Register
BBP	Baseband Processor
BGSEL	Band Gap Select
BIST	Built-In Self-Test
BSC	Basic Spacing between Centers
BJT	
BSSID	Basic Service Set Identifier
BW	Bandwidth
CCA	Clear Channel Assessment
CCK	Complementary Code Keying
CCMP	Counter Mode with Cipher Block Chaining Message Authentication Code Protocol
CCX	Cisco Compatible Extensions
CF-END	Control Frame End
CF-ACK	Control Frame Acknowledgement

Abbrev.	Description
CLK	Clock
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CSR	Control Status Register
CTS	Clear to Send
CW	Contention Window
CWmax	Maximum Contention Window
CWmin	Minimum Contention Window
DAC	Digital-To-Analog Converter
DCF	Distributed Coordination Function
DDONE	DMA Done
DDR	Double Data Rate
DFT	Discrete Fourier Transform
DIFS	DCF Inter-Frame Space
DMA	Direct Memory Access
DSP	Digital Signal Processor
DW	DWORD
EAP	Expert Antenna Processor
EDCA	Enhanced Distributed Channel Access
EECS	EEPROM chip select
EEDI	EEPROM data input
EEDO	EEPROM data output
EEPROM	Electrically Erasable Programmable Read-Only Memory
eFUSE	electrical Fuse
EESK	EEPROM source clock
EIFS	Extended Inter-Frame Space
EIV	Extend Initialization Vector
EVM	Error Vector Magnitude
FDS	Frequency Domain Spreading
FEM	Front-End Module
FEQ	Frequency Equalization
FIFO	First In First Out
FSM	Finite-State Machine
GF	Green Field
GND	Ground
GP	General Purpose

Abbrev.	Description
GPO	General Purpose Output
GPIO	General Purpose Input/Output
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
HT	High Throughput
HTC	High Throughput Control
ICV	Integrity Check Value
IFS	Inter-Frame Space
iNIC	Intelligent Network Interface Card
IV	Initialization Vector
I^2C	Inter-Integrated Circuit
I^2S	Integrated Inter-Chip Sound
I/O	Input/Output
IPI	Idle Power Indicator
IQ	In phase/Quadrature phase
JEDEC	Joint Electron Devices Engineering Council
JTAG	Joint Test Action Group
kbps	kilo (1000) bits per second
KB	Kilo (1024) Bytes
LDO	Low-Dropout Regulator
LDODIG	LDO for DIGital part output voltage
LED	Light-Emitting Diode
LNA	Low Noise Amplifier
LO	Local Oscillator
L-SIG	Legacy Signal Field
MAC	Medium Access Control
MCU	Microcontroller Unit
MCS	Modulation and Coding Scheme
MDC	Management Data Clock
MDIO	Management Data Input/Output
MEM	Memory
MFB	MCS Feedback
MFS	MFB Sequence
MIC	Message Integrity Code
MIMO	Multiple-Input Multiple-Output
MLNA	Monolithic Low Noise Amplifier
MM	Mixed Mode

Abbrev.	Description
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MPDU	MAC Protocol Data Units
MSB	Most Significant Bit
NAV	Network Allocation Vector
NAS	Network-Attached Server
NAT	Network Address Translation
NDP	Null Data Packet
NVM	Non-Volatile Memory
ODT	On-die Termination
Oen	Output Enable
OFDM	Orthogonal Frequency-Division Multiplexing
OSC	Open Sound Control
PA	Power Amplifier
PAPE	Provider Authentication Policy Extension
PBC	Push Button Configuration
PBF	Packet Buffer
PCB	Printed Circuit Board
PCF	Point Coordination Function
PCM	Pulse-Code Modulation
PHY	Physical Layer
PIFS	PCF Interframe Space
PLCP	Physical Layer Convergence Protocol
PLL	Phase-Locked Loop
PME	Physical Medium Entities
PMU	Power Management Unit
PN	Packet Number
PROM	Programmable Read-Only Memory
PSDU	Physical layer Service Data Unit
PSI	Power supply Strength Indication
PSM	Power Save Mode
PTN	Packet Transport Network
QoS	Quality of Service
RDG	Reverse Direction Grant
RAM	Random Access Memory
RF	Radio Frequency

Abbrev.	Description
RGMII	Reduced Gigabit Media Independent Interface
RH	Relative Humidity
RoHS	Restriction on Hazardous Substances
ROM	Read-Only Memory
RSSI	Received Signal Strength Indication (Indicator)
RTS	Request to Send
RvMII	Reverse Media Independent Interface
Rx	Receive
RXD	Received Data
RXINFO	Receive Information
RXWI	Receive Wireless Information
S	Stream
SDXC	Secure Digital eXtended Capacity
SDIO	Secure Digital Input Output
SDRAM	Synchronous Dynamic Random Access Memory
SEC	Security
SGI	Short Guard Interval
SIFS	Short Inter-Frame Space
SoC	System-on-a-Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SSCG	Spread Spectrum Clock Generator
STBC	Space–Time Block Code
SW	Switch Regulator
TA	Transmitter Address
TBTT	Target Beacon Transmission Time

Abbrev.	Description
TDLS	Tunnel Direct Link Setup
TKIP	Temporal Key Integrity Protocol
TRSW	Tx/Rx Switch
TSF	Timing Synchronization Function
TSSI	Transmit Signal Strength Indication
Tx	Transmit
TxBF	Transmit Beamforming
TXD	Transmitted Data
TXDAC	Transmit Digital-Analog Converter
TXINFO	Transmit Information
TXOP	Opportunity to Transmit
TXWI	Tx Wireless Information
UART	Universal Asynchronous Rx/ Tx
USB	Universal Serial Bus
UTIF	Universal Test Interface
VGA	Variable Gain Amplifier
VCO	Voltage Controlled Amplifier
VIH	High Level Input Voltage
VIL	Low Level Input Voltage
VoIP	Voice over IP
WCID	Wireless Client Identification
WEP	Wired Equivalent
WI	Wireless Information
WIV	Wireless Information Valid
WMM	Wi-Fi Multimedia
WPA	Wi-Fi Protected Access
WPDMA	Wireless Polarization Division Multiple Access
WS	Word Select

This product is not designed for use in medical and/or life support applications. Do not use this product in these types of equipment or applications. This document is subject to change without notice and Ralink assumes no responsibility for any inaccuracies that may be contained in this document. Ralink reserves the right to make changes in its products to improve function, performance, reliability, and to attempt to supply the best product possible.