



# Allwinner A80 User Manual

Revision 1.3.1

2015/05/12

## Declaration

This documentation is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This documentation neither states nor implies warranty of any kind, including fitness for any particular application.

Third party licences may be required to implement the solution/product. Customers shall be solely responsible to obtain all appropriately required third party licences. Allwinner shall not be liable for any licence fee or royalty due in respect of any required third party licence. Allwinner shall have no warranty, indemnity or other obligations with respect to matters covered under any required third party licence.

## Revision History

Version	Date	Description
V1.0	2014/04/11	Initial version for public release
V1.1	2014/09/29	Change feature description
V1.2	2015/03/26	Add PRCM Description
V1.3	2015/04/23	Add the programming guide of the security system
V1.3.1	2015/05/12	Change security system register

## Table of Contents

Declaration.....	2
Revision History.....	3
Table of Contents .....	4
Chapter 1 About This Documentation .....	64
1.1. Documentation Overview .....	64
1.2. Acronyms and abbreviations.....	64
Chapter 2 Overview .....	69
2.1. Processor Features .....	70
2.1.1. CPU Architecture.....	70
2.1.2. GPU Architecture .....	70
2.1.3. Memory Subsystem .....	70
2.1.3.1. Boot ROM.....	70
2.1.3.2. SDRAM .....	71
2.1.3.3. NAND Flash .....	71
2.1.3.4. SD/MMC Interface .....	71
2.1.4. System Peripheral .....	71
2.1.4.1. Timer .....	72
2.1.4.2. High Speed Timer .....	72
2.1.4.3. OSC24M .....	72
2.1.4.4. GIC.....	72

2.1.4.5.	DMA .....	72
2.1.4.6.	CCU.....	72
2.1.4.7.	PWM .....	73
2.1.4.8.	Security System.....	73
2.1.4.9.	Security ID .....	73
2.1.4.10.	Trustzone .....	73
2.1.4.11.	CPU Configuration.....	73
2.1.4.12.	Power Management .....	74
2.1.5.	Display Subsystem.....	74
2.1.5.1.	Display Engine .....	74
2.1.5.2.	Video Output.....	74
2.1.6.	Video Engine .....	75
2.1.6.1.	Video Decoding .....	75
2.1.6.2.	Video Encoding .....	75
2.1.7.	Image Subsystem .....	75
2.1.7.1.	CSI.....	75
2.1.7.2.	MIPI CSI .....	75
2.1.7.3.	ISP.....	76
2.1.7.4.	Face Detection .....	76
2.1.8.	External Peripherals .....	76
2.1.8.1.	USB.....	77
2.1.8.2.	EMAC.....	77
2.1.8.3.	ADC.....	78

2.1.8.4.	Digital Audio Interface .....	78
2.1.8.5.	Transport Stream .....	78
2.1.8.6.	CIR .....	78
2.1.8.7.	UART.....	78
2.1.8.8.	SPI.....	78
2.1.8.9.	TWI .....	79
2.1.8.10.	One Wire .....	79
2.1.8.11.	RSB™ (Reduced Serial Bus).....	79
2.1.9.	Process and Package .....	79
2.2.	System Block Diagram .....	80
Chapter 3	System .....	81
3.1.	Memory Mapping .....	82
3.1.1.	Application Notes .....	85
3.1.1.1.	BROM Address .....	85
3.2.	Boot System .....	86
3.3.	CCU.....	87
3.3.1.	Overview .....	87
3.3.2.	Functionalities Description.....	88
3.3.2.1.	System Bus .....	88
3.3.2.2.	Bus clock tree .....	89
3.3.3.	Typical Applications.....	90
3.3.4.	Register List .....	91
3.3.5.	Register Description .....	93

3.3.5.1.	PLL_COCPUX Control Register (Default: 0x02001100).....	93
3.3.5.2.	PLL_C1CPUX Control Register (Default: 0x02001100).....	93
3.3.5.3.	PLL_Audio Control Register (Default: 0x00042B14).....	94
3.3.5.4.	PLL_Peripheral0 Control Register (Default: 0x00002800).....	95
3.3.5.5.	PLL_VE Control Register (Default: 0x00042400).....	96
3.3.5.6.	PLL_DDR Control Register (Default: 0x00042400).....	96
3.3.5.7.	PLL_Video0 Control Register (Default: 0x00016300).....	97
3.3.5.8.	PLL_Video1 Control Register (Default: 0x00016300).....	98
3.3.5.9.	PLL_GPU Control Register (Default: 0x00042400).....	98
3.3.5.10.	PLL_DE Control Register (Default: 0x00042400).....	99
3.3.5.11.	PLL_ISP Control Register (Default: 0x00042400).....	100
3.3.5.12.	PLL_Peripheral1 Control Register (Default: 0x00042400).....	100
3.3.5.13.	CPU Clock Source Register (Default: 0x00000000).....	101
3.3.5.14.	Cluster 0 Clock Configuration Register (Default: 0x00000000).....	101
3.3.5.15.	Cluster 1 Clock Configuration Register (Default: 0x00000000).....	102
3.3.5.16.	GTBUS Clock Configuration Register (Default: 0x00000000).....	102
3.3.5.17.	AHB0 Clock Configuration Register (Default: 0x00000000).....	103
3.3.5.18.	AHB1 Clock Configuration Register (Default: 0x00000000).....	103
3.3.5.19.	AHB2 Clock Configuration Register (Default: 0x00000000).....	103
3.3.5.20.	APB0 Clock Configuration Register (Default: 0x00000000).....	104
3.3.5.21.	APB1 Clock Configuration Register (Default: 0x00000000).....	104
3.3.5.22.	CCI-400 Configuration Register (Default: 0x00000000).....	105
3.3.5.23.	ATS Clock Configuration Register (Default: 0x80000000).....	105

3.3.5.24.	Trace Clock Configuration Register (Default: 0x80000000) .....	106
3.3.5.25.	PLL Stable Time Register 0 (Default: 0x000000FF).....	106
3.3.5.26.	PLL Stable Time Register 1 (Default: 0x000000FF).....	106
3.3.5.27.	PLL Stable Status Register (Default: 0x00000000) .....	107
3.3.5.28.	PLL_COCPUX Bias Register (Default: 0x80100000).....	108
3.3.5.29.	PLL_C1CPUX Bias Register (Default: 0x80100000).....	108
3.3.5.30.	PLL_AUDIO Bias Register (Default: 0x00080000).....	108
3.3.5.31.	PLL_PERIPH0 Bias Register (Default: 0x00080000) .....	108
3.3.5.32.	PLL_VE Bias Register (Default: 0x00080000).....	109
3.3.5.33.	PLL_DDR Bias Register (Default: 0x00080000).....	109
3.3.5.34.	PLL_VIDEO0 Bias Register (Default: 0x00100000).....	109
3.3.5.35.	PLL_VIDEO1 Bias Register (Default: 0x00100000).....	109
3.3.5.36.	PLL_GPU Bias Register (Default: 0x00080000).....	110
3.3.5.37.	PLL_DE Bias Register (Default: 0x00080000) .....	110
3.3.5.38.	PLL_ISP Bias Register (Default: 0x00080000).....	110
3.3.5.39.	PLL_PERIPH1 Bias Register (Default: 0x00080000) .....	110
3.3.5.40.	PLL_COCPUX Tuning Register (Default: 0x44404000).....	111
3.3.5.41.	PLL_C1CPUX Tuning Register (Default: 0x44404000).....	111
3.3.5.42.	PLL_AUDIO Pattern Control Register (Default: 0x00000000).....	112
3.3.5.43.	PLL_PERIPH0 Pattern Control Register (Default: 0x00000000).....	112
3.3.5.44.	PLL_VE Pattern Control Register (Default: 0x00000000) .....	113
3.3.5.45.	PLL_DDR Pattern Control Register (Default: 0x00000000).....	113
3.3.5.46.	PLL_VIDEO0 Pattern Control Register (Default: 0x00000000) .....	114



3.3.5.47.	PLL_VIDEO1 Pattern Control Register (Default: 0x00000000) .....	115
3.3.5.48.	PLL_GPU Pattern Control Register (Default: 0x00000000).....	115
3.3.5.49.	PLL_DE Pattern Control Register (Default: 0x00000000) .....	116
3.3.5.50.	PLL_ISP Pattern Control Register (Default: 0x00000000).....	116
3.3.5.51.	PLL_PERIPH1 Pattern Control Register (Default: 0x00000000).....	117
3.3.5.52.	Clock Output A Register (Default: 0x00000000) .....	118
3.3.5.53.	Clock Output B Register (Default: 0x00000000).....	118
3.3.6.	Programming Guidelines.....	120
3.3.6.1.	PLL.....	120
3.3.6.2.	BUS.....	120
3.4.	CCU_SCLK.....	121
3.4.1.	Overview .....	121
3.4.2.	Typical Applications.....	122
3.4.3.	Register List .....	123
3.4.4.	Register Description .....	124
3.4.4.1.	NAND0 Clock Register 0 (Default: 0x00000000) .....	124
3.4.4.2.	NAND0 Clock Register 1 (Default: 0x00000000) .....	124
3.4.4.3.	SDMMC0 Clock Register (Default: 0x00000000).....	125
3.4.4.4.	SDMMC1 Clock Register (Default: 0x00000000).....	126
3.4.4.5.	SDMMC2 Clock Register (Default: 0x00000000).....	126
3.4.4.6.	TS Clock Register (Default: 0x00000000).....	127
3.4.4.7.	SS Clock Register (Default: 0x00000000) .....	128
3.4.4.8.	SPI0 Clock Register (Default: 0x00000000) .....	128

3.4.4.9.	SPI1 Clock Register (Default: 0x00000000) .....	129
3.4.4.10.	SPI2 Clock Register (Default: 0x00000000) .....	130
3.4.4.11.	SPI3 Clock Register (Default: 0x00000000) .....	130
3.4.4.12.	DE Clock Register (Default: 0x00000000).....	131
3.4.4.13.	MP Clock Register (Default: 0x0B000000).....	131
3.4.4.14.	LCD0 Clock Register (Default: 0x08000000) .....	132
3.4.4.15.	LCD1 Clock Register (Default: 0x09000000) .....	132
3.4.4.16.	CSI ISP Clock Register (Default: 0x00000000) .....	133
3.4.4.17.	CSI0 Master Clock Register (Default: 0x00000000).....	133
3.4.4.18.	CSI1 Master Clock Register (Default: 0x00000000).....	134
3.4.4.19.	FD Clock Register (Default: 0x0C000000).....	134
3.4.4.20.	VE Clock Register (Default: 0x00000000) .....	135
3.4.4.21.	AVS Clock Register (Default: 0x00000000).....	135
3.4.4.22.	GPU Core Clock Register (Default: 0x00000000).....	135
3.4.4.23.	GPU Memory Clock Register (Default: 0x00000000) .....	136
3.4.4.24.	GPU AXI Clock Register (Default: 0x01000000).....	136
3.4.4.25.	GPADC Clock Register (Default: 0x00000000) .....	136
3.4.4.26.	Bus Clock Gating Register 0 (Default: 0x00000000).....	137
3.4.4.27.	Bus Clock Gating Register 1 (Default: 0x00000000).....	138
3.4.4.28.	Bus Clock Gating Register 2 (Default: 0x00000000).....	139
3.4.4.29.	Bus Clock Gating Register 3 (Default: 0x00040000).....	140
3.4.4.30.	Bus Clock Gating Register 4 (Default: 0x00000000).....	141
3.4.4.31.	Bus Software Reset Register 0 (Default: 0x00000000).....	142

3.4.4.32.	Bus Software Reset Register 1 (Default: 0x00000000) .....	143
3.4.4.33.	Bus Software Reset Register 2 (Default: 0x00000000) .....	144
3.4.4.34.	Bus Software Reset Register 3 (Default: 0x00000000) .....	145
3.4.4.35.	Bus Software Reset Register 4 (Default: 0x00000000) .....	145
3.4.5.	Programming Guidelines .....	147
3.4.5.1.	Clock Switch .....	147
3.4.5.2.	Gating and reset .....	147
3.5.	CPU Configuration .....	148
3.5.1.	Overview .....	148
3.5.2.	Functionalities Description .....	148
3.5.2.1.	Module Operation description .....	148
3.5.3.	Block Diagram .....	149
3.5.4.	Operation Principle .....	150
3.5.5.	Register List .....	151
3.5.6.	Register Description .....	152
3.5.6.1.	Cluster 0 Control Register0(Default :0x60000000).....	152
3.5.6.2.	Cluster 0 Control Register1(Default :0x00000000).....	153
3.5.6.3.	Cluster 0 adb400 pwrdrnreqn Register(Default :0x00000001).....	154
3.5.6.4.	Cluster 1 Control Register0(Default :0x70000000).....	154
3.5.6.5.	Cluster 1 Control Register1(Default :0x80000000).....	155
3.5.6.6.	Cluster 1 adb400 pwrdrnreqn Register(Default :0x00000001).....	156
3.5.6.7.	General Control Register0(Default :0x00000000) .....	156
3.5.6.8.	General Control Register1(Default :0x00000000) .....	157

3.5.6.9.	Cluster0 CPU Status Register(Default : 0x00000000) .....	157
3.5.6.10.	Cluster1 CPU Status Register(Default : 0x00000000) .....	158
3.5.6.11.	Cluster CPU IRQ and FIQ Status Register(Default : 0x00000000) .....	158
3.5.6.12.	Cluster 0 Reset Control Register(Default: 0x01FF1101) .....	159
3.5.6.13.	Cluster 1 Reset Control Register(Default: 0x010F11F0) .....	160
3.5.6.14.	GIC and Jtag Reset Control Register(Default: 0x00000F01) .....	162
3.6.	TimeStamp .....	163
3.6.1.	Overview .....	163
3.6.2.	Register List .....	164
3.6.3.	Timestamp Status Register Description .....	165
3.6.3.1.	Counter Low Register (Default: 0x00000000) .....	165
3.6.3.2.	Counter High Register (Default: 0x00000000) .....	165
3.6.4.	Timestamp Control Register Description .....	166
3.6.4.1.	Timestamp Control Register (Default: 0x00000000) .....	166
3.6.4.2.	Counter Low Register (Default: 0x00000000) .....	166
3.6.4.3.	Counter High Register (Default: 0x00000000) .....	166
3.6.4.4.	Cluster 0 Control Register1(Default :0x00000000).....	166
3.6.5.	Programming Guidelines .....	168
3.7.	System Control .....	169
3.7.1.	Overview .....	169
3.7.2.	System Control Register List .....	170
3.7.3.	System Control Register Description .....	171
3.7.3.1.	Version Register .....	171

3.7.3.2.	EMAC Clock Register (Default: 0x00000000) .....	171
3.7.3.3.	Display Mux Control Register (Default: 0x00000000).....	172
3.8.	Timer.....	173
3.8.1.	Overview .....	173
3.8.2.	Functionalities Description.....	174
3.8.2.1.	Typical Applications .....	174
3.8.2.2.	Functional Block Diagram.....	174
3.8.2.3.	Operation Principle .....	175
3.8.3.	Timer Register List.....	176
3.8.4.	Timer Register Description.....	177
3.8.4.1.	Timer IRQ Enable Register (Default: 0x00000000).....	177
3.8.4.2.	Timer IRQ Status Register (Default: 0x00000000).....	177
3.8.4.3.	Timer 0 Control Register (Default: 0x00000004).....	178
3.8.4.4.	Timer 0 Interval Value Register .....	179
3.8.4.5.	Timer 0 Current Value Register .....	179
3.8.4.6.	Timer 1 Control Register (Default: 0x00000004).....	180
3.8.4.7.	Timer 1 Interval Value Register .....	181
3.8.4.8.	Timer 1 Interval Value Register .....	181
3.8.4.9.	Timer 2 Control Register (Default: 0x00000004).....	181
3.8.4.10.	Timer 2 Interval Value Register .....	182
3.8.4.11.	Timer 2 Current Value Register .....	182
3.8.4.12.	Timer 3 Control Register (Default: 0x00000004).....	183
3.8.4.13.	Timer 3 Interval Value Register .....	184

3.8.4.14.	Timer 3 Current Value Register .....	184
3.8.4.15.	Timer 4 Control Register (Default: 0x00000004).....	184
3.8.4.16.	Timer 4 Interval Value Register .....	185
3.8.4.17.	Timer 4 Current Value Register .....	185
3.8.4.18.	Timer 5 Control Register (Default: 0x00000004).....	186
3.8.4.19.	Timer 5 Interval Value Register .....	187
3.8.4.20.	Timer 5 Current Value Register .....	187
3.8.4.21.	AVS Counter Control Register (Default: 0x00000000).....	187
3.8.4.22.	AVS Counter 0 Register (Default: 0x00000000).....	188
3.8.4.23.	AVS Counter 1 Register (Default: 0x00000000).....	188
3.8.4.24.	AVS Counter Divisor Register (Default: 0x05DB05DB) .....	188
3.8.4.25.	Watchdog IRQ Enable Register (Default: 0x00000000) .....	189
3.8.4.26.	Watchdog Status Register (Default: 0x00000000).....	189
3.8.4.27.	Watchdog Control Register (Default: 0x00000000).....	190
3.8.4.28.	Watchdog Configuration Register (Default: 0x00000001).....	190
3.8.4.29.	Watchdog Mode Register (Default: 0x00000000) .....	190
3.8.5.	Programming Guidelines.....	192
3.8.5.1.	Timer .....	192
3.8.5.2.	Watchdog .....	192
3.9.	Trusted Watchdog .....	193
3.9.1.	Overview .....	193
3.9.2.	Functionalities Description.....	193
3.9.2.1.	Typical Applications .....	193

3.9.2.2.	Functional Block Diagram.....	194
3.9.3.	Operation Modes .....	194
3.9.3.1.	Clock.....	194
3.9.3.2.	Interrupt.....	194
3.9.4.	TWD Register List .....	195
3.9.5.	TWD Register Description .....	196
3.9.5.1.	TWD Status Register (Default: 0x00000000) .....	196
3.9.5.2.	TWD Control Register (Default: 0x00000000) .....	196
3.9.5.3.	TWD Restart Register (Default: 0x00000000).....	197
3.9.5.4.	TWD Low Counter Register (Default: 0x00000000).....	197
3.9.5.5.	TWD High Counter Register (Default: 0x00000000).....	197
3.9.5.6.	TWD Interval Value Register (Default: 0x00000000).....	197
3.9.5.7.	TWD Low Counter Compare Register (Default: 0x00000000).....	198
3.9.5.8.	TWD High Counter Compare Register (Default: 0x00000000).....	198
3.9.5.9.	Secure Storage NV-Counter Register (Default: 0x00000000).....	198
3.9.5.10.	Synchronize Data Counter Register 0 (Default: 0x00000000) .....	198
3.9.5.11.	Synchronize Data Counter Register 1 (Default: 0x00000000) .....	198
3.9.5.12.	Synchronize Data Counter Register 2 (Default: 0x00000000) .....	199
3.9.5.13.	Synchronize Data Counter Register 3 (Default: 0x00000000) .....	199
3.9.6.	Programming Guidelines.....	199
3.10.	High-speed Timer.....	200
3.10.1.	Overview .....	200
3.10.2.	Functionalities Description.....	201

3.10.2.1.	Typical Applications .....	201
3.10.2.2.	Functional Block Diagram.....	201
3.10.3.	Operation Principle .....	201
3.10.4.	HSTimer Register List .....	202
3.10.5.	HSTimer Register Description .....	203
3.10.5.1.	HS Timer IRQ Enable Register (Default: 0x00000000) .....	203
3.10.5.2.	HS Timer IRQ Status Register (Default: 0x00000000) .....	203
3.10.5.3.	HS Timer N Control Register (Default: 0x00000000) .....	204
3.10.5.4.	HS Timer N Interval Value Lo Register .....	205
3.10.5.5.	HS Timer N Interval Value Hi Register.....	205
3.10.5.6.	HS Timer N Current Value Lo Register .....	205
3.10.5.7.	HS Timer N Current Value Hi Register.....	206
3.10.6.	Programming Guidelines.....	207
3.11.	PWM.....	208
3.11.1.	Overview .....	208
3.11.2.	Functionalities Description.....	209
3.11.2.1.	Typical Applications .....	209
3.11.2.2.	Functional Block Diagram.....	209
3.11.3.	Operation Principle .....	210
3.11.3.1.	PWM output pins .....	210
3.11.4.	PWM Register List .....	211
3.11.5.	PWM Register Description .....	212
3.11.5.1.	PWM Channel 0 Control Register .....	212



3.11.5.2.	PWM Channel 0 Period Register .....	213
3.11.5.3.	PWM Channel 1 Control Register (Default: 0x00000000) .....	214
3.11.5.4.	PWM Channel 1 Period Register .....	215
3.11.5.5.	PWM Channel 2 Control Register (Default: 0x00000000) .....	215
3.11.5.6.	PWM Channel 2 Period Register .....	217
3.11.5.7.	PWM Channel 3 Control Register (Default: 0x00000000) .....	217
3.11.5.8.	PWM Channel 3 Period Register .....	218
3.11.6.	Programming Guidelines .....	219
3.11.6.1.	PWM0 normal mode .....	219
3.12.	DMA .....	220
3.12.1.	Overview .....	220
3.12.2.	Signal Description .....	220
3.12.3.	Functionalities Description .....	221
3.12.3.1.	Block Diagram .....	221
3.12.3.2.	DRQ Type and Corresponding Relation .....	223
3.12.3.3.	DMA Descriptor .....	223
3.12.3.4.	Interrupt .....	224
3.12.3.5.	Security .....	224
3.12.3.6.	Clock gating .....	224
3.12.3.7.	Transmission mode .....	224
3.12.3.8.	Un-alignment .....	225
3.12.3.9.	Beyond 4G space .....	225
3.12.3.10.	Clock and reset .....	225

3.12.4.	Operation Principle .....	226
3.12.4.1.	DMA Device Management .....	226
3.12.5.	DMA Register List .....	227
3.12.6.	DMA Register Description .....	228
3.12.6.1.	DMA IRQ Enable Register 0 (Default: 0x00000000) .....	228
3.12.6.2.	DMA IRQ Enable Register 1 (Default: 0x00000000) .....	228
3.12.6.3.	DMA IRQ Pending Register 0 (Default: 0x00000000) .....	229
3.12.6.4.	DMA IRQ Pending Register 1 (Default: 0x00000000) .....	229
3.12.6.5.	DMA Secure Register (Default: 0x00000000).....	230
3.12.6.6.	DMA Gating Register (Default: 0x00000000) .....	230
3.12.6.7.	DMA Status Register (Default: 0x00000000).....	230
3.12.6.8.	DMA Enable Register (Default: 0xC0000000).....	231
3.12.6.9.	DMA Pause Register (Default: 0x00000000) .....	231
3.12.6.10.	DMA Descriptor Address Register (Default: 0x00000000) .....	231
3.12.6.11.	DMA Configuration Register (Default: 0x00000000).....	231
3.12.6.12.	DMA Current Source Address Register (Default: 0x00000000).....	232
3.12.6.13.	DMA Current Destination Register (Default: 0x00000000).....	233
3.12.6.14.	DMA Byte Counter Left Register (Default: 0x00000000) .....	233
3.12.6.15.	DMA Parameter Register (Default: 0x00000000).....	233
3.12.6.16.	DMA Mode Register (Default: 0x00000000) .....	233
3.12.6.17.	DMA Former Descriptor Address Register (Default: 0x00000000) .....	234
3.12.6.18.	DMA Package Number Register (Default: 0x00000000) .....	234
3.12.7.	Programming Guidelines .....	235

3.13.	GIC.....	236
3.13.1.	Interrupt Source .....	236
3.14.	Message Box.....	241
3.14.1.	Overview .....	241
3.14.2.	Functionalities Description.....	242
3.14.2.1.	Typical Applications .....	242
3.14.2.2.	Functional Block Diagram.....	243
3.14.3.	Operation Principle .....	243
3.14.3.1.	Message Box clock gating and software reset.....	243
3.14.3.2.	Message Queue Assignment.....	244
3.14.3.3.	Interrupt request.....	244
3.14.3.4.	Transmit and receive messages .....	244
3.14.4.	Message Box Register List .....	245
3.14.5.	Message Box Register Description .....	246
3.14.5.1.	MSGBox Control Register 0(Default: 0x10101010) .....	246
3.14.5.2.	MSGBox Control Register 1(Default : 0x10101010) .....	247
3.14.5.3.	MSGBox IRQ Enable Register u(u=0,1)(Default : 0x00000000) .....	248
3.14.5.4.	MSGBox IRQ Status Register u(Default : 0x0000AAAA) .....	249
3.14.5.5.	MSGBox FIFO Status Register m(Default : 0x00000000) .....	251
3.14.5.6.	MSGBox Message Status Register m(Default : 0x00000000) .....	251
3.14.5.7.	MSGBox Message Queue Register m(Default : 0x00000000) .....	251
3.14.6.	Programming Guidelines.....	252
3.15.	Spinlock.....	254

3.15.1.	Overview .....	254
3.15.2.	Functionalities Description.....	255
3.15.2.1.	Typical Applications .....	255
3.15.2.2.	Functional Block Diagram.....	255
3.15.3.	Operation Principle .....	256
3.15.3.1.	Spinlock clock gating and software reset .....	256
3.15.3.2.	Take and free a spinlock .....	256
3.15.4.	Spinlock Register List.....	257
3.15.5.	Spinlock Register Description .....	258
3.15.5.1.	Spinlock System Status Register (Default: 0x10000000) .....	258
3.15.5.2.	Spinlock Register Status (Default: 0x00000000) .....	258
3.15.5.3.	Spinlock Register N (N=0 to 31)(Default :0x00000000).....	258
3.15.6.	Programming Guidelines.....	259
3.16.	Security System .....	260
3.16.1.	Overview .....	260
3.16.2.	Block Diagram .....	261
3.16.3.	Security System Register List.....	262
3.16.4.	Security System Register Description .....	263
3.16.4.1.	Security System Control Register .....	263
3.16.4.2.	Security System Interrupt Control Register .....	266
3.16.4.3.	Security System Interrupt Status Register .....	267
3.16.4.4.	Security System Key Address Register .....	268
3.16.4.5.	Security System Pubic Modulus/IV/ Counter Address Register.....	268

3.16.4.6.	Security System DataSrc_Address Register .....	268
3.16.4.7.	Security System DataDst_Address Register .....	268
3.16.4.8.	Security System Data Length Register .....	269
3.16.4.9.	Security System Counter[n] Register .....	269
3.16.4.10.	Security System CLK GATING Register .....	错误!未定义书签。
3.16.4.11.	Security System Counter[n] Register .....	269
3.16.5.	Security System Clock Requirement .....	270
3.16.6.	Programming Guidelines .....	270
3.17.	Security ID .....	272
3.17.1.	Overview .....	272
3.18.	Secure Memory Controller .....	273
3.18.1.	Overview .....	273
3.18.2.	Signal Description .....	273
3.18.3.	Functionalities Description .....	274
3.18.3.1.	Typical Applications .....	274
3.18.4.	Diagram and Table .....	275
3.18.4.1.	DRM Diagram .....	275
3.18.4.2.	Master ID Table .....	275
3.18.4.3.	Region Size Table .....	276
3.18.5.	Operation Modes .....	277
3.18.5.1.	Security inversion .....	277
3.18.5.2.	Inversion is disabled .....	277
3.18.5.3.	Inversion is enabled .....	277

3.18.6.	DRM.....	278
3.18.7.	SMC Register List.....	279
3.18.8.	SMC Register Description.....	280
3.18.8.1.	Configuration Register (Default: 0x00001F0F).....	280
3.18.8.2.	Action Register (Default: 0x00000001).....	280
3.18.8.3.	Lockdown Range Register (Default: 0x00000000).....	280
3.18.8.4.	Lockdown Select Register (Default: 0x00000000).....	281
3.18.8.5.	Interrupt Status Register (Default: 0x00000000).....	281
3.18.8.6.	Interrupt Clear Register (Default: 0x00000000).....	282
3.18.8.7.	Master Bypass Register 0 (Default: 0xFFFFFFFF).....	282
3.18.8.8.	Master Security Register 0 (Default: 0x00000000).....	282
3.18.8.9.	Fail Address Register (Default: 0x00000000).....	283
3.18.8.10.	Fail Control Register (Default: 0x00000000).....	283
3.18.8.11.	Fail ID Register (Default: 0x00001F00).....	283
3.18.8.12.	Speculation Control Register (Default: 0x00000000).....	284
3.18.8.13.	Security Inversion Enable Register (Default: 0x00000000).....	284
3.18.8.14.	Master Bypass Register 1 (Default: 0x00000007).....	285
3.18.8.15.	Master Security Register 1 (Default: 0x00000000).....	285
3.18.8.16.	Master DRM Enable Register (Default: 0x00000000).....	285
3.18.8.17.	DRM Illegal Access Register 0 (Default: 0x00000000).....	286
3.18.8.18.	DRM Illegal Access Register 1 (Default: 0x00000000).....	286
3.18.8.19.	DRM Low Start Address Register (Default: 0x00000000).....	286
3.18.8.20.	DRM High Start Address Register (Default: 0x00000000).....	287

3.18.8.21.	DRM Low End Address Register (Default: 0x00000000) .....	287
3.18.8.22.	DRM High End Address Register (Default: 0x00000000).....	287
3.18.8.23.	Region Setup Low Register (Default: 0x00000000).....	287
3.18.8.24.	Region Setup High Register (Default: 0x00000000) .....	288
3.18.8.25.	Region Attribute Register (Default: 0x00000000) .....	288
3.19.	SMTA .....	290
3.19.1.	Overview .....	290
3.19.2.	Signal Description.....	290
3.19.3.	Functionalities Description.....	291
3.19.3.1.	Typical Applications .....	291
3.19.3.2.	Functional Block Diagram.....	291
3.19.3.3.	SMTA Configuration Table .....	293
3.19.4.	SMTA Register List .....	294
3.19.5.	SMTA Register Description .....	295
3.19.5.1.	SMTA Decode Port0 Status Register (Default: 0x00000000).....	295
3.19.5.2.	SMTA Decode Port0 Set Register (Default: 0x00000000).....	295
3.19.5.3.	SMTA Decode Port0 Clear Register (Default: 0x00000000) .....	295
3.19.5.4.	SMTA Decode Port1 Status Register (Default: 0x00000000).....	296
3.19.5.5.	SMTA Decode Port1 Set Register (Default: 0x00000000).....	296
3.19.5.6.	SMTA Decode Port1 Clear Register (Default: 0x00000000) .....	296
3.19.5.7.	SMTA Decode Port2 Status Register (Default: 0x00000000).....	296
3.19.5.8.	SMTA Decode Port2 Set Register (Default: 0x00000000).....	297
3.19.5.9.	SMTA Decode Port2 Clear Register (Default: 0x00000000) .....	297

3.19.6.	Programming Guidelines .....	298
3.20.	GPADC .....	299
3.20.1.	Overview .....	299
3.20.2.	Block Diagram .....	299
3.20.3.	Clock Tree and ADC Conversion Time .....	300
3.20.3.1.	Clock Tree .....	300
3.20.3.2.	A/D Conversion Time .....	300
3.20.4.	Programming Guide .....	301
3.20.5.	GPADC Register List .....	302
3.20.6.	GPADC Register Description .....	303
3.20.6.1.	ADC Control Register 0 .....	303
3.20.6.2.	ADC Control Register 1 .....	303
3.20.6.3.	ADC Interrupt & FIFO Control Register .....	304
3.20.6.4.	ADC Interrupt& FIFO Status Register .....	305
3.20.6.5.	ADC Data Register .....	305
3.20.6.6.	ADC Common Data Register .....	306
3.20.6.7.	ADC IO Configure Register .....	306
3.20.6.8.	ADC Port Data Register .....	307
3.21.	Thermal Sensor Controller .....	308
3.21.1.	Overview .....	308
3.21.2.	Thermal Sensor Register List .....	309
3.21.3.	Thermal Sensor Register Description .....	310
3.21.3.1.	THS Control Register .....	310



3.21.3.2.	THS Interrupt Control Register .....	310
3.21.3.3.	THS Status Register .....	312
3.21.3.4.	Alarm Threshold Control Register 0 .....	313
3.21.3.5.	Alarm Threshold Control Register 1 .....	313
3.21.3.6.	Alarm Threshold Control Register 2 .....	314
3.21.3.7.	Alarm Threshold Control Register 3 .....	314
3.21.3.8.	Shutdown Threshold Control Register 0 .....	314
3.21.3.9.	Shutdown Threshold Control Register 1 .....	315
3.21.3.10.	Shutdown Threshold Control Register 2 .....	315
3.21.3.11.	Shutdown Threshold Control Register 3 .....	315
3.21.3.12.	Average Filter Control Register .....	316
3.21.3.13.	Thermal Sensor0_1 Calibration Data Register .....	316
3.21.3.14.	Thermal Sensor2_3 Calibration Data Register .....	317
3.21.3.15.	THS0 Data Register .....	317
3.21.3.16.	THS1 Data Register .....	317
3.21.3.17.	THS2 Data Register .....	318
3.21.3.18.	THS3 Data Register .....	318
3.21.4.	Programming Guide .....	319
3.22.	KeyADC .....	321
3.22.1.	Overview .....	321
3.22.2.	Principle of operation .....	321
3.22.3.	KEYADC Register List .....	323
3.22.4.	KEYADC Register Description .....	324

3.22.4.1.	KEYADC Control Register .....	324
3.22.4.2.	KEYADC Interrupt Control Register.....	325
3.22.4.3.	KEYADC Interrupt Status Register.....	327
3.22.4.4.	KEYADC Data Register 0.....	330
3.22.4.5.	KEYADC Data Register 1.....	330
3.23.	R_PRCM.....	331
3.23.1.	Overview .....	331
3.23.2.	Block Diagram .....	331
3.23.3.	Operation Modes .....	333
3.23.3.1.	HOSC .....	333
3.23.3.2.	PLL Gating.....	333
3.23.3.3.	Pad Hold .....	334
3.23.3.4.	Power Switch.....	334
3.23.3.5.	Signal Description.....	334
3.23.3.6.	PLL_AUDIO Control Switch.....	334
3.23.3.7.	CPUS Clock Source.....	335
3.23.3.8.	OSC24 Control .....	335
3.23.3.9.	Super Standby .....	335
3.23.3.10.	NMI IRQ.....	335
3.23.4.	R_PRCM Register List .....	336
3.23.5.	R_PRCM Register Description .....	337
3.23.5.1.	CPUS Reset Register(Default: 0x00000000) .....	337
3.23.5.2.	Cluster0 CPUX Reset Register (Default: 0x0000000F) .....	337

3.23.5.3.	Cluster1 CPUX Reset Register (Default: 0x0000000F) .....	337
3.23.5.4.	CPUS Clock Configuration Register (Default: 0x00010000).....	338
3.23.5.5.	APBS Clock Divide Register (Default: 0x00000000).....	338
3.23.5.6.	APBS Clock Gating Register (Default: 0x00100000) .....	339
3.23.5.7.	PLL Control Register 1 (Default: 0x00040015).....	340
3.23.5.8.	R_ONE_WIRE Clock Register (Default: 0x00000000) .....	340
3.23.5.9.	R_CIR_RX Clock Register (Default: 0x00000000).....	341
3.23.5.10.	R_DAUDIO0 Clock Register (Default: 0x00000000).....	342
3.23.5.11.	R_DAUDIO1 Clock Register (Default: 0x00000000).....	342
3.23.5.12.	APBS Software Reset Register (Default: 0x00000000) .....	342
3.23.5.13.	Cluster0 CPUX Power Off Gating Register (Default: 0x00000000) .....	344
3.23.5.14.	Cluster1 CPUX Power Off Gating Register (Default: 0x0000001F) .....	345
3.23.5.15.	VDD_SYS Power Off Gating Register (Default: 0x00000000).....	345
3.23.5.16.	GPU Power Off Gating Register (Default: 0x00000000) .....	347
3.23.5.17.	VDD_SYS Reset Register (Default: 0x00000001).....	347
3.23.5.18.	Cluster0 CPU0 Power Switch Register (Default: 0x00000000) .....	348
3.23.5.19.	Cluster0 CPU1 Power Switch Register (Default: 0x00000000) .....	348
3.23.5.20.	Cluster0 CPU2 Power Switch Register (Default: 0x00000000) .....	348
3.23.5.21.	Cluster0 CPU3 Power Switch Register (Default: 0x00000000) .....	348
3.23.5.22.	Cluster1 CPU0 Power Switch Register (Default: 0x000000FF) .....	349
3.23.5.23.	Cluster1 CPU1 Power Switch Register (Default: 0x000000FF) .....	349
3.23.5.24.	Cluster1 CPU2 Power Switch Register (Default: 0x000000FF) .....	349
3.23.5.25.	Cluster1 CPU3 Power Switch Register (Default: 0x000000FF) .....	350

3.23.5.26.	Super Standby Flag Register (Default: 0x00000000) .....	350
3.23.5.27.	CPU Software Entry Register (Default: 0x00000000) .....	350
3.23.5.28.	Super Standby Software Entry Register (Default: 0x00000000) .....	350
3.23.5.29.	NMI IRQ Control Register (Default: 0x00000000) .....	351
3.23.5.30.	NMI IRQ Enable Register (Default: 0x00000000) .....	351
3.23.5.31.	NMI IRQ Status Register (Default: 0x00000000) .....	351
3.23.5.32.	PLL_AUDIO Control Register (Default: 0x00042B14) .....	351
3.23.5.33.	PLL_AUDIO Pattern Control Register (Default: 0x00000000).....	352
3.23.5.34.	PLL_AUDIO Control Switch Register (Default: 0x00000000) .....	353
3.23.5.35.	R_PIO Hold Control Register (Default: 0x00000000) .....	353
3.23.5.36.	OSC24M Control Register (Default: 0x00000000).....	354
3.23.6.	Programming Guidelines.....	354
3.24.	Port Controller(CPUx-Port).....	355
3.24.1.	Overview .....	355
3.24.2.	Port Register List.....	356
3.24.3.	Port Register Description.....	357
3.24.3.1.	PA Configuration Register 0.....	357
3.24.3.2.	PA Configuration Register 1.....	359
3.24.3.3.	PA Configuration Register 2.....	361
3.24.3.4.	PA Configuration Register 3.....	361
3.24.3.5.	PA Data Register .....	362
3.24.3.6.	PA Multi-Driving Register 0 .....	362
3.24.3.7.	PA Multi-Driving Register 1 .....	362

3.24.3.8.	PA Pull Register 0.....	363
3.24.3.9.	PA Pull Register 1.....	363
3.24.3.10.	PB Configuration Register 0 .....	363
3.24.3.11.	PB Configuration Register 1 .....	364
3.24.3.12.	PB Configuration Register 2 .....	365
3.24.3.13.	PB Configuration Register 3 .....	365
3.24.3.14.	PB Data Register .....	365
3.24.3.15.	PB Multi-Driving Register 0 .....	366
3.24.3.16.	PB Multi-Driving Register 1 .....	366
3.24.3.17.	PB Pull Register 0.....	366
3.24.3.18.	PB Pull Register 1.....	367
3.24.3.19.	PC Configuration Register 0.....	367
3.24.3.20.	PC Configuration Register 1.....	369
3.24.3.21.	PC Configuration Register 2.....	371
3.24.3.22.	PC Configuration Register 3.....	372
3.24.3.23.	PC Data Register .....	372
3.24.3.24.	PC Multi-Driving Register 0 .....	373
3.24.3.25.	PC Multi-Driving Register 1 .....	373
3.24.3.26.	PC Pull Register 0.....	374
3.24.3.27.	PC Pull Register 1.....	374
3.24.3.28.	PD Configuration Register 0 .....	374
3.24.3.29.	PD Configuration Register 1 .....	376
3.24.3.30.	PD Configuration Register 2 .....	378

3.24.3.31.	PD Configuration Register 3 .....	380
3.24.3.32.	PD Data Register.....	381
3.24.3.33.	PD Multi-Driving Register 0.....	382
3.24.3.34.	PD Multi-Driving Register 1.....	382
3.24.3.35.	PD Pull Register 0 .....	382
3.24.3.36.	PD Pull Register 1 .....	383
3.24.3.37.	PE Configuration Register 0.....	383
3.24.3.38.	PE Configuration Register 1.....	385
3.24.3.39.	PE Configuration Register 2.....	387
3.24.3.40.	PE Configuration Register 3.....	388
3.24.3.41.	PE Data Register .....	388
3.24.3.42.	PE Multi-Driving Register 0 .....	388
3.24.3.43.	PE Multi-Driving Register 1 .....	389
3.24.3.44.	PE Pull Register 0.....	389
3.24.3.45.	PE Pull Register 1.....	389
3.24.3.46.	PF Configuration Register 0.....	390
3.24.3.47.	PF Configuration Register 1.....	391
3.24.3.48.	PF Configuration Register 2.....	392
3.24.3.49.	PF Configuration Register 3.....	392
3.24.3.50.	PF Data Register .....	392
3.24.3.51.	PF Multi-Driving Register 0.....	392
3.24.3.52.	PF Multi-Driving Register 1.....	393
3.24.3.53.	PF Pull Register 0.....	393

3.24.3.54.	PF Pull Register 1.....	393
3.24.3.55.	PG Configuration Register 0.....	394
3.24.3.56.	PG Configuration Register 1.....	396
3.24.3.57.	PG Configuration Register 2.....	398
3.24.3.58.	PG Configuration Register 3.....	398
3.24.3.59.	PG Data Register.....	398
3.24.3.60.	PG Multi-Driving Register 0.....	398
3.24.3.61.	PG Multi-Driving Register 1.....	399
3.24.3.62.	PG Pull Register 0.....	399
3.24.3.63.	PG Pull Register 1.....	399
3.24.3.64.	PH Configuration Register 0.....	400
3.24.3.65.	PH Configuration Register 1.....	401
3.24.3.66.	PH Configuration Register 2.....	403
3.24.3.67.	PH Configuration Register 3.....	405
3.24.3.68.	PH Data Register.....	405
3.24.3.69.	PH Multi-Driving Register 0.....	405
3.24.3.70.	PH Multi-Driving Register 1.....	406
3.24.3.71.	PH Pull Register 0.....	406
3.24.3.72.	PH Pull Register 1.....	406
3.24.3.73.	PA External Interrupt Configuration Register 0.....	407
3.24.3.74.	PA External Interrupt Configuration Register 1.....	408
3.24.3.75.	PA External Interrupt Configuration Register 2.....	408
3.24.3.76.	PA External Interrupt Configuration Register 3.....	409

3.24.3.77.	PA External Interrupt Control Register .....	409
3.24.3.78.	PA External Interrupt Status Register .....	409
3.24.3.79.	PA External Interrupt Debounce Register .....	410
3.24.3.80.	PB External Interrupt Configuration Register 0 .....	410
3.24.3.81.	PB External Interrupt Configuration Register 1 .....	411
3.24.3.82.	PB External Interrupt Configuration Register 2 .....	411
3.24.3.83.	PB External Interrupt Configuration Register 3 .....	412
3.24.3.84.	PB External Interrupt Control Register .....	412
3.24.3.85.	PB External Interrupt Status Register .....	412
3.24.3.86.	PB External Interrupt Debounce Register .....	413
3.24.3.87.	PE External Interrupt Configuration Register 0 .....	413
3.24.3.88.	PE External Interrupt Configuration Register 1 .....	414
3.24.3.89.	PE External Interrupt Configuration Register 2 .....	414
3.24.3.90.	PE External Interrupt Configuration Register 3 .....	415
3.24.3.91.	PE External Interrupt Control Register .....	415
3.24.3.92.	PE External Interrupt Status Register .....	415
3.24.3.93.	PE External Interrupt Debounce Register.....	416
3.24.3.94.	PG External Interrupt Configuration Register 0.....	416
3.24.3.95.	PG External Interrupt Configuration Register 1.....	417
3.24.3.96.	PG External Interrupt Configuration Register 2.....	417
3.24.3.97.	PG External Interrupt Configuration Register 3.....	418
3.24.3.98.	PG External Interrupt Control Register .....	418
3.24.3.99.	PG External Interrupt Status Register .....	418



3.24.3.100.	PG External Interrupt Debounce Register .....	419
3.24.3.101.	PH External Interrupt Configuration Register 0.....	419
3.24.3.102.	PH External Interrupt Configuration Register 1.....	420
3.24.3.103.	PH External Interrupt Configuration Register 2.....	420
3.24.3.104.	PH External Interrupt Configuration Register 3.....	421
3.24.3.105.	PH External Interrupt Control Register.....	421
3.24.3.106.	PH External Interrupt Status Register.....	421
3.24.3.107.	PH External Interrupt Debounce Register .....	422
3.24.3.108.	PA Group Configuration Register.....	422
3.24.3.109.	PB Group Configuration Register .....	423
3.24.3.110.	PC Group Configuration Register .....	424
3.24.3.111.	PD Group Configuration Register .....	424
3.24.3.112.	PE Group Configuration Register.....	425
3.24.3.113.	PF Group Configuration Register.....	425
3.24.3.114.	PG Group Configuration Register .....	426
3.24.3.115.	PH Group Configuration Register .....	427
3.25.	Port Controller(CPUs-Port) .....	428
3.25.1.	Overview .....	428
3.25.2.	Port Register List.....	429
3.25.3.	Port Register Description.....	430
3.25.3.1.	PL Configuration Register 0 .....	430
3.25.3.2.	PL Configuration Register 1 .....	432
3.25.3.3.	PL Configuration Register 2 .....	433

3.25.3.4.	PL Configuration Register 3 .....	433
3.25.3.5.	PL Data Register .....	433
3.25.3.6.	PL Multi-Driving Register 0.....	433
3.25.3.7.	PL Multi-Driving Register 1.....	434
3.25.3.8.	PL Pull Register 0 .....	434
3.25.3.9.	PL Pull Register 1 .....	434
3.25.3.10.	PM Configuration Register 0 .....	435
3.25.3.11.	PM Configuration Register 1 .....	436
3.25.3.12.	PM Configuration Register 2 .....	438
3.25.3.13.	PM Configuration Register 3 .....	438
3.25.3.14.	PM Data Register.....	438
3.25.3.15.	PM Multi-Driving Register 0.....	439
3.25.3.16.	PM Multi-Driving Register 1.....	439
3.25.3.17.	PM Pull Register 0 .....	439
3.25.3.18.	PM Pull Register 1 .....	440
3.25.3.19.	PN Configuration Register 0 .....	440
3.25.3.20.	PN Configuration Register 1 .....	441
3.25.3.21.	PN Configuration Register 2 .....	441
3.25.3.22.	PN Configuration Register 3 .....	441
3.25.3.23.	PN Data Register .....	441
3.25.3.24.	PN Multi-Driving Register 0.....	442
3.25.3.25.	PN Multi-Driving Register 1.....	442
3.25.3.26.	PN Pull Register 0 .....	442

3.25.3.27.	PN Pull Register 1 .....	443
3.25.3.28.	PL External Interrupt Configure Register 0.....	443
3.25.3.29.	PL External Interrupt Configure Register 1.....	443
3.25.3.30.	PL External Interrupt Configure Register 2.....	444
3.25.3.31.	PL External Interrupt Configure Register 3.....	444
3.25.3.32.	PL External Interrupt Control Register .....	444
3.25.3.33.	PL External Interrupt Status Register .....	445
3.25.3.34.	PL External Interrupt Debounce Register .....	445
3.25.3.35.	PM External Interrupt Configure Register 0.....	446
3.25.3.36.	PM External Interrupt Configure Register 1.....	446
3.25.3.37.	PM External Interrupt Configure Register 2.....	447
3.25.3.38.	PM External Interrupt Configure Register 3.....	447
3.25.3.39.	PM External Interrupt Control Register.....	447
3.25.3.40.	PM External Interrupt Status Register.....	447
3.25.3.41.	PM External Interrupt Debounce Register .....	448
3.25.3.42.	PL Group Configuration Register.....	448
3.25.3.43.	PM Group Configuration Register .....	449
Chapter 4	Memory.....	450
4.1.	SDRAM .....	451
4.1.1.	Overview .....	451
4.2.	NAND Flash .....	452
4.2.1.	Overview .....	452
4.2.2.	Block Diagram .....	453

4.2.3.	NDFC Timing Diagram .....	454
4.2.4.	NDFC Operation Guide .....	460
Chapter 5	Graphic .....	462
5.1.	GPU Control.....	463
5.1.1.	GPU Control Module Register List.....	463
5.1.2.	GPU Control Module Register Description .....	464
5.1.2.1.	GCM_IDLE_STATUS_REG .....	464
5.1.2.2.	GCM_QOS_REG .....	464
5.1.2.3.	GCM_INT_PWROFF_GATING_REG .....	464
5.1.2.4.	GCM_INT_PWR_MOD_REG .....	465
5.1.2.5.	GCM_INT_PWR_DLY_REG .....	466
5.1.2.6.	GCM_INT_PWR_EVENT_REQ_REG .....	466
5.1.2.7.	GCM_INT_PWR_RESPONSE_REG.....	467
5.2.	Mixer Processor.....	468
5.2.1.	Overview .....	468
5.2.2.	Block Diagram .....	469
5.2.3.	MP Register List.....	470
5.2.4.	MP Register Description.....	473
5.2.4.1.	Mixer Control Register .....	473
5.2.4.2.	Mixer Status Register .....	474
5.2.4.3.	Input DMA Globe Control Register.....	475
5.2.4.4.	Input DMA Start Address High 4bits Register.....	476
5.2.4.5.	Input DMA Start Address Low 32bits Register .....	477

5.2.4.6.	Input DMA Line Width Register .....	477
5.2.4.7.	Input DMA Memory Block Size Register .....	478
5.2.4.8.	Input DMA Memory Block Coordinate Control Register .....	479
5.2.4.9.	Input DMA setting register .....	480
5.2.4.10.	Input DMA Fill-Color Register .....	484
5.2.4.11.	Color Space Converter 0 Control Register .....	485
5.2.4.12.	Color Space Converter 1 Control Register .....	486
5.2.4.13.	Scaler Control Register .....	488
5.2.4.14.	Scaling Output Size Register .....	489
5.2.4.15.	Scaler Horizontal Scaling Factor Register .....	490
5.2.4.16.	Scaler Vertical Scaling Factor Register .....	490
5.2.4.17.	Scaler Horizontal Start Phase Setting Register .....	491
5.2.4.18.	Scaler Vertical Start Phase Setting Register .....	491
5.2.4.19.	ROP Control Register .....	492
5.2.4.20.	ROP Channel 3 Index 0 Control Table Setting Register .....	495
5.2.4.21.	ROP Channel 3 Index 1 Control Table Setting Register .....	499
5.2.4.22.	Alpha / Color Key Control Register .....	503
5.2.4.23.	Color Key Min Color Register .....	508
5.2.4.24.	Color Key Max Color Register .....	508
5.2.4.25.	Fill Color Of ROP Output Setting Register .....	509
5.2.4.26.	Color Space Converter 2 Control Register .....	509
5.2.4.27.	Output Control Register .....	509
5.2.4.28.	Output Size Register .....	512

5.2.4.29.	Output Address High 4bits Register .....	512
5.2.4.30.	Output Address Low 32bits Register .....	513
5.2.4.31.	Output Line Width Register .....	514
5.2.4.32.	Output Alpha Control Register .....	514
5.2.4.33.	MB Control Register .....	517
5.2.4.34.	CSC0/1 Y/G coefficient register.....	517
5.2.4.35.	CSC0/1 Y/G constant register.....	518
5.2.4.36.	CSC0/1 U/R coefficient register .....	518
5.2.4.37.	CSC0/1 U/R constant register .....	519
5.2.4.38.	CSC0/1 V/B coefficient register.....	520
5.2.4.39.	CSC0/1 V/B constant register.....	520
5.2.4.40.	CSC2 Y/G coefficient register.....	521
5.2.4.41.	CSC2 Y/G constant register.....	522
5.2.4.42.	CSC2 U/R coefficient register .....	522
5.2.4.43.	CSC2 U/R constant register .....	522
5.2.4.44.	CSC2 V/B coefficient register.....	523
5.2.4.45.	CSC2 V/B constant register.....	523
5.2.4.46.	Scaling horizontal filtering coefficient RAM block.....	524
5.2.4.47.	Scaling vertical filtering coefficient RAM block.....	525
5.2.4.48.	Palette table .....	525
5.2.4.49.	Command queue control register .....	526
5.2.4.50.	Command queue status register .....	527
5.2.4.51.	Command queue storage start address register .....	527

5.2.4.52.	Input data pixel sequence table .....	528
5.2.4.53.	Output data pixel sequence .....	536
Chapter 6	Image.....	538
6.1.	CSI.....	539
6.1.1.	Overview .....	539
6.1.2.	Block Diagram .....	540
6.1.3.	CSI Description .....	541
6.1.3.1.	CSI FIFO Distribution .....	541
6.1.3.2.	Pixel format Arrangement.....	544
6.1.3.3.	Timing.....	545
6.1.3.4.	Offset / scale / flip function.....	547
6.1.3.5.	Camera Communication Interface .....	548
6.1.4.	CSI Register list .....	551
6.1.5.	CSI Register Description .....	555
6.1.5.1.	CSI Enable Register .....	555
6.1.5.2.	CSI Interface Configuration Register .....	556
6.1.5.3.	CSI Capture Register .....	560
6.1.5.4.	CSI Synchronization Counter Register .....	564
6.1.5.5.	CSI FIFO Threshold Register.....	564
6.1.5.6.	CSI BT656 Header Configuration Register .....	565
6.1.5.7.	CSI Pattern Generation Length Register .....	566
6.1.5.8.	CSI Pattern Generation Address Register .....	566
6.1.5.9.	CSI Version Register.....	566

6.1.5.10.	CSI Channel_0 configuration register .....	567
6.1.5.11.	CSI Channel_0 scale register .....	572
6.1.5.12.	CSI Channel_0 FIFO 0 output buffer-A address register .....	573
6.1.5.13.	CSI Channel_0 FIFO 1 output buffer-A address register .....	573
6.1.5.14.	CSI Channel_0 FIFO 2 output buffer-A address register .....	573
6.1.5.15.	CSI Channel_0 status register .....	574
6.1.5.16.	CSI Channel_0 interrupt enable register .....	575
6.1.5.17.	CSI Channel_0 interrupt status register .....	577
6.1.5.18.	CSI Channel_0 horizontal size register .....	578
6.1.5.19.	CSI Channel_0 vertical size register .....	578
6.1.5.20.	CSI Channel_0 buffer length register .....	579
6.1.5.21.	CSI Channel_0 flip size register .....	579
6.1.5.22.	CSI Channel_0 frame clock counter register .....	580
6.1.5.23.	CSI Channel_0 accumulated and internal clock counter register .....	580
6.1.5.24.	CSI Channel_0 FIFO Statistic Register .....	581
6.1.5.25.	CSI Channel_0 PCLK Statistic Register .....	582
6.1.5.26.	CSI Channel_1 configuration register .....	582
6.1.5.27.	CSI Channel_1 scale register .....	588
6.1.5.28.	CSI Channel_1 FIFO 0 output buffer-A address register .....	588
6.1.5.29.	CSI Channel_1 FIFO 1 output buffer-A address register .....	589
6.1.5.30.	CSI Channel_1 FIFO 2 output buffer-A address register .....	589
6.1.5.31.	CSI Channel_1 status register .....	589
6.1.5.32.	CSI Channel_1 interrupt enable register .....	590



6.1.5.33.	CSI Channel_1 interrupt status register .....	592
6.1.5.34.	CSI Channel_1 horizontal size register .....	593
6.1.5.35.	CSI Channel_1 vertical size register.....	594
6.1.5.36.	CSI Channel_1 buffer length register .....	594
6.1.5.37.	CSI Channel_1 flip size register .....	595
6.1.5.38.	CSI Channel_1 frame clock counter register .....	595
6.1.5.39.	CSI Channel_1 accumulated and internal clock counter register .....	596
6.1.5.40.	CSI Channel_1 FIFO Statistic Register .....	596
6.1.5.41.	CSI Channel_1 PCLK Statistic Register .....	597
6.1.5.42.	CSI Channel_2 configuration register .....	597
6.1.5.43.	CSI Channel_2 scale register .....	603
6.1.5.44.	CSI Channel_2 FIFO 0 output buffer-A address register .....	603
6.1.5.45.	CSI Channel_2 FIFO 1 output buffer-A address register .....	604
6.1.5.46.	CSI Channel_2 FIFO 2 output buffer-A address register .....	604
6.1.5.47.	CSI Channel_2 status register .....	604
6.1.5.48.	CSI Channel_2 interrupt enable register .....	605
6.1.5.49.	CSI Channel_2 interrupt status register .....	607
6.1.5.50.	CSI Channel_2 horizontal size register .....	608
6.1.5.51.	CSI Channel_2 vertical size register .....	609
6.1.5.52.	CSI Channel_2 buffer length register .....	609
6.1.5.53.	CSI Channel_2 flip size register .....	610
6.1.5.54.	CSI Channel_2 frame clock counter register .....	610
6.1.5.55.	CSI Channel_2 accumulated and internal clock counter register.....	611

6.1.5.56.	CSI Channel_2 FIFO Statistic Register .....	612
6.1.5.57.	CSI Channel_2 PCLK Statistic Register .....	612
6.1.5.58.	CSI Channel_3 configuration register .....	612
6.1.5.59.	CSI Channel_3 scale register .....	618
6.1.5.60.	CSI Channel_3 FIFO 0 output buffer-A address register .....	619
6.1.5.61.	CSI Channel_3 FIFO 1 output buffer-A address register .....	619
6.1.5.62.	CSI Channel_3 FIFO 2 output buffer-A address register .....	619
6.1.5.63.	CSI Channel_3 status register .....	620
6.1.5.64.	CSI Channel_3 interrupt enable register .....	621
6.1.5.65.	CSI Channel_3 interrupt status register .....	622
6.1.5.66.	CSI Channel_3 horizontal size register .....	623
6.1.5.67.	CSI Channel_3 vertical size register .....	624
6.1.5.68.	CSI Channel_3 buffer length register .....	625
6.1.5.69.	CSI Channel_3 flip size register .....	625
6.1.5.70.	CSI Channel_3 frame clock counter register .....	626
6.1.5.71.	CSI Channel_3 accumulated and internal clock counter register .....	626
6.1.5.72.	CSI Channel_3 FIFO Statistic Register .....	627
6.1.5.73.	CSI Channel_3 PCLK Statistic Register .....	627
6.1.5.74.	CCI Control Register.....	628
6.1.5.75.	CCI Transmission Configuration Register.....	630
6.1.5.76.	CCI Packet Format Register .....	632
6.1.5.77.	CCI Bus Control Register.....	633
6.1.5.78.	CCI Interrupt Control Register .....	634

6.1.5.79.	CCI Line Counter Trigger Control Register .....	635
6.1.5.80.	CCI FIFO Access Register.....	635
Chapter 7	Display.....	636
7.1.	Display Engine System.....	637
7.1.1.	Display Engine Block Diagram .....	637
7.2.	TCON .....	638
7.2.1.	Overview .....	638
7.2.2.	Block Diagram .....	639
7.2.3.	TCON Structure .....	640
7.2.3.1.	Input Port and Stream FIFO.....	640
7.2.3.2.	FIFO1/FIFO2 Attribute: .....	640
7.2.3.3.	Frame Synchronization.....	641
7.2.3.4.	DE_TCON_BRIDGE.....	641
7.2.3.5.	Function description .....	642
7.2.3.6.	Clock Generator .....	642
7.2.4.	HV interface.....	642
7.2.5.	CCIR interface.....	644
7.2.6.	CPU interface.....	645
7.2.7.	LVDS interface .....	646
7.2.7.1.	Single channel: JEDIA mode .....	646
7.2.7.2.	Single channel: NS mode.....	646
7.2.7.3.	Dual Channels: NS Mode.....	647
7.2.7.4.	Control signal and data port mapping.....	647

7.2.7.5.	Gamma correction .....	647
7.2.7.6.	Color Enhance Unit.....	648
7.2.7.7.	Color Mapping Unit.....	648
7.2.8.	TCON Register List .....	649
7.2.9.	TCON0 Register Description .....	651
7.2.9.1.	TCON global control register .....	651
7.2.9.2.	TCON global interrupt register0 .....	651
7.2.9.3.	TCON global interrupt register1 .....	653
7.2.9.4.	TCON FRM control register.....	654
7.2.9.5.	TCON0 control register .....	655
7.2.9.6.	TCON0 data clock register .....	657
7.2.9.7.	TCON0 basic timing register0 .....	658
7.2.9.8.	TCON0 basic timing register1 .....	658
7.2.9.9.	TCON0 basic timing register2 .....	659
7.2.9.10.	TCON0 basic timing register3 .....	660
7.2.9.11.	TCON0 HV panel interface register.....	660
7.2.9.12.	TCON0 CPU panel interface register.....	662
7.2.9.13.	TCON0 CPU panel write data register.....	665
7.2.9.14.	TCON0 CPU panel read data register0.....	665
7.2.9.15.	TCON0 CPU panel read data register1.....	665
7.2.9.16.	TCON0 LVDS panel interface register.....	666
7.2.9.17.	TCON0 IO polarity register.....	668
7.2.9.18.	TCON0 IO control register.....	669

7.2.9.19.	TCON ECC FIFO register .....	671
7.2.9.20.	TCON CEU control register.....	671
7.2.9.21.	TCON CEU coefficient register0 .....	672
7.2.9.22.	TCON CEU coefficient register1 .....	673
7.2.9.23.	TCON CEU coefficient register2 .....	673
7.2.9.24.	TCON0 CPU panel trigger register0 .....	674
7.2.9.25.	TCON0 CPU panel trigger register1 .....	674
7.2.9.26.	TCON0 CPU panel trigger register2 .....	674
7.2.9.27.	TCON0 CPU panel trigger register3 .....	675
7.2.9.28.	TCON0 CPU panel trigger register4 .....	676
7.2.9.29.	TCON0 CPU panel trigger register5 .....	676
7.2.9.30.	TCON color map control register .....	677
7.2.9.31.	TCON color map odd line register0 .....	677
7.2.9.32.	TCON color map odd line register1 .....	679
7.2.9.33.	TCON color map even line register0.....	679
7.2.9.34.	TCON color map even line register1.....	679
7.2.10.	TCON1 Register List .....	680
7.2.11.	TCON 1 Register Description .....	682
7.2.11.1.	TCON global control register .....	682
7.2.11.2.	TCON global interrupt register0 .....	682
7.2.11.3.	TCON global interrupt register1 .....	685
7.2.11.4.	TCON1 control register .....	685
7.2.11.5.	TCON1 basic timing register0 .....	686

7.2.11.6.	TCON1 basic timing register1 .....	686
7.2.11.7.	TCON1 basic timing register2 .....	687
7.2.11.8.	TCON1 basic timing register3 .....	687
7.2.11.9.	TCON1 basic timing register4 .....	688
7.2.11.10.	TCON1 basic timing register5 .....	688
7.2.11.11.	TCON1 sync register .....	689
7.2.11.12.	TCON1 IO polarity register.....	689
7.2.11.13.	TCON1 IO control register.....	690
7.2.11.14.	TCON ECC FIFO register .....	691
7.2.11.15.	TCON CEU control register.....	692
7.2.11.16.	TCON CEU coefficient register0 .....	693
7.2.11.17.	TCON CEU coefficient register1 .....	693
7.2.11.18.	TCON CEU coefficient register2 .....	694
7.3.	DEFE .....	695
7.3.1.	Overview .....	695
7.3.2.	Block Diagram .....	696
7.3.3.	DEFE Register List .....	697
7.3.4.	DEFE Register Description .....	702
7.3.4.1.	DEFE Module Enable Register .....	702
7.3.4.2.	DEFE Frame Process Control Register .....	702
7.3.4.3.	DEFE CSC By-Pass Register.....	706
7.3.4.4.	DEFE Algorithm Selection Register.....	707
7.3.4.5.	DEFE Line Interrupt Control Register.....	707

7.3.4.6.	DEFE 3D Pre-Luma Buffer Address Register .....	708
7.3.4.7.	DEFE Input Channel 0 Buffer Address Register .....	708
7.3.4.8.	DEFE Input Channel 1 Buffer Address Register .....	709
7.3.4.9.	DEFE Input Channel 2 Buffer Address Register .....	710
7.3.4.10.	DEFE Field Sequence Register .....	710
7.3.4.11.	DEFE Channel 0 Tile-Based Offset Register .....	711
7.3.4.12.	DEFE Channel 1 Tile-Based Offset Register .....	712
7.3.4.13.	DEFE Channel 2 Tile-Based Offset Register .....	713
7.3.4.14.	DEFE 3D Pre-Chroma Buffer Address Register .....	713
7.3.4.15.	DEFE Channel 0 Pitch Register .....	714
7.3.4.16.	DEFE Channel 1 Pitch Register .....	714
7.3.4.17.	DEFE Channel 2 Pitch Register .....	715
7.3.4.18.	DEFE Input Format Register .....	715
7.3.4.19.	DEFE Write Back Address0 Register .....	721
7.3.4.20.	DEFE Write Back Address1 Register .....	722
7.3.4.21.	DEFE Write Back Address2 Register .....	722
7.3.4.22.	DEFE Output Format Register .....	722
7.3.4.23.	DEFE Interrupt Enable Register .....	724
7.3.4.24.	DEFE Interrupt Status Register .....	725
7.3.4.25.	DEFE Status Register.....	726
7.3.4.26.	DEFE CSC Coefficient 00 Register .....	728
7.3.4.27.	DEFE CSC Coefficient 01 Register .....	728
7.3.4.28.	DEFE CSC Coefficient 02 Register .....	729

7.3.4.29.	DEFE CSC Coefficient 03 Register .....	729
7.3.4.30.	DEFE CSC Coefficient 10 Register .....	729
7.3.4.31.	DEFE CSC Coefficient 11 Register .....	730
7.3.4.32.	DEFE CSC Coefficient 12 Register .....	730
7.3.4.33.	DEFE CSC Coefficient 13 Register .....	731
7.3.4.34.	DEFE CSC Coefficient 20 Register .....	731
7.3.4.35.	DEFE CSC Coefficient 21 Register .....	731
7.3.4.36.	DEFE CSC Coefficient 22 Register .....	732
7.3.4.37.	DEFE CSC Coefficient 23 Register .....	732
7.3.4.38.	DEFE De-interlacing Control Register .....	732
7.3.4.39.	DEFE De-interlacing Diag-Interpolate Register.....	734
7.3.4.40.	DEFE De-interlacing Temp-Difference Register .....	734
7.3.4.41.	DEFE De-interlacing Luma Motion Threshold Register .....	735
7.3.4.42.	DEFE De-interlacing Spatial Compare Register.....	735
7.3.4.43.	DEFE De-interlacing Chroma Diff Register.....	735
7.3.4.44.	DEFE De-interlacing Pre-Frame Luma Address Register.....	736
7.3.4.45.	DEFE De-interlacing Pre-Frame Chroma Address Register.....	736
7.3.4.46.	DEFE De-interlacing Tile Flag0 Address Register.....	737
7.3.4.47.	DEFE De-interlacing Tile Flag1 Address Register.....	737
7.3.4.48.	DEFE De-interlacing Tile Flag Pitch Register.....	737
7.3.4.49.	DEFE De-interlacing Flag Sequence Register.....	738
7.3.4.50.	DEFE Write Back Pitch Enable Register .....	738
7.3.4.51.	DEFE Write Back Channel 3 Pitch Register .....	739



7.3.4.52.	DEFE Write Back Channel 4 Pitch Register .....	739
7.3.4.53.	DEFE Write Back Channel 5 Pitch Register .....	739
7.3.4.54.	DEFE 3D Mode Control Register .....	740
7.3.4.55.	DEFE 3D Channel 0 Buffer Address Register .....	741
7.3.4.56.	DEFE 3D Channel 1 Buffer Address Register .....	742
7.3.4.57.	DEFE 3D Channel 2 Buffer Address Register .....	742
7.3.4.58.	DEFE 3D Channel 0 Tile-Based Offset Register .....	742
7.3.4.59.	DEFE 3D Channel 1 Tile-Based Offset Register .....	743
7.3.4.60.	DEFE 3D Channel 2 Tile-Based Offset Register .....	744
7.3.4.61.	DEFE 3D Write Back Line-Stride Register .....	744
7.3.4.62.	DEFE Channel 0 Input Size Register .....	745
7.3.4.63.	DEFE Channel 0 Output Size Register .....	745
7.3.4.64.	DEFE Channel 0 Horizontal Factor Register .....	746
7.3.4.65.	DEFE Channel 0 Vertical factor Register .....	747
7.3.4.66.	DEFE Channel 0 Horizontal Initial Phase Register .....	748
7.3.4.67.	DEFE Channel 0 Vertical Initial Phase 0 Register .....	748
7.3.4.68.	DEFE Channel 0 Vertical Initial Phase 1 Register .....	748
7.3.4.69.	DEFE Channel 0 Horizontal Tap Offset 0 Register .....	749
7.3.4.70.	DEFE Channel 0 Horizontal Tap Offset 1 Register .....	749
7.3.4.71.	DEFE Channel 0 Vertical Tap Offset Register .....	750
7.3.4.72.	DEFE Channel 1 Input Size Register .....	751
7.3.4.73.	DEFE Channel 1 Output Size Register .....	752
7.3.4.74.	DEFE Channel 1 Horizontal Factor Register .....	752

7.3.4.75.	DEFE Channel 1 Vertical factor Register .....	753
7.3.4.76.	DEFE Channel 1 Horizontal Initial Phase Register .....	754
7.3.4.77.	DEFE Channel 1 Vertical Initial Phase 0 Register .....	754
7.3.4.78.	DEFE Channel 1 Vertical Initial Phase 1 Register .....	754
7.3.4.79.	DEFE Channel 1 Horizontal Tap Offset 0 Register .....	755
7.3.4.80.	DEFE Channel 1 Horizontal Tap Offset 1 Register .....	755
7.3.4.81.	DEFE Channel 1 Vertical Tap Offset Register .....	756
7.3.4.82.	DEFE Channel 0 Horizontal Filter Coefficient Register .....	757
7.3.4.83.	DEFE Channel 0 Horizontal Filter Coefficient Register .....	758
7.3.4.84.	DEFE Channel 0 Vertical Filter Coefficient Register.....	759
7.3.4.85.	DEFE Channel 1 Horizontal Filter Coefficient Register .....	760
7.3.4.86.	DEFE Channel 1 Horizontal Filter Coefficient Register .....	761
7.3.4.87.	DEFE Channel 1 Vertical Filter Coefficient Register.....	762
7.3.4.88.	DEFE Channel 3 Horizontal Filter Coefficient Register .....	763
7.3.4.89.	DEFE Channel 3 Horizontal Filter Coefficient Register .....	764
7.3.4.90.	DEFE Channel 3 Vertical Filter Coefficient Register.....	765
7.3.4.91.	DEFE Input Channel Buffer Address High bits Register .....	766
7.3.4.92.	DEFE 3D Channel Buffer Address High bits Register .....	766
7.3.4.93.	DEFE De-interlacing Buffer Address High bits Register .....	767
7.3.4.94.	DEFE 3D De-interlacing Buffer Address High bits Register .....	768
7.3.4.95.	DEFE Write Back Address High bits Register .....	768
7.4.	DEBE .....	770
7.4.1.	Overview .....	770

7.4.2.	Block Diagram .....	771
7.4.3.	DEBE Register List.....	772
7.4.4.	DEBE Register Description.....	774
7.4.4.1.	DE back-end mode control register.....	774
7.4.4.2.	DE-back color control register.....	776
7.4.4.3.	DE-back display size setting register .....	777
7.4.4.4.	DE-layer size register .....	778
7.4.4.5.	DE-layer coordinate control register .....	778
7.4.4.6.	DE-layer frame buffer pitch register.....	779
7.4.4.7.	DE-layer frame buffer low 32 bit address register .....	780
7.4.4.8.	DE-layer frame buffer high 8 bit address register .....	780
7.4.4.9.	DE-Register buffer control register.....	781
7.4.4.10.	DE-color key MAX register.....	782
7.4.4.11.	DE-color key MIN register .....	783
7.4.4.12.	DE-color key configuration register.....	783
7.4.4.13.	DE-layer attribute control register0 .....	784
7.4.4.14.	DE-layer attribute control register1 .....	788
7.4.4.15.	Pixels sequence table .....	792
7.4.4.16.	DE-HWC coordinate control register .....	797
7.4.4.17.	DE-HWC frame buffer format register.....	798
7.4.4.18.	DE backend input YUV channel control register .....	799
7.4.4.19.	DE backend YUV channel frame buffer low 32 bit address register.....	802
7.4.4.20.	DE backend YUV channel frame buffer high 8 bit address register.....	802

7.4.4.21.	DE backend YUV channel buffer line width register .....	803
7.4.4.22.	E backend output color control register.....	803
7.4.4.23.	DE backend output color R coefficient register.....	805
7.4.4.24.	DE backend output color R constant register.....	805
7.4.4.25.	DE backend output color G coefficient register .....	806
7.4.4.26.	DE backend output color G constant register .....	806
7.4.4.27.	DE backend output color B coefficient register.....	807
7.4.4.28.	DE backend output color B constant register.....	807
7.4.4.29.	DE-HWC pattern memory block .....	808
7.4.4.30.	DE-HWC palette table .....	809
7.4.4.31.	Palette mode.....	810
7.4.4.32.	Internal frame buffer mode: .....	811
7.4.4.33.	Internal frame buffer mode palette table .....	813
7.4.4.34.	Gamma correction mode .....	814
7.4.5.	Display engine memory mapping.....	816
7.5.	CMU .....	817
7.5.1.	Overview .....	817
7.5.2.	Block Diagram .....	818
7.6.	DEU.....	819
7.6.1.	Overview .....	819
7.7.	DRC.....	820
7.7.1.	Overview .....	820
7.7.2.	Block Diagram .....	821

Chapter 8	Interfaces.....	822
8.1.	SD3.0 .....	823
8.1.1.	Overview .....	823
8.1.2.	Block Diagram .....	824
8.1.3.	SD3.0 Timing Diagram.....	824
8.1.4.	SD3.0 Register List .....	825
8.1.5.	SD3.0 Register Description .....	826
8.1.5.1.	SD Global Control Register .....	826
8.1.5.2.	SD Clock Control Register .....	827
8.1.5.3.	SD Timeout Register.....	828
8.1.5.4.	SD Bus Width Register .....	828
8.1.5.5.	SD Block Size Register.....	829
8.1.5.6.	SD Byte Count Register.....	829
8.1.5.7.	SD Command Register.....	829
8.1.5.8.	SD Command Argument Register.....	833
8.1.5.9.	SD Response 0 Register .....	833
8.1.5.10.	SD Response 1 Register .....	833
8.1.5.11.	SD Response 2 Register .....	834
8.1.5.12.	SD Response 3 Register .....	834
8.1.5.13.	SD Interrupt Mask Register .....	834
8.1.5.14.	SD Masked Interrupt Status Register.....	835
8.1.5.15.	SD Raw Interrupt Status Register .....	836
8.1.5.16.	SD Status Register .....	837

8.1.5.17.	SD FIFO Water Level Register .....	840
8.1.5.18.	SD Function Select Register.....	841
8.1.5.19.	SD Auto Command 12 Argument Register .....	843
8.1.5.20.	SD Hardware Reset Register.....	843
8.1.5.21.	SD DMAC Control Register .....	844
8.1.5.22.	SD Descriptor List Base Address Register .....	845
8.1.5.23.	SD DMAC Status Register .....	846
8.1.5.24.	SD DMAC Interrupt Enable Register.....	848
8.1.5.25.	Card Threshold Control Register .....	849
8.1.5.26.	eMMC4.5 DDR Start Bit Detection Control Register .....	850
8.1.5.27.	SD FIFO Register .....	851
8.1.6.	SD/MMC Special Requirement.....	852
8.1.6.1.	SD/MMC Pin List .....	852
8.1.7.	SD/MMC DMA Controller Description .....	852
8.1.7.1.	IDMAC Descriptor Structure.....	852
8.1.7.2.	DES0 definition.....	853
8.1.7.3.	DES1 definition.....	854
8.1.7.4.	DES2 definition.....	854
8.1.7.5.	DES3 definition.....	854
8.1.8.	SD/MMC-COMM .....	855
8.2.	TWI .....	857
8.2.1.	Overview .....	857
8.2.2.	Timing Diagram .....	858

8.2.3.	TWI Controller Register List.....	859
8.2.4.	TWI Controller Register Description.....	860
8.2.4.1.	TWI Slave Address Register .....	860
8.2.4.2.	TWI Extend Address Register.....	861
8.2.4.3.	TWI Data Register .....	861
8.2.4.4.	TWI Control Register.....	861
8.2.4.5.	TWI Status Register .....	863
8.2.4.6.	TWI Clock Register.....	865
8.2.4.7.	TWI Soft Reset Register .....	866
8.2.4.8.	TWI Enhance Feature Register .....	866
8.2.4.9.	TWI Line Control Register .....	866
8.2.4.10.	TWI DVFS Control Register .....	868
8.2.5.	TWI Controller Special Requirement.....	869
8.2.5.1.	TWI Pin List.....	869
8.2.5.2.	TWI Controller Operation.....	869
8.3.	SPI.....	870
8.3.1.	Overview .....	870
8.3.2.	SPI Timing Diagram .....	870
8.3.3.	SPI Register List .....	872
8.3.4.	SPI Register Description .....	873
8.3.4.1.	SPI Version Number Register.....	873
8.3.4.2.	SPI Global Control Register.....	873
8.3.4.3.	SPI Transfer Control Register .....	874

8.3.4.4.	SPI Interrupt Control Register .....	878
8.3.4.5.	SPI Interrupt Status Register .....	880
8.3.4.6.	SPI FIFO Control Register.....	882
8.3.4.7.	SPI FIFO Status Register.....	885
8.3.4.8.	SPI Wait Clock Register.....	886
8.3.4.9.	SPI Clock Control Register .....	886
8.3.4.10.	SPI Master Burst Counter Register .....	887
8.3.4.11.	SPI Master Transmit Counter Register.....	888
8.3.4.12.	SPI Master Burst Control Counter Register .....	888
8.3.4.13.	SPI DMA Control Register.....	889
8.3.4.14.	SPI TX Data Register .....	890
8.3.4.15.	SPI RX Data Register .....	890
8.3.4.16.	SPI Special Requirement .....	891
8.3.5.	SPI Pin List .....	891
8.3.6.	SPI Module Clock Source and Frequency.....	891
8.4.	UART.....	892
8.4.1.	Overview .....	892
8.4.2.	UART Timing Diagram .....	893
8.4.3.	UART Register List .....	894
8.4.4.	UART Register Description.....	895
8.4.4.1.	UART Receiver Buffer Register .....	895
8.4.4.2.	UART Transmit Holding Register.....	895
8.4.4.3.	UART Divisor Latch Low Register.....	896



8.4.4.4.	UART Divisor Latch High Register .....	896
8.4.4.5.	UART Interrupt Enable Register .....	897
8.4.4.6.	UART Interrupt Identity Register .....	898
8.4.4.7.	UART FIFO Control Register .....	900
8.4.4.8.	UART Line Control Register .....	902
8.4.4.9.	UART Modem Control Register .....	904
8.4.4.10.	UART Line Status Register .....	905
8.4.4.11.	UART Modem Status Register .....	908
8.4.4.12.	UART Scratch Register .....	911
8.4.4.13.	UART Status Register .....	911
8.4.4.14.	UART Transmit FIFO Level Register.....	913
8.4.4.15.	UART Receive FIFO Level Register .....	913
8.4.4.16.	UART DMA Handshake Config Register.....	913
8.4.4.17.	UART Halt TX Register .....	914
8.4.4.18.	UART DBG DLL Register .....	915
8.4.4.19.	UART DBG DLH Register .....	915
8.4.5.	UART Pin List .....	916
8.5.	RSB .....	918
8.5.1.	Overview .....	918
8.5.2.	Terminology Definition.....	918
8.5.3.	RSB Command Set.....	918
8.5.4.	Software Operation Flow .....	919
8.5.5.	RSB Controller Register List .....	922

8.5.6.	RBS Register Description .....	923
8.5.6.1.	RSB Control Register.....	923
8.5.6.2.	RSB Clock Control Register .....	923
8.5.6.3.	RSB Interrupt Enable Register .....	924
8.5.6.4.	RSB Status Register .....	924
8.5.6.5.	RSB Address Register .....	926
8.5.6.6.	RSB Data Buffer Register.....	926
8.5.6.7.	RSB Line Control Register.....	926
8.5.6.8.	RSB Device Mode Control Register .....	928
8.5.6.9.	RSB Command Register.....	928
8.5.6.10.	RSB Device Address Register.....	928
8.5.7.	RSB General Specification .....	929
8.6.	One Wire Interface.....	932
8.6.1.	Overview .....	932
8.6.2.	One Wire Interface Register List.....	937
8.6.3.	One Wire Interface Register Description.....	938
8.6.3.1.	One Wire Data Register.....	938
8.6.3.2.	One Wire Control Register .....	938
8.6.3.3.	One WIRE STANDARD MODE SPECIAL CONTROL REGISTER.....	941
8.6.3.4.	One Wire Standard Mode CRC Register.....	942
8.6.3.5.	One Wire Interrupt Status Register.....	942
8.6.3.6.	One Wire Interrupt Masking Register .....	944
8.6.3.7.	HDQ/One Wire Function Clock Register .....	945

8.6.3.8.	One Wire Line Control Register.....	945
8.6.3.9.	Standard Mode Write Read Timing Control Register.....	946
8.6.3.10.	Standard Mode Reset Presence Timing Control Register .....	947
8.6.3.11.	Simple Mode Write Read Timing Control Register.....	948
8.6.3.12.	Simple Mode Break Timing Control Register .....	949
8.7.	CIR Receiver.....	950
8.7.1.	Overview .....	950
8.7.2.	CIR Receiver Register List .....	951
8.7.3.	CIR Receiver Register Description .....	952
8.7.3.1.	CIR Receiver Control Register.....	952
8.7.3.2.	CIR Receiver Configure Register .....	953
8.7.3.3.	CIR Receiver FIFO Register .....	953
8.7.3.4.	CIR Receiver Interrupt Control Register .....	953
8.7.3.5.	CIR Receiver Status Register.....	954
8.7.3.6.	CIR Receiver Configure Register .....	956
8.8.	USB.....	958
8.8.1.	Block Diagram .....	958
8.8.2.	USB DRD Controller .....	959
8.8.3.	USB Host Controller.....	960
8.8.3.1.	Overview .....	960
8.8.3.2.	Block Diagram .....	961
8.8.3.3.	USB Host Timing Diagram .....	963
8.8.3.4.	USB Host Register List .....	963

8.8.3.5.	EHCI Register Description.....	964
8.8.3.6.	OHCI Register Description.....	983
8.8.3.7.	HCI Interface Control and Status Register.....	1005
8.8.3.8.	HCI Clock Control Register.....	1008
8.8.3.9.	USB Host Special Requirement.....	1010
8.9.	Digital Audio Interface.....	1011
8.9.1.	Overview.....	1011
8.9.2.	Signal Description.....	1012
8.9.2.1.	Digital Audio Interface Pin List.....	1012
8.9.2.2.	Digital Audio Interface Clock Source and Frequency.....	1012
8.9.3.	Functionalities Description.....	1013
8.9.3.1.	Typical Applications.....	1013
8.9.3.2.	Functional Block Diagram.....	1013
8.9.4.	Timing Diagram.....	1014
8.9.5.	Operation Modes.....	1016
8.9.5.1.	System setup and PCM/I2S initialization.....	1016
8.9.5.2.	The channel setup and DMA setup.....	1016
8.9.5.3.	Enable and disable the PCM/I2S.....	1016
8.9.6.	Digital Audio Interface Register List.....	1017
8.9.7.	Digital Audio Interface Register Description.....	1018
8.9.7.1.	Digital Audio Control Register.....	1018
8.9.7.2.	Digital Audio Format Register 0.....	1019
8.9.7.3.	Digital Audio Format Register 1.....	1021

8.9.7.4.	Digital Audio Interrupt Status Register.....	1021
8.9.7.5.	Digital Audio RX FIFO register .....	1022
8.9.7.6.	Digital Audio FIFO Control Register.....	1023
8.9.7.7.	Digital Audio FIFO Status Register.....	1024
8.9.7.8.	Digital Audio DMA & Interrupt Control Register .....	1024
8.9.7.9.	Digital Audio TX FIFO register.....	1025
8.9.7.10.	Digital Audio Clock Divide Register .....	1025
8.9.7.11.	Digital Audio TX Counter register.....	1027
8.9.7.12.	Digital Audio RX Counter register.....	1027
8.9.7.13.	Digital Audio Channel Configuration register.....	1027
8.9.7.14.	Digital Audio TXn Channel Select register .....	1028
8.9.7.15.	Digital Audio TXn Channel Mapping Register.....	1028
8.9.7.16.	Digital Audio RX Channel Select register.....	1030
8.9.7.17.	Digital Audio RX Channel Mapping Register .....	1030
8.10.	Transport Stream.....	1032
8.10.1.	Transport Stream Controller Overview .....	1032
8.10.2.	Block Diagram .....	1033
8.10.3.	Transport Stream Input Timing Diagram.....	1034
8.10.4.	TSC Register List .....	1036
8.10.5.	TSC Register Description .....	1038
8.10.5.1.	TSC Control Register .....	1038
8.10.5.2.	TSC Status Register .....	1038
8.10.5.3.	TSC Port Control Register .....	1038

8.10.5.4.	TSC Port Parameter Register .....	1038
8.10.5.5.	TSC TSF Input Multiplex Control Register .....	1039
8.10.5.6.	TSC Port Output Multiplex Control Register .....	1040
8.10.5.7.	TSG Control and Status Register .....	1040
8.10.5.8.	TSG Packet Parameter Register .....	1041
8.10.5.9.	TSG Interrupt Enable and Status Register .....	1042
8.10.5.10.	TSG Clock Control Register .....	1043
8.10.5.11.	TSG Buffer Base Address Register.....	1043
8.10.5.12.	TSG Buffer Size Register.....	1044
8.10.5.13.	TSG Buffer Pointer Register .....	1044
8.10.5.14.	TSF Control and Status Register.....	1044
8.10.5.15.	TSF Packet Parameter Register .....	1045
8.10.5.16.	TSF Interrupt Enable and Status Register .....	1046
8.10.5.17.	TSF DMA Interrupt Enable Register .....	1047
8.10.5.18.	TSF Overlap Interrupt Enable Register .....	1047
8.10.5.19.	TSF DMA Interrupt Status Register.....	1047
8.10.5.20.	TSF Overlap Interrupt Status Register .....	1048
8.10.5.21.	TSF PCR Control Register .....	1048
8.10.5.22.	TSF PCR Data Register.....	1048
8.10.5.23.	TSF Channel Enable Register .....	1049
8.10.5.24.	TSF Channel PES Enable Register.....	1049
8.10.5.25.	TSF Channel Descramble Enable Register .....	1049
8.10.5.26.	TSF Channel Index Register .....	1050

8.10.5.27.	TSF Channel Control Register .....	1050
8.10.5.28.	TSF Channel Status Register .....	1050
8.10.5.29.	TSF Channel CW Index Register .....	1050
8.10.5.30.	TSF Channel PID Register .....	1051
8.10.5.31.	TSF Channel Buffer Base Address Register .....	1051
8.10.5.32.	TSF Channel Buffer Size Register .....	1051
8.10.5.33.	TSF Channel Buffer Write Pointer Register .....	1052
8.10.5.34.	TSF Channel Buffer Read Pointer Register .....	1052
8.11.	EMAC.....	1054
8.11.1.	Overview .....	1054
8.11.2.	Block Diagram .....	1055
Appendix	.....	1056

# Chapter 1 About This Documentation

## 1.1. Documentation Overview

This documentation provides an overall description of the Allwinner UltraOcta A80 application processor, which will provide instructions to programmers from several sections, including system, memory, graphic, image, display and interface.

## 1.2. Acronyms and abbreviations

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

A		
AES	Advanced Encryption Standard	A specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001
AGC	Automatic Gain Control	An adaptive system found in electronic devices that automatically controls the gain of a signal: the average output signal level is fed back to adjust the gain to an appropriate level for a range of input signal levels.
AHB	AMBA High-speed Bus	A bus protocol introduced in Advanced Microcontroller Bus Architecture version 2 published by ARM Ltd company
APB	Advanced Peripheral Bus	APB is designed for low bandwidth control accesses, which has an address and data phase similar to AHB, but a much reduced, low complexity signal list (for example no bursts).
AVS	Audio Video Standard	A compression standard for digital audio and video
C		
CIR	Consumer IR	The CIR (Consumer IR) interface is used for remote control through infra-red light



CRC	Cyclic Redundancy Check	A type of hash function used to produce a checksum in order to detect errors in data storage or transmission
CSI	CMOS Sensor Interface	The hardware block that interfaces with different image sensor interfaces and provides a standard output that can be used for subsequent image processing
D		
DES	Data Encryption Standard	A previously predominant algorithm for the encryption of electronic data
DEU	Detail Enhancement Unit	A unit used for display engine frontend data post-processing
DLL	Delay-Locked Loop	A digital circuit similar to a phase-locked loop (PLL), with the main difference being the absence of an internal voltage-controlled oscillator, replaced by a delay line
DRC	Dynamic Range Compression	It reduces the volume of loud sounds or amplifies quiet sounds by narrowing or "compressing" an audio signal's dynamic range.
DVFS	Dynamic Voltage and Frequency Scaling	Dynamic voltage scaling is a power management technique where the voltage used in a component is increased or decreased, depending on circumstances. Dynamic frequency scaling is a technique whereby the frequency of a microprocessor can be automatically adjusted on the fly so that the power consumption or heat generated by the chip can be reduced. These two are often used together to save power in mobile devices.
E		
EHCI	Enhanced Host Controller Interface	The register-level interface for a Host Controller for the USB Revision 2.0.
eMMC	Embedded Multi-Media Card	An architecture consisting of an embedded storage solution with MMC interface, flash memory and controller, all in a small BGA package.
F		

FBGA	Fine Ball Grid Array	FBGA is based on BGA technology, but comes with thinner contacts and is mainly used in SoC design
G		
GIC	Generic Interrupt Controller	A centralized resource for supporting and managing interrupts in a system that includes at least one processor
H		
HDMI	High-Definition Multimedia Interface	A compact audio/video interface for transmitting uncompressed digital data
I		
IEP	Image Enhancement Processor	A unit used for the improvement of digital image quality, including DEU, DRC, CMU.
I2S	Inter IC Sound	An electrical serial bus interface standard used for connecting digital audio devices together
L		
LSB	Least Significant Bit	The bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. It is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digits further to the right.
KEYADC	Analog to Digital Converter	Used for KEY Application
M		
MAC	Media Access Control	A sublayer of the data link layer, which provides addressing and channel access control mechanisms that make it possible for several terminals or network nodes to communicate within a multiple access network that incorporates a shared medium, e.g. Ethernet.
MII	Media Independent Interface	An interface originally designed to connect a fast Ethernet MAC-block to a PHY chip, which now has been extended to support reduced signals and increased speeds.

MIPI	Mobile Industry Processor Interface	MIPI alliance is an open membership organization that includes leading companies in the mobile industry that share the objective of defining and promoting open specifications for interfaces inside mobile terminals.
MIPI DSI	MIPI Display Serial Interface	A specification by the Mobile Industry Processor Interface (MIPI) Alliance aimed at reducing the cost of display sub-systems in a mobile device
MSB	Most Significant Bit	The bit position in a binary number having the greatest value, which is sometimes referred to as the left-most bit due to the convention in positional notation of writing more significant digits further to the left
N		
NTSC	National Television System Committee	An analog television system that is used in most of North America, and many other countries
O		
OHCI	Open Host Controller Interface	A register-level interface that enables a host controller for USB to communicate with a host controller driver in software
OSD	On-Screen Display	A feature of visual devices like VCRs and DVD players that displays program, position, and setting data on a connected TV or computer display
P		
PAL	Phase Alternating Line	An analogue television color encoding system used in broadcast television systems in many countries
PCM	Pulse Code Modulation	A method used to digitally represent sampled analog signals
PID	Packet Identifier	Each table or elementary stream in a transport stream is identified by a 13-bit packet ID (PID). A demultiplexer extracts elementary streams from the transport stream in part by looking for packets identified by the same PID.
S		

SPI	Synchronous Peripheral Interface	A synchronous serial data link standard that operates in full duplex mode. Devices communicate in master/slave mode where the master device initiates the data frame.
T		
TP	Touch Panel	A human-machine interactive interface
TS	Transport Stream	A data stream defined by ISO13818-1, which consists of one or more programs with video and audio data.
U		
USB DRD	Universal Serial Bus Dual Role Device	A Dual Role Device controller, which supports both USB Host and USB Device functions .

## Chapter 2 Overview

Allwinner's latest flagship octa-core application processor is a revolutionary advance in mobile processor technology. The A80 packs octa-core big.LITTLE Cortex™-A15/7 CPU architecture in a 28nm process to deliver an outstanding combination of both computing power and efficiency.

Fast, smooth, and fluid graphics drive the user experience on premium devices. A80 features the lightning-fast 64-core PowerVR G6230 GPU from Imagination Technologies, delivering industry-leading graphics performance and enabling console-class performance even on the most graphics-intensive games. In addition to superior graphics performance, the A80 excels in multimedia with an advanced HawkView™ ISP supporting cameras up to 16M, innovative video engine technology with 4Kx2K video encoding/decoding, high resolution displays up to 2560x1600, HDMI output, advanced HD DRM support, and low power LTE connectivity.

Application usage is extremely diverse on tablets and smartphones: users listen to music, watch movies, browse the web, share photos, play games, send emails, and more. Each application requires different levels of processing power, and that means different applications run more efficiently on different cores. To deliver optimal efficiency, the Allwinner A80 features all-new CoolFlex technology that enables devices to seamlessly run different applications on different CPU cores-saving power by giving applications just the right amount of juice that they need.

## 2.1. Processor Features

### 2.1.1. CPU Architecture

The A80 platform is based on octa-core big. LITTLE Cortex™-A15/7 CPU architecture.

- ARMv7 ISA standard instruction set plus Thumb-2 and Jazeller RCT
- NEON with SIMD and VFPv4 support
- Support LPAE
- Support 32KB I-cache and 32KB D-cache per CPU
- Support 2MB+512KB L2-cache

### 2.1.2. GPU Architecture

- 64-core PowerVR G6230 GPU
- Support OpenGL ES 1.1/2.0/3.0, OpenCL 1.x, Direct 9.3 standards
- Geometry ability up to 133M/s
- Effective pixel ability up to 5.33G/s
- Support up to 85 GFLOPs

### 2.1.3. Memory Subsystem

This section consists of:

- Boot ROM
- SDRAM
- NAND Flash
- SD/MMC interface

#### 2.1.3.1. Boot ROM

- On-chip ROM boot loader
- Support secure and non-secure boot
- Support system boot from Raw NAND, eMMC, SPI NOR Flash, and SD/TF card
- Support system code download through USB DRD(Dual Role Device)

### 2.1.3.2. SDRAM

- Compatible with JEDEC standard LPDDR2/LPDDR3/DDR3/DDR3L SDRAM
- Support 8GB address space
- Support dual-channel 64-bit bus width
- Support 2 chip select signals per channel
- 16 address lines and three bank address lines per channel
- Support Memory Dynamic Frequency Scale

### 2.1.3.3. NAND Flash

- Support 8-bit data BUS width
- Support 72-bit ECC per 1024 bytes
- Support 4 flash chips
- Support 1024, 2048, 4096, 8192, 16K, 32K bytes size per page
- Support SDR, ONFI NV-DDR/NV-DDR2 and Toggle DDR/DDR2

### 2.1.3.4. SD/MMC Interface

- Comply to eMMC standard specification V4.5, SD physical layer specification V3.0, SDIO card specification V2.0
- 1/4/8-bit bus width
- Support data rate up to 80MB/s
- Support three SD/MMC controllers
- Support SDIO interrupt detection
- Support hardware CRC generation and error detection
- Support block size from 1 to 65535 bytes
- Support 3.3V/1.8V IO voltage

## 2.1.4. System Peripheral

This section includes:

- Timer
- High Speed Timer
- GIC
- DMA
- CCU
- PWM
- Security System
- Security ID

- Trustzone
- CPU Configuration
- Power Management

#### **2.1.4.1. Timer**

- Support 6 timers
- Support two 33-bit Audio/Video Sync(AVS) counters
- Support one watchdog to generate reset signal or interrupts

#### **2.1.4.2. High Speed Timer**

- Support 5 high speed timers
- Support five counters up to 56 bits
- Clock source is synchronized with AHB clock, which means calculating much more accurate than other timer

#### **2.1.4.3. OSC24M**

- Support 1.8v oscillator
- Support internal RC oscillator

#### **2.1.4.4. GIC**

- Support 16 Software Generated Interrupts(SGIs), 16 Private Peripheral Interrupts(PPIs) and 192 Shared Peripheral Interrupts(SPIs)

#### **2.1.4.5. DMA**

- 16-channel DMA
- Support data width of 8/16/32 bits
- Support linear and IO address modes
- Support data transfer types with memory-to-memory, memory-to-peripheral, peripheral-to-memory

#### **2.1.4.6. CCU**

- 12 PLLs



- Support a 24MHz oscillator and an on-chip RC oscillator
- Support clock configuration for corresponding modules
- Support software-controlled clock gating and software-controlled reset for corresponding modules

#### **2.1.4.7. PWM**

- Support four PWM channel outputs
- Support outputting 2 kinds of waveform: continuous waveform and pulse waveform
- 0% to 100% adjustable duty cycle
- 0Hz~12MHz output frequency

#### **2.1.4.8. Security System**

- Support AES, DES, 3DES, SHA1/224/256, MD5, PRNG, TRNG, RSA2048
- Support ECB, CBC, CTR, CTS modes for AES
- Support ECB, CBC, CTR modes for DES/3DES
- 128-bit, 192-bit and 256-bit key size for AES
- 160-bit hardware PRNG with 175-bit seed
- 256bits TRNG

#### **2.1.4.9. Security ID**

- Support 4Kb EFUSE for chip ID and security application

#### **2.1.4.10. Trustzone**

- Support TrustZone technology
- Support 256KB secure SRAM

#### **2.1.4.11. CPU Configuration**

- Support power clamp
- Support flexible CPU configuration

#### 2.1.4.12. Power Management

- Support DVFS for CPU frequency and voltage adjustment
- Support flexible clock gate and module reset
- Support dynamic frequency adjustment for external DRAM
- Support multiple power domains

#### 2.1.5. Display Subsystem

This section includes:

- Display engine
- Video output

##### 2.1.5.1. Display Engine

- Four movable layers, each layer size up to 8192x8192 pixels
- Support alpha blending / color key
- Support multiple image input formats, including 16/24/32bpp RGB, planar YUV444/420/422/411
- Support hardware cursor
- Support Color Management Unit (CMU) and Dynamic Range Controller (DRC)
- Support realtime write back function
- Ultra-Scaling engine
  - 16/32bpp ARGB/YUV444/420/422/411
  - Support simultaneous input/output size up to 4096x4096 pixels
  - resize ratio from 1/16x to 32x
  - Support 8-tap anti-aliasing filter in horizontal and 4-tap in vertical

##### 2.1.5.2. Video Output

- Support three independent display channels
- Support 3D function
- Support parallel LCD port up to 2048x1536@60Hz resolution
- Support dual-channel LVDS up to 1920x1080@60Hz resolution
- Support 4-lane MIPI DSI (V1.0) up to 1920x1200@60Hz resolution
- Support 4-lane eDP (V1.2) up to 2560x1600@60Hz resolution
- Support HDMI V1.4 output

## 2.1.6. Video Engine

### 2.1.6.1. Video Decoding

- Support video playback up to 4096x2048@30fps
- Support multi-format video playback, including MPEG1/2, MPEG4 SP/ASP GMC, H.263 including Sorenson Spark, H.264 BP/ MP/HP, VP8, WMV9/VC-1, JPEG/MJPEG, etc
- Support H.265 1080P@30fps by software

### 2.1.6.2. Video Encoding

- Support H.264/VP8 video encoding up to 4096x2048@30fps, 1080p@120fps, 720p@240fps
- JPEG baseline: picture size up to 8192x8192
- Support input picture size up to 4800x4800
- Support input format: tiled /YUV planner/YUV semi-planner/ARGB/YUYV/UYVY
- Support Alpha blending
- Support thumb generation
- Support 4x2 scaling ratio: from 1/16 to 64 arbitrary non-integer ratio
- Support rotated input

## 2.1.7. Image Subsystem

### 2.1.7.1. CSI

- Support 12-bit parallel camera sensor
- Support up to 5M pixel camera sensor
- Support video shot up to 720p@30fps

### 2.1.7.2. MIPI CSI

- Support 4-lane MIPI CSI
- Support up to 16M pixel camera sensor
- Support video shot up to 1080p@60fps

### 2.1.7.3. ISP

- Supported input formats: 8/10/12-bit RAW RGB, 8/10-bit YCbCr
- Supported output formats: YCbCr420 semi-planar, YCrCb420 semi-planar, YCbCr422 semi-planar, YCrCb422 semi-planar, YUV420 planar, YUV422 planar.
- Support image mirror flip and rotation
- Support thumb image generation
- Support two output channels
- Support valid picture size up to 4800x4800
- Support speed up to 400M pixels
- Black clamp with horizontal/vertical offset compensation
- Static and dynamic defect pixel correction.
- Super lens shading correction.
- Anisotropic non-linear Bayer interpolation with false color suppression.
- Programmable color correction.
- Advanced contrast enhance
- Advanced saturation adjust
- Advanced spatial (2D) de-noise filter
- Advanced chrominance noise reduction
- Zone-based AE/AF/AWB statistics
- Anti-flick detection statistics
- Histogram statistics

### 2.1.7.4. Face Detection

- Support up to 900 faces per frame
- Support up to 15 frames per second, with up to 15 faces on each frame
- Support 15 ROI region (size up to 320 x 320 per ROI)
- Support front-view face and side-view face: -90/-45/0/45/90 degree
- Support 0/90/180/270 degree detection
- Support input image resize

### 2.1.8. External Peripherals

This section includes:

- USB
- EMAC
- ADC
- Digital Audio Interface
- Transport Stream

- CIR
- UART
- SPI
- TWI
- One Wire
- RSB™

#### 2.1.8.1. USB

- USB 3.0 DRD SIE with both USB 2.0 and USB 3.0 PHY
  - Support Super-Speed(SS,5-Gbps),High-Speed(HS,480-Mbps),Full-Speed(FS,12-Mbps) in Device mode
  - Support Super-Speed(SS,5-Gbps),High-Speed(HS,480-Mbps),Full-Speed(FS,12-Mbps) and Low-Speed(LS,1.5-Mbps) in Host mode
  - Support Device or Host operation at a time
  - Simultaneous IN and OUT transfer support in superspeed mode
  - Implements both static and dynamic power reduction techniques at multiple levels
- Three EHCI/OHCI compliant Host SIE multiplexed with two USB 2.0 analog PHYs, one HSIC PHY
  - Complies with Enhanced Host Controller Interface(EHCI) Specification,Version 1.0,and the Open Host Controller Interface(OHCI) Specification, Version 1.0a
  - HCI0 support High-Speed(HS,480-Mbps),Full-Speed(FS,12-Mbps), and Low-Speed(LS,1.5-Mbps) USB Device through standard USB difference port
  - HCI1 support only High-Speed(HS,480-Mbps) mode through HSIC port
  - HCI1 support only High-Speed(HS,480-Mbps) mode through ULPI Slave port
  - HCI1 support only High-Speed(HS,480-Mbps) mode through ULPI Master port
  - HCI2 support High-Speed(HS,480-Mbps),Full-Speed(FS,12-Mbps),and Low-Speed(LS,1.5-Mbps) USB Device through standard USB difference port

#### 2.1.8.2. EMAC

- Support 10/100/1000Mbps data transfer rate
- Support MII/RGMII PHY interface
- Support full-duplex and half-duplex operation
- Programmable frame length
- Flexible address filtering modes
- Automatic CRC and pad generation controllable on a per-frame basis
- Options for Automatic Pad/CRC Stripping on receive frames
- Programmable Inter Frame Gap(40-96 bit times in steps of 8)
- Support a variety of flexible address filtering modes

#### 2.1.8.3. ADC

- KeyADC with 6-bit resolution
- GPADC with 12-bit resolution

#### 2.1.8.4. Digital Audio Interface

- I2S or PCM configured by software
- Master/Slave mode configurable
- Audio data resolution of 16,20,24
- Support I2S 3 data format:standard I2S,left justified and right justified
- PCM supports linear sample(8-bits or 16-bits),8-bits u-law and A-law companded sample

#### 2.1.8.5. Transport Stream

- Support both Synchronous Parallel Interface (SPI) and Synchronous Serial Interface (SSI)
- Support speed up to 150Mbps for both SPI and SSI interface
- Support 32-channel PID filter
- Configurable SPI and SSI timing parameters
- Hardware packet synchronous byte error detecting
- Hardware PCR packet detecting

#### 2.1.8.6. CIR

- Support a flexible receiver for IR remote
- Programmable FIFO thresholds

#### 2.1.8.7. UART

- Support seven UART controllers
- Compliant with industry-standard 16550 UARTs
- 64-bytes transmit and receive data FIFOs for all UART
- Software/ Hardware Flow Control

#### 2.1.8.8. SPI

- Support four SPI controllers

- Master/Slave configurable
- Mode0~3 are supported for both transmit and receive operation
- DMA-based or interrupt-based operation
- SPI Clock configurable

#### **2.1.8.9. TWI**

- Support seven Two Wire Interface(TWI) controllers
- Support Standard mode(up to 100Kbps) and Fast mode(up to 400kbps)
- Master/Slave configurable
- Allows 10-bits addressing transactions

#### **2.1.8.10. One Wire**

- Support an One Wire controller for signal wire communication
- Support simple mode or standard mode at one time

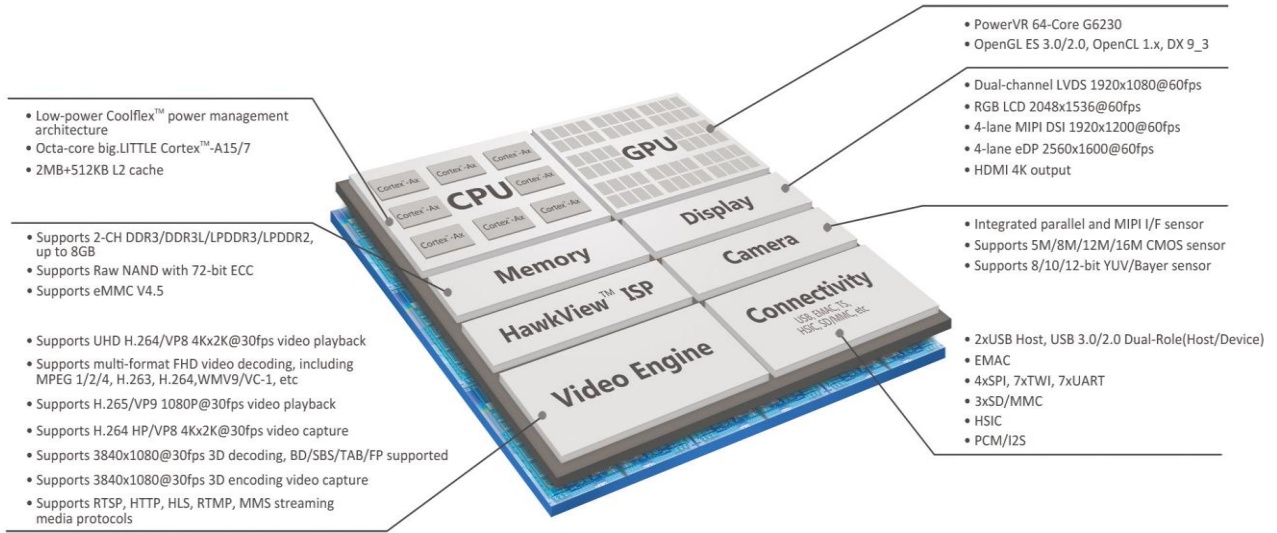
#### **2.1.8.11. RSB™ (Reduced Serial Bus)**

- Support one RSB controller
- Support transfer speed up to 20Mbps using a 2-wire push-pull bus

#### **2.1.9. Process and Package**

- 28nm process
- FCBGA 636 balls, 0.65mm ball pitch, 19mm x 19mm

## 2.2. System Block Diagram





## Chapter 3 System

The chapter describes the A80 system from following sections:

- Memory mapping
- Boot system
- CCU
- CCU\_SCLK
- CPU configuration
- Timestamp
- System control
- Timer
- Trusted watchdog
- High-speed timer
- PWM
- DMA
- GIC
- Message box
- Spinlock
- Security system
- Security ID
- Secure memory controller
- SMTA
- GPADC
- Thermal sensor controller
- KeyADC
- R\_PRCM
- Port controller

### 3.1. Memory Mapping

Module	Address (It is for Cluster CPU)	Size (byte)
BROM-N	0x0000 0000---0x0000 7FFF	32K
BROM-S	0x0000 0000---0x0000 FFFF	64K
SRAM A1	0x0001 0000---0x0001 9FFF	40K
SRAM A2	0x0810 0000---0x0812 7FFF	160K
SRAM B (Secure)	0x0002 0000---0x0005 FFFF	256K
CoreSight Debug Module	0x0140 0000---0x0141 FFFF	128K
TSGEN RO	0x0150 6000---0x0150 6FFF	4K
TSGEN CTRL	0x0160 7000---0x0160 7FFF	4K
CPU_CFG	0x0170 0000---0x0170 03FF	1K
TIMESTAMP	0x0171 0000---0x0172 FFFF	128K
SS	0x01C0 2000---0x01C0 2FFF	4K
NDFCO	0x01C0 3000---0x01C0 3FFF	4K
TS	0x01C0 6000---0x01C0 6FFF	4K
GPU_CTRL	0x01C0 8000---0x01C0 8FFF	4K
GTBUS	0x01C0 9000---0x01C0 9FFF	4K
SMC	0x01C0 B000---0x01C0 BFFF	4K
SID	0x01C0 E000---0x01C0 EFFF	4K
SD/MMC 0	0x01C0 F000---0x01C0 FFFF	4K
SD/MMC 1	0x01C1 0000---0x01C1 0FFF	4K
SD/MMC 2	0x01C1 1000---0x01C1 1FFF	4K
/	0x01C1 2000---0x01C1 2FFF	4K
SD/MMC-COMM	0x01C1 3000---0x01C1 3FFF	4K
SPI0	0x01C1 A000---0x01C1 AFFF	4K
SPI1	0x01C1 B000---0x01C1 BFFF	4K
SPI2	0x01C1 C000---0x01C1 CFFF	4K
SPI3	0x01C1 D000---0x01C1 DFFF	4K
GIC-400	0x01C4 0000---0x01C4 7FFF	32K
DRAMCTL0	0x01C6 3000---0x01C6 3FFF	4K
DRAMCTL1	0x01C6 4000---0x01C6 4FFF	4K
CCI-400	0x01C9 0000---0x01C9 FFFF	64K
GPU Mem	0x0200 0000---0x02FF FFFF	16M
SYS_CTRL	0x0080 0000---0x0080 0FFF	4K
HS TIMER	0x0080 1000---0x0080 1FFF	4K
DMA	0x0080 2000---0x0080 2FFF	4K
MSG-BOX	0x0080 3000---0x0080 3FFF	4K
SPINLOCK	0x0080 4000---0x0080 4FFF	4K
USB-EHCI0/OHCI0	0x00A0 0000---0x00A0 0FFF	4K

USB-EHCI1	0x00A0 1000---0x00A0 1FFF	4K
USB-EHCI2/OHCI2	0x00A0 2000---0x00A0 2FFF	4K
DE_SYS	0x0300 0000---0x0300 FFFF	64K
DISP_SYS	0x0301 0000---0x0301 FFFF	64K
FE0	0x0310 0000---0x0313 FFFF	256K
FE1	0x0314 0000---0x0317 FFFF	256K
FE2	0x0318 0000---0x031B FFFF	256K
BE0	0x0320 0000---0x0323 FFFF	256K
BE1	0x0324 0000---0x0327 FFFF	256K
BE2	0x0328 0000---0x032B FFFF	256K
DEU0	0x0330 0000---0x0333 FFFF	256K
DEU1	0x0334 0000---0x0337 FFFF	256K
DRC0	0x0340 0000---0x0343 FFFF	256K
DRC1	0x0344 0000---0x0347 FFFF	256K
CSI	0x0380 0000---0x039F FFFF	2M
CSI0	0x0380 0000---0x0380 0FFF	4K
CSI0-CCI	0x0380 3000---0x0380 7FFF	20K
ISP	0x0380 8000---0x038F FFFF	992K
CSI1	0x0390 0000---0x0390 0FFF	4K
CSI1-CCI	0x0390 3000---0x0390 7FFF	20K
FD IO	0x03A0 0000---0x03A0 FFFF	64K
FD Memory	0x03A1 0000---0x03A1 FFFF	64K
LCD0	0x03C0 0000---0x03C0 FFFF	64K
LCD1	0x03C1 0000---0x03C1 FFFF	64K
LCD2	0x03C2 0000---0x03C2 FFFF	64K
MP	0x03F0 0000---0x03F1 FFFF	128K
CCU	0x0600 0000---0x0600 03FF	1K
CCU_SCLK	0x0600 0400---0x0600 07FF	1K
PIO	0x0600 0800---0x0600 0BFF	1K
TIMER	0x0600 0C00---0x0600 0FFF	1K
PWM	0x0600 1400---0x0600 17FF	1K
KEYADC	0x0600 1800---0x0600 1BFF	1K
SMTA	0x0600 3400---0x0600 37FF	1K
GPADC	0x0600 4C00---0x0600 4FFF	1K
UART 0	0x0700 0000---0x0700 03FF	1K
UART 1	0x0700 0400---0x0700 07FF	1K
UART 2	0x0700 0800---0x0700 0BFF	1K
UART 3	0x0700 0C00---0x0700 0FFF	1K
UART 4	0x0700 1000---0x0700 13FF	1K
UART 5	0x0700 1400---0x0700 17FF	1K
TWI 0	0x0700 2800---0x0700 2BFF	1K
TWI 1	0x0700 2C00---0x0700 2FFF	1K
TWI 2	0x0700 3000---0x0700 33FF	1K

TWI 3	0x0700 3400---0x0700 37FF	1K
TWI 4	0x0700 3800---0x0700 3BFF	1K
<b>DRAM SPACE</b>		
DRAM SPACE	0x2000 0000---0xFFFF FFFF	3.5G
(Up to 8G)	0x1 0000 0000---0x2 1FFF FFFF	4.5G

### **3.1.1. Application Notes**

#### **3.1.1.1. BROM Address**

There are two Boot ROMs in the SOC: one is BROM-S, called secure BROM, the other is BROM-N, called normal BROM or non-secure BROM. The start addresses of them are 0x0. When the system security feature is enabled, the master, such as Cluster CPUs, can only access the BROM-S, otherwise, the master can only access the BROM-N.

## 3.2. Boot System

The Boot System includes the following features:

- The system will boot in different ways based on whether its security features are enabled
- Support CPU-0 boot process and CPU-0+ boot process
- Support CPU hot plug process
- Support super standby wakeup process
- Support mandatory upgrade process through SDC0 and USB DRD
- Support fast boot process from Raw NAND, eMMC NAND, SPI NOR Flash, and SD/TF card

## 3.3. CCU

### 3.3.1. Overview

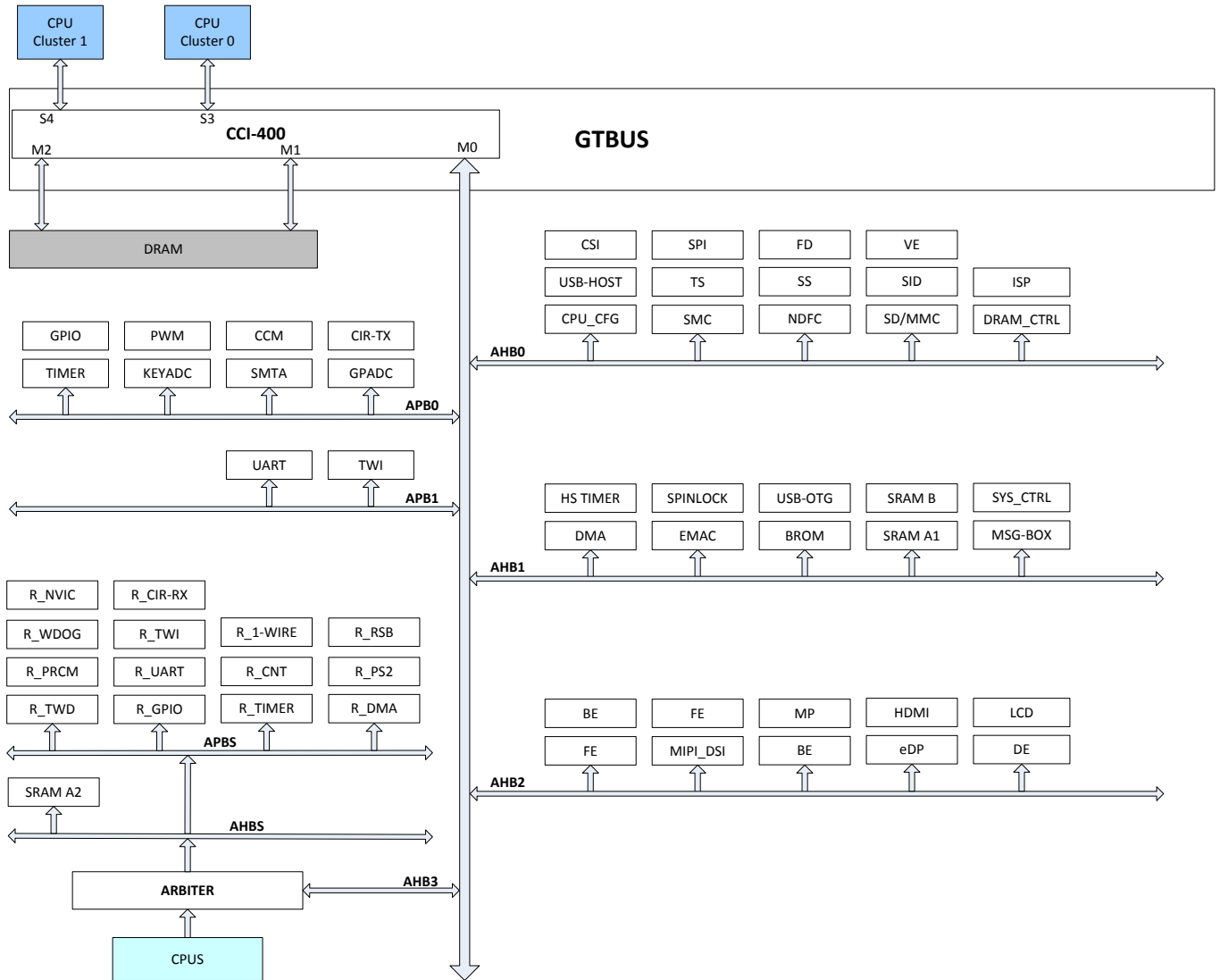
The CCU controls the PLLs configuration and most of the clock generation, division, distribution, synchronization and gating. CCU input signals include the external clock for the reference frequency (24MHz) and the external clock for the RTC (32.768KHz). The outputs from CCU are mostly clocks to other blocks in the system.

The CCU includes the following features:

- 12 PLLs, independent PLL for Cluster 0 CPUX or Cluster 1 CPUX
- Bus divisions
- Clock output control
- PLLs bias control

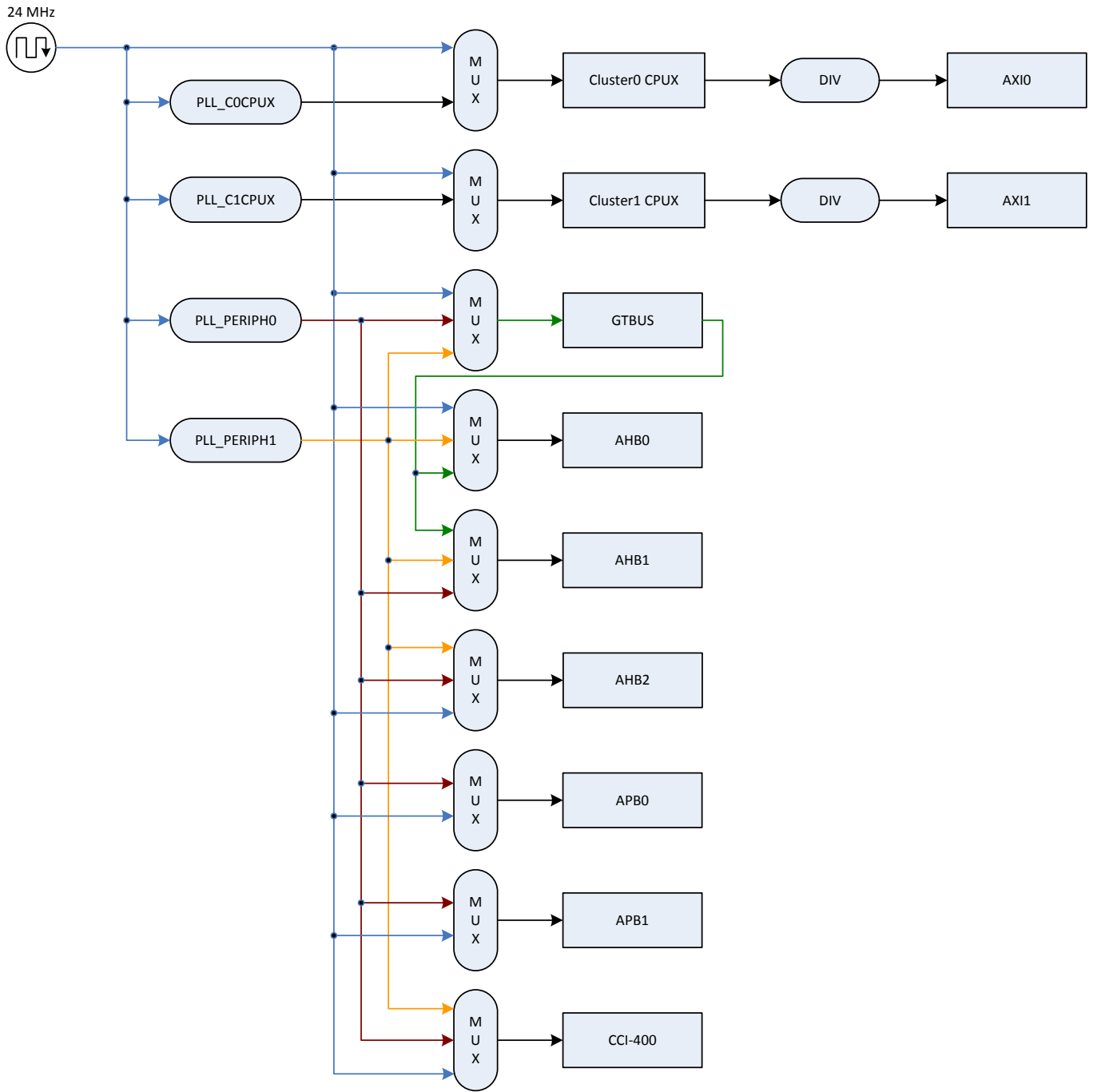
### 3.3.2. Functionalities Description

#### 3.3.2.1. System Bus





3.3.2.2. Bus clock tree



Bus Clock Tree

### 3.3.3. Typical Applications

Clock output of PLL\_C0\_CPUX is used only for Cluster 0 CPUX, and the frequency factor can be dynamically modified for DVFS;

Clock output of PLL\_C1\_CPUX is used only for Cluster 1 CPUX, and the frequency factor can be dynamically modified for DVFS;

Clock output of PLL\_AUDIO can be used for R\_DAUDIO, GPADC, etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_PERIPH0 can be used for GTBUS/AHB0/AHB1/AHB2/APB0/APB1/CCI-400,etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_VE can be used for VE only, and dynamic frequency scaling is not supported;

Clock output of PLL\_DDR can be used for DRAM only, and dynamic frequency scaling is not supported;

Clock output of PLL\_VIDEO0 can be used for LCD0/LCD1/MIPI\_DSI/HDMI, etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_VIDEO1 can be used for MP/LCD0/LCD1/MIPI\_DSI/HDMI/CSI\_MCLK, etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_DE can be used for MP, DE, etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_ISP can be used for ISP, FD, etc, and dynamic frequency scaling is not supported;

Clock output of PLL\_PERIPH1 can be used for GTBUS, AHB0, AHB1, AHB2, CCI-400, and SS module, etc, and dynamic frequency scaling is not supported;

### 3.3.4. Register List

Module Name	Base Address
CCU	0x06000000

Register Name	Offset	Description
PLL_COCPUX_CTRL_REG	0x0000	PLL_COCPUX Control Register
PLL_C1CPUX_CTRL_REG	0x0004	PLL_C1CPUX Control Register
PLL_AUDIO_CTRL_REG	0x0008	PLL_AUDIO Control Register
PLL_PERIPH0_CTRL_REG	0x000C	PLL_Peripheral0 Control Register
PLL_VE_CTRL_REG	0x0010	PLL_VE Control Register
PLL_DDR_CTRL_REG	0x0014	PLL_DDR Control Register
PLL_VIDEO0_CTRL_REG	0x0018	PLL_Video0 Control Register
PLL_VIDEO1_CTRL_REG	0x001C	PLL_Video1 Control Register
PLL_GPU_CTRL_REG	0x0020	PLL_GPU Control Register
PLL_DE_CTRL_REG	0x0024	PLL_DE Control Register
PLL_ISP_CTRL_REG	0x0028	PLL_ISP Control Register
PLL_PERIPH1_CTRL_REG	0x002C	PLL_Peripheral1 Control Register
CPU_CLK_SRC_REG	0x0050	CPU Clock Source Register
C0_CLK_CFG_REG	0x0054	Cluster 0 Clock Configuration Register
C1_CLK_CFG_REG	0x0058	Cluster 1 Clock Configuration Register
GTBUS_CLK_CFG_REG	0x005C	GTBUS Clock Configuration Register
AHB0_CLK_CFG_REG	0x0060	AHB0 Clock Configuration Register
AHB1_CLK_CFG_REG	0x0064	AHB1 Clock Configuration Register
AHB2_CLK_CFG_REG	0x0068	AHB2 Clock Configuration Register
APB0_CLK_CFG_REG	0x0070	APB0 Clock Configuration Register
APB1_CLK_CFG_REG	0x0074	APB1 Clock Configuration Register
CCI400_CLK_CFG_REG	0x0078	CCI-400 Clock Configuration Register
ATS_CLK_CFG_REG	0x0080	ATS Clock Configuration Register
TRACE_CLK_CFG_REG	0x0084	Trace Clock Configuration Register
PLL_STABLE_TIME_REG0	0x0090	PLL Stable Time Register 0
PLL_STABLE_TIME_REG1	0x0094	PLL Stable Time Register 1
PLL_STABLE_STATUS_REG	0x009C	PLL Stable Status Register
PLL_COCPUX_BIAS_REG	0x00A0	PLL_COCPUX Bias Register
PLL_C1CPUX_BIAS_REG	0x00A4	PLL_C1CPUX Bias Register
PLL_AUDIO_BIAS_REG	0x00A8	PLL_AUDIO Bias Register
PLL_PERIPH0_BIAS_REG	0x00AC	PLL_PERIPH0 Bias Register
PLL_VE_BIAS_REG	0x00B0	PLL_VE Bias Register
PLL_DDR_BIAS_REG	0x00B4	PLL_DDR Bias Register
PLL_VIDEO0_BIAS_REG	0x00B8	PLL_VIDEO0 Bias Register
PLL_VIDEO1_BIAS_REG	0x00BC	PLL_VIDEO1 Bias Register

PLL_GPU_BIAS_REG	0x00C0	PLL_GPU Bias Register
PLL_DE_BIAS_REG	0x00C4	PLL_DE Bias Register
PLL_ISP_BIAS_REG	0x00C8	PLL_ISP Bias Register
PLL_PERIPH1_BIAS_REG	0x00CC	PLL_PERIPH1 Bias Register
PLL_COCPUX_TUN_REG	0x00E0	PLL_COCPUX Tuning Register
PLL_C1CPUX_TUN_REG	0x00E4	PLL_C1CPUX Tuning Register
PLL_AUDIO_PAT_CTRL_REG	0x0108	PLL_AUDIO Pattern Control Register
PLL_PERIPH0_PAT_CTRL_REG	0x010C	PLL_PERIPH0 Pattern Control Register
PLL_VE_PAT_CTRL_REG	0x0110	PLL_VE Pattern Control Register
PLL_DDR_PAT_CTRL_REG	0x0114	PLL_DDR Pattern Control Register
PLL_VIDEO0_PAT_CTRL_REG	0x0118	PLL_VIDEO0 Pattern Control Register
PLL_VIDEO1_PAT_CTRL_REG	0x011C	PLL_VIDEO1 Pattern Control Register
PLL_GPU_PAT_CTRL_REG	0x0120	PLL_GPU Pattern Control Register
PLL_DE_PAT_CTRL_REG	0x0124	PLL_DE Pattern Control Register
PLL_ISP_PAT_CTRL_REG	0x0128	PLL_ISP Pattern Control Register
PLL_PERIPH1_CTRL_CFG_REG	0x012C	PLL_PERIPH1 Pattern Control Register
CLK_OUTPUT_A_REG	0x0180	Clock Output A Register
CLK_OUTPUT_B_REG	0x0184	Clock Output B Register

### 3.3.5. Register Description

#### 3.3.5.1. PLL\_COCPUX Control Register (Default: 0x02001100)

Offset: 0x0000			Register Name: PLL_COCPUX_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. The PLL output= 24MHz*N/P. The PLL output is for the Cluster 0 CPUX clock. The PLL output ranges from 200MHz to 3GHz, 408MHz by default.
30:27	/	/	/
26:24	R/W	0x2	PLL_LOCK_TIME. PLL lock time. This bit defines the clock change rate.
23:17	/	/	/
16	R/W	0x0	PLL_OUT_EXT_DIVP. PLL Output external divider P. 0: /1 1: /4 When the output is lower than 288MHz, set P = 4 to output required clock frequency.
15:8	R/W	0x11	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:2	/	/	/
1:0	R/W	0x0	PLL_POSTDIV_M. Post-div factor M. The range is from 0 to 3. M factor is for test only.

#### 3.3.5.2. PLL\_C1CPUX Control Register (Default: 0x02001100)

Offset: 0x0004			Register Name: PLL_C1CPUX_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable.

			1: Enable. The PLL output= 24MHz*N/P. The PLL output is for the Cluster 1 CPUX clock. The PLL output ranges from 200MHz to 3GHz, 408MHz by default.
30:27	/	/	/
26:24	R/W	0x2	PLL_LOCK_TIME. PLL lock time. This bit defines the clock change rate.
23:17	/	/	/
16	R/W	0x0	PLL_OUT_EXT_DIVP. PLL Output external divider P. 0: /1 1: /4 When the output is lower than 288MHz, set P = 4 to output required clock frequency.
15:8	R/W	0x11	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:2	/	/	/
1:0	R/W	0x0	PLL_POSTDIV_M. Post-div factor M. The range is from 0 to 3. M factor is for test only.

### 3.3.5.3. PLL\_Audio Control Register (Default: 0x00042B14)

Offset: 0x0008			Register Name: PLL_AUDIO_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for Audio. The PLL Output = 24MHz*N/(Input_div+1)/(Output_div+1)/(P+1). Its default is 24.5714 MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.

17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x2B	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:6	/	/	/
5:0	R/W	0x14	PLL_POSTDIV_P. Post-div factor P, it is the post counter. The range is from 0 to 63.

#### 3.3.5.4. PLL\_Peripheral0 Control Register (Default: 0x00002800)

Offset: 0x000C			Register Name: PLL_PERIPH0_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for Bus or the peripheral devices. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1)$ . Its default is 960MHz. PLL_PEPH0 output is fixed to 960MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x0	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x28	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

**3.3.5.5. PLL\_VE Control Register (Default: 0x00042400)**

Offset: 0x0010			Register Name: <b>PLL_VE_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for VE. The PLL Output = 24MHz*N/(Input_div+1)/(Output_div+1). Its default is 432MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

**3.3.5.6. PLL\_DDR Control Register (Default: 0x00042400)**

Offset: 0x0014			Register Name: <b>PLL_DDR_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for DRAM. The PLL Output = 24MHz*N/(Input_div+1)/(Output_div+1). Its default is 432MHz.
30	R/W	0x0	PLL_DDR_CFG_UPDATE. PLL_DDR configuration update. When PLL_DDR has been changed, this bit should be set to 1 to validate the PLL, otherwise the change would be invalid. And this bit would be cleared automatically after the PLL change is valid. 0: No effect.



			1: Validating the PLL_DDR.
29:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

### 3.3.5.7. PLL\_Video0 Control Register (Default: 0x00016300)

Offset: 0x0018			Register Name: PLL_VIDEO0_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable.  This PLL is for Display modules or interfaces . For application, PLL Output = 24MHz*N/(Input_div+1). For test, PLL Output = 24MHz*N/(Input_div+1)/(Output_div+1). Its default is 1188MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x0	PLL_OUTPUT_DIV. Div factor = 0 or 1. The factor is for test only.
17	/	/	/
16	R/W	0x1	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x63	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.

7:0	/	/	/
-----	---	---	---

### 3.3.5.8. PLL\_Video1 Control Register (Default: 0x00016300)

Offset: 0x001C			Register Name: PLL_VIDEO1_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for Display modules or interfaces . For application, PLL Output = 24MHz*N/(Input_div+1)/P. For test, PLL Output = 24MHz*N/(Input_div+1)/(Output_div+1)/P. Its default is 1188MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x0	PLL_OUTPUT_DIV. Div factor = 0 or 1. This factor is for test only.
17	/	/	/
16	R/W	0x1	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x63	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:2	/	/	/
1:0	R/W	0x0	PLL_OUTPUT_EXT_DIV. PLL output external divider P. 00: /1 01: /2 10: /4 11: /8

### 3.3.5.9. PLL\_GPU Control Register (Default: 0x00042400)

Offset: 0x0020			Register Name: PLL_GPU_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE.

			0: Disable. 1: Enable. This PLL is for GPU or MP. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1)$ . Its default is 432MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

### 3.3.5.10. PLL\_DE Control Register (Default: 0x00042400)

Offset: 0x0024			Register Name: PLL_DE_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for DE top or MP. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1)$ . Its default is 432MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N.

			PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

### 3.3.5.11. PLL\_ISP Control Register (Default: 0x00042400)

Offset: 0x0028			Register Name: PLL_ISP_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for ISP. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1)$ . Its default is 432MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

### 3.3.5.12. PLL\_Peripheral1 Control Register (Default: 0x00042400)

Offset: 0x002C			Register Name: PLL_PERIPH1_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for Bus or SS. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1)$ . Its default is 432MHz.

30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x24	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, N should be no less than 12.
7:0	/	/	/

### 3.3.5.13. CPU Clock Source Register (Default: 0x00000000)

Offset: 0x0050			Register Name: <b>CPU_CLK_SRC_REG</b>
Bit	R/W	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	C1CPUX_CLK_SRC_SELECT. Cluster 1 CPUs clock source select. 0: OSC24M. 1: PLL_C1_CPUX.
7:1	/	/	/
0	R/W	0x0	COCPUX_CLK_SRC_SELECT. Cluster 0 CPUs clock source select. 0: OSC24M. 1: PLL_C0_CPUX.

### 3.3.5.14. Cluster 0 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0054			Register Name: <b>CO_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:10	/	/	/
9:8	R/W	0x0	ATB0_APB_CLK_DIV. 00: /1 01: /2 1x: /4 ATB0 clock source is Cluster 0 CPUX clock.

7:3	/	/	/
2:0	R/W	0x0	AXIO_CLK_DIV_RATIO. 000: /1 001: /2 010: /3 011: /4 1xx: /4 AXIO clock source is Cluster 0 CPU clock.

### 3.3.5.15. Cluster 1 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0058			Register Name: C1_CLK_CFG_REG
Bit	R/W	Default/Hex	Description
31:10	/	/	/
9:8	R/W	0x0	ATB1_APB_CLK_DIV. 00: /1 01: /2 1x: /4 ATB1 clock source is Cluster 1 CPUX clock.
7:3	/	/	/
2:0	R/W	0x0	AXI1_CLK_DIV_RATIO. 000: /1 001: /2 010: /3 011: /4 1xx: /4 AXI1 clock source is Cluster 1 CPUX clock.

### 3.3.5.16. GTBUS Clock Configuration Register (Default: 0x00000000)

Offset: 0x005C			Register Name: GTBUS_CLK_CFG_REG
Bit	R/W	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	GTBUS_SRC_CLK_SELECT. 00: OSC24M. 01: PLL_PERIPH0. 1x: PLL_PERIPH1.
23:2	/	/	/
1:0	R/W	0x0	GTBUS_CLK_DIV_RATIO. 00: /1 01: /2

			10: /3 11: /4
--	--	--	------------------

### 3.3.5.17. AHB0 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0060			Register Name: <b>AHB0_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	AHB0_SRC_CLK_SELECT. 00: GTBUS_CLK. 01: PLL_PERIPH0. 1x: PLL_PERIPH1.
23:2	/	/	/
1:0	R/W	0x0	AHB0_CLK_DIV_RATIO. 00: /1 01: /2 10: /4 11: /8

### 3.3.5.18. AHB1 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0064			Register Name: <b>AHB1_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	AHB1_SRC_CLK_SELECT. 00: GTBUS_CLK. 01: PLL_PERIPH0. 1x: PLL_PERIPH1.
23:2	/	/	/
1:0	R/W	0x0	AHB1_CLK_DIV_RATIO. 00: /1 01: /2 10: /4 11: /8

### 3.3.5.19. AHB2 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0068	Register Name: <b>AHB2_CLK_CFG_REG</b>
----------------	--

Bit	R/W	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	AHB2_SRC_CLK_SELECT. 00: OSC24M. 01: PLL_PERIPH0. 1x: PLL_PERIPH1.
23:2	/	/	/
1:0	R/W	0x0	AHB2_CLK_DIV_RATIO. 00: /1 01: /2 10: /4 11: /8

### 3.3.5.20. APB0 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0070			Register Name: <b>APB0_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	APB0_SRC_CLK_SELECT. 0: OSC24M. 1: PLL_PERIPH0.
23:2	/	/	/
1:0	R/W	0x0	AHB1_CLK_DIV_RATIO. 00: /1 01: /2 10: /4 11: /8

### 3.3.5.21. APB1 Clock Configuration Register (Default: 0x00000000)

Offset: 0x0074			Register Name: <b>APB1_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	APB1_SRC_CLK_SELECT. 0: OSC24M. 1: PLL_PERIPH0.
23:18	/	/	/
17:16	R/W	0x0	APB1_PREDIV_N. 00: /1 01: /2 10: /4



			11: /8
15:5	/	/	/
4:0	R/W	0x0	APB1_DIV_M. This divide is from 1 to 32.

### 3.3.5.22. CCI-400 Configuration Register (Default: 0x00000000)

Offset: 0x0078			Register Name: <b>CCI400_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	CCI400_SRC_CLK_SELECT. 00: OSC24M. 01: PLL_PERIPH0. 1x: PLL_PERIPH1.
23:2	/	/	/
1:0	R/W	0x0	CCI400_CLK_DIV_RATIO. 00: /1 01: /2 10: /3 11: /4 When the clock source changes from OSC24M to PLL_PERIPH0 or PLL_PERIPH1, you should: 1) Modify the CCI-400 clock division first; 2) Wait after more than ten CCI-400 clock cycles; 3) Switch the S clock source in the end

### 3.3.5.23. ATS Clock Configuration Register (Default: 0x80000000)

Offset: 0x0080			Register Name: <b>ATS_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x1	SCLK_GATING. 0: Clock is off. 1: Clock is on. The SCLK = Clock Source / Divider M.
30:26	/	/	/
25:24	R/W	0x0	ATS_SRC_CLK_SELECT. 00: OSC24M. 01: PLL_PERIPH0. 1x: /
23:3	/	/	/
2:0	R/W	0x0	ATS_CLK_DIV_RATIO_M.

			The divide is from 1 to 8.
--	--	--	----------------------------

### 3.3.5.24. Trace Clock Configuration Register (Default: 0x80000000)

Offset: 0x0084			Register Name: TRACE_CLK_CFG_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x1	SCLK_GATING. 0: Clock is off. 1: Clock is on. The SCLK = Clock Source / Divider M.
30:26	/	/	/
25:24	R/W	0x0	ATS_SRC_CLK_SELECT. 00: OSC24M. 01: PLL_PERIPH0. 1x: /
23:3	/	/	/
2:0	R/W	0x0	ATS_CLK_DIV_RATIO_M. The divide is from 1 to 8.

### 3.3.5.25. PLL Stable Time Register 0 (Default: 0x000000FF)

Offset: 0x0090			Register Name: PLL_STABLE_TIME_REG0
Bit	R/W	Default/Hex	Description
30:16	/	/	/
15:0	R/W	0x00FF	PLL_STABLE_TIME0. PLL stable time (unit: us). Except PLL_COCPUX and PLL_C1CPUX, when the enable/disable or settings of other PLLs changes, corresponding PLL stable bit will change to 1 after the period of time defined in PLL stable time, and that means the PLL clock is stable.

### 3.3.5.26. PLL Stable Time Register 1 (Default: 0x000000FF)

Offset: 0x0094			Register Name: PLL_STABLE_TIME_REG1
Bit	R/W	Default/Hex	Description
30:16	/	/	/
15:0	R/W	0x00FF	PLL_STABLE_TIME1. PLL stable time (unit: us). When the enable/disable or settings of PLL_COCPUX and PLL_C1CPUX,

			changes, corresponding PLL stable bit will change to 1 after the period of time defined in PLL stable time, and that means the PLL clock is stable.
--	--	--	---

### 3.3.5.27. PLL Stable Status Register (Default: 0x00000000)

Offset: 0x009C			Register Name: PLL_STABLE_STATUS_REG
Bit	R/W	Default/Hex	Description
30:12	/	/	/
11	RO	0x0	PLL_PERIPH1_STATUS. 0: Unstable. 1: Stable.
10	RO	0x0	PLL_ISP_STATUS. 0: Unstable. 1: Stable.
9	RO	0x0	PLL_DE_STATUS. 0: Unstable. 1: Stable.
8	RO	0x0	PLL_GPU_STATUS. 0: Unstable. 1: Stable.
7	RO	0x0	PLL_VIDEO1_STATUS. 0: Unstable. 1: Stable.
6	RO	0x0	PLL_VIDEO0_STATUS. 0: Unstable. 1: Stable.
5	RO	0x0	PLL_DDR_STATUS. 0: Unstable. 1: Stable.
4	RO	0x0	PLL_VE_STATUS. 0: Unstable. 1: Stable.
3	RO	0x0	PLL_PERIPH0_STATUS. 0: Unstable. 1: Stable.
2	RO	0x0	PLL_AUDIO_STATUS. 0: Unstable. 1: Stable.
1	RO	0x0	PLL_C1CPUX_STATUS. 0: Unstable. 1: Stable.
0	RO	0x0	PLL_COCPUX_STATUS. 0: Unstable.

			1: Stable.
--	--	--	------------

### 3.3.5.28. PLL\_COCPUX Bias Register (Default: 0x80100000)

Offset: 0x00A0			Register Name: <b>PLL_COCPUX_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x1	VCO_RST. VCO reset in.
30:21	/	/	/
20:16	R/W	0x10	PLL_BIAS_CURRENT. PLL current bias control [4:0], CPU_CP.
15:0	/	/	/

### 3.3.5.29. PLL\_C1CPUX Bias Register (Default: 0x80100000)

Offset: 0x00A4			Register Name: <b>PLL_C1CPUX_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x1	VCO_RST. VCO reset in.
30:21	/	/	/
20:16	R/W	0x10	PLL_BIAS_CURRENT. PLL current bias control [4:0], CPU_CP.
15:0	/	/	/

### 3.3.5.30. PLL\_AUDIO Bias Register (Default: 0x00080000)

Offset: 0x00A8			Register Name: <b>PLL_AUDIO_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.31. PLL\_PERIPH0 Bias Register (Default: 0x00080000)

Offset: 0x00AC			Register Name: <b>PLL_PERIPH0_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/

20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.32. PLL\_VE Bias Register (Default: 0x00080000)

Offset: 0x00B0			Register Name: <b>PLL_VE_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.33. PLL\_DDR Bias Register (Default: 0x00080000)

Offset: 0x00B4			Register Name: <b>PLL_DDR_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.34. PLL\_VIDEO0 Bias Register (Default: 0x00100000)

Offset: 0x00B8			Register Name: <b>PLL_VIDEO0_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x10	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.35. PLL\_VIDEO1 Bias Register (Default: 0x00100000)

Offset: 0x00BC			Register Name: <b>PLL_VIDEO1_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x10	PLL_BIAS_CTRL. PLL bias control [4:0].

15:0	/	/	/
------	---	---	---

### 3.3.5.36. PLL\_GPU Bias Register (Default: 0x00080000)

Offset: 0x00C0			Register Name: <b>PLL_GPU_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.37. PLL\_DE Bias Register (Default: 0x00080000)

Offset: 0x00C4			Register Name: <b>PLL_DE_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.38. PLL\_ISP Bias Register (Default: 0x00080000)

Offset: 0x00C8			Register Name: <b>PLL_ISP_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

### 3.3.5.39. PLL\_PERIPH1 Bias Register (Default: 0x00080000)

Offset: 0x00CC			Register Name: <b>PLL_PERIPH1_BIAS_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x8	PLL_BIAS_CTRL. PLL bias control [4:0].
15:0	/	/	/

**3.3.5.40. PLL\_COCPUX Tuning Register (Default: 0x44404000)**

Offset: 0x00E0			Register Name: PLL_COCPUX_TUN_REG
Bit	R/W	Default/Hex	Description
31	/	/	/
30:28	R/W	0x4	VCO_RNG_CTRL. VCO range control [2:0].
27	/	/	/
36:24	R/W	0x4	KVCO_GAIN_CTRL. KVCO gain control [2:0].
23	/	/	/
22:16	R/W	0x40	CNT_INIT_CTRL. Counter initial control [6:0].
15	R/W	0x0	C_OD0. C-REG-OD0 for verify.
14:8	R/W	0x40	C_B_IN. C-B-IN [6:0] for verify.
7	R/W	0x0	C_OD1. C-REG-OD1 for verify.
6:0	R/W	0x0	C_B_OUT. C-B-OUT [6:0] for verify.

**3.3.5.41. PLL\_C1CPUX Tuning Register (Default: 0x44404000)**

Offset: 0x00E4			Register Name: PLL_C1CPUX_TUN_REG
Bit	R/W	Default/Hex	Description
31	/	/	/
30:28	R/W	0x4	VCO_RNG_CTRL. VCO range control [2:0].
27	/	/	/
26:24	R/W	0x4	KVCO_GAIN_CTRL. KVCO gain control [2:0].
23	/	/	/
22:16	R/W	0x40	CNT_INIT_CTRL. Counter initial control [6:0].
15	R/W	0x0	C_OD0. C-REG-OD0 for verify.
14:8	R/W	0x40	C_B_IN. C-B-IN [6:0] for verify.
7	R/W	0x0	C_OD1. C-REG-OD1 for verify.

6:0	R/W	0x0	C_B_OUT. C-B-OUT [6:0] for verify.
-----	-----	-----	---------------------------------------

### 3.3.5.42. PLL\_AUDIO Pattern Control Register (Default: 0x00000000)

Offset: 0x0108			Register Name: PLL_AUDIO_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.43. PLL\_PERIPH0 Pattern Control Register (Default: 0x00000000)

Offset: 0x010C			Register Name: PLL_PERIPH0_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP.



			Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

#### 3.3.5.44. PLL\_VE Pattern Control Register (Default: 0x00000000)

Offset: 0x0110			Register Name: <b>PLL_VE_PAT_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

#### 3.3.5.45. PLL\_DDR Pattern Control Register (Default: 0x00000000)

Offset: 0x0114			Register Name: <b>PLL_DDR_PAT_CTRL_REG</b>
Bit	R/W	Default/Hex	Description

31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.46. PLL\_VIDEO0 Pattern Control Register (Default: 0x00000000)

Offset: 0x0118			Register Name: PLL_VIDEO0_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz

			11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.47. PLL\_VIDEO1 Pattern Control Register (Default: 0x00000000)

Offset: 0x011C			Register Name: PLL_VIDEO1_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.48. PLL\_GPU Pattern Control Register (Default: 0x00000000)

Offset: 0x0120			Register Name: PLL_GPU_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).

28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.49. PLL\_DE Pattern Control Register (Default: 0x00000000)

Offset: 0x0124			Register Name: <b>PLL_DE_PAT_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.50. PLL\_ISP Pattern Control Register (Default: 0x00000000)

Offset: 0x0128			Register Name: <b>PLL_ISP_PAT_CTRL_REG</b>
----------------	--	--	--

Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.51. PLL\_PERIPH1 Pattern Control Register (Default: 0x00000000)

Offset: 0x012C			Register Name: PLL_PERIPH1_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz. 1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz

			11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.3.5.52. Clock Output A Register (Default: 0x00000000)

Offset: 0x0180			Register Name: <b>CLK_OUTPUT_A_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	CLK_OUTPUT_ENABLE. 0: Disable. 1: Enable. Output A clock = Clock Source /Divide N /Divide M.
30:26	/	/	/
25:24	R/W	0x0	CLK_OUTPUT_SRC_SELECT. 00: OSC24M / 750 (32K). 01: X32KI. 10: OSC24M. 11: /
23:22	/	/	/
21:20	R/W	0x0	CLK_DIV_N. 00: /1 01: /2 10: /4 11: /8
19:13	/	/	/
12:8	R/W	0x0	CLK_DIV_M. 00000: /1 00001: /2 00010: /3 00011: /4 ..... 11111: /32
7:0	/	/	/

### 3.3.5.53. Clock Output B Register (Default: 0x00000000)

Offset: 0x0184			Register Name: <b>CLK_OUTPUT_B_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	CLK_OUTPUT_ENABLE. 0: Disable. 1: Enable.

			Output B clock = Clock Source /Divide N /Divide M.
30:26	/	/	/
25:24	R/W	0x0	CLK_OUTPUT_SRC_SELECT. 00: OSC24M / 750 (32K). 01: X32KI. 10: OSC24M. 11: /
23:22	/	/	/
21:20	R/W	0x0	CLK_DIV_N. 00: /1 01: /2 10: /4 11: /8
19:13	/	/	/
12:8	R/W	0x0	CLK_DIV_M. 00000: /1 00001: /2 00010: /3 00011: /4 ..... 11111: /32
7:0	/	/	/

### 3.3.6. Programming Guidelines

#### 3.3.6.1. PLL

- 1) In practise, other PLLs doesn't support dynamic frequency scaling except for Cluster-related PLLs;
- 2) The PLL division factor N should be no less than 12;
- 3) PLL\_PERIPH0 output is used for AHB, APB, and device; the frequency is 960MHz;
- 4) After the PLL\_DDR frequency changes, the 30-bit of PLL\_DDR Control Register should be written 1 to make it valid;
- 5) The output clock of PLL\_PERIPH1 is mainly use for GTBUS and CCI-400. Because the theoretical frequency of GTBUS is 400MHz, the output clock's frequency of PLL\_PERIPH1 should be from 384MHz to 432Mhz.

#### 3.3.6.2. BUS

- 1) When setting the BUS clock , you should set the division factor first, and after the division factor becomes valid, switch the clock source. The clock source will be switched after at least three clock cycles;
- 2) The BUS clock should not be dynamically changed in most applications.



## 3.4. CCU\_SCLK

### 3.4.1. Overview

The CCU\_SCLK provides registers to control most blocks of the clock division, distribution, synchronization, gating and software reset.

The CCU\_SCLK includes the following features:

- Configuring modules clock
- Bus clock gating
- Bus software reset

### 3.4.2. Typical Applications

If not specifically indicated, the module clocks don't support dynamic frequency scaling, and that means during the module clock switch or division adjustment, glitches or system instability may occur, which may lead to module controller hang.

Make sure that the module rest signal has been released before enabling the module clock gating.

### 3.4.3. Register List

Module Name	Base Address
CCU_SCLK	0x06000400

Register Name	Offset	Description
NAND0_CLK_REG0	0x0000	NAND0 Clock Register 0
NAND0_CLK_REG1	0x0004	NAND0 Clock Register 1
SDMMC0_CLK_REG	0x0010	SDMMC0 Clock Register
SDMMC1_CLK_REG	0x0014	SDMMC1 Clock Register
SDMMC2_CLK_REG	0x0018	SDMMC2 Clock Register
TS_CLK_REG	0x0028	TS Clock Register
SS_CLK_REG	0x002C	SS Clock Register
SPI0_CLK_REG	0x0030	SPI0 Clock Register
SPI1_CLK_REG	0x0034	SPI1 Clock Register
SPI2_CLK_REG	0x0038	SPI2 Clock Register
SPI3_CLK_REG	0x003C	SPI3 Clock Register
DE_CLK_REG	0x0090	DE Clock Register
MP_CLK_REG	0x0098	MP Clock Register
LCD0_CLK_REG	0x009C	LCD0 Clock Register
LCD1_CLK_REG	0x00A0	LCD1 Clock Register
CSI_ISP_CLK_REG	0x00C0	CSI ISP Clock Register
CSI0_MCLK_REG	0x00C4	CSI0 Master Clock Register
CSI1_MCLK_REG	0x00C8	CSI1 Master Clock Register
FD_CLK_REG	0x00CC	FD Clock Register
VE_CLK_REG	0x00D0	VE Clock Register
AVS_CLK_REG	0x00D4	AVS Clock Register
GPU_CORE_CLK_REG	0x00F0	GPU Core Clock Register
GPU_MEM_CLK_REG	0x00F4	GPU Memory Clock Register
GPU_AXI_CLK_REG	0x00F8	GPU AXI Clock Register
GP_ADC_REG	0x010C	GPADC Clock Register
BUS_CLK_GATING_REG0	0x0180	Bus Clock Gating Register 0
BUS_CLK_GATING_REG1	0x0184	Bus Clock Gating Register 1
BUS_CLK_GATING_REG2	0x0188	Bus Clock Gating Register 2
BUS_CLK_GATING_REG3	0x0190	Bus Clock Gating Register 3
BUS_CLK_GATING_REG4	0x0194	Bus Clock Gating Register 4
BUS_SOFT_RST_REG0	0x01A0	Bus Software Reset Register 0
BUS_SOFT_RST_REG1	0x01A4	Bus Software Reset Register 1
BUS_SOFT_RST_REG2	0x01A8	Bus Software Reset Register 2
BUS_SOFT_RST_REG3	0x01B0	Bus Software Reset Register 3
BUS_SOFT_RST_REG4	0x01B4	Bus Software Reset Register 4

### 3.4.4. Register Description

#### 3.4.4.1. NAND0 Clock Register 0 (Default: 0x00000000)

Offset: 0x0000			Register Name: <b>NAND0_CLK_REG0</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.2. NAND0 Clock Register 1 (Default: 0x00000000)

Offset: 0x0004			Register Name: <b>NAND0_CLK_REG1</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M.

			0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2^n. The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.3. SDMMC0 Clock Register (Default: 0x00000000)

Offset: 0x0010			Register Name: <b>SDMMC0_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23	/	/	/
22:20	R/W	0x0	SAMPLE_CLK_PHASE_CTR. Sample Clock Phase Control. The sample clock phase delay is based on the number of source clock that is from 0 to 7.
19:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n). The select clock source is pre-divided by 2^n. The divider is 1/2/4/8.
15:11	/	/	/
10:8	R/W	0x0	OUTPUT_CLK_PHASE_CTR. Output Clock Phase Control. The output clock phase delay is based on the number of source clock that is from 0 to 7.
7:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M.

			Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.
--	--	--	--

#### 3.4.4.4. SDMMC1 Clock Register (Default: 0x00000000)

Offset: 0x0014			Register Name: <b>SDMMC1_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23	/	/	/
22:20	R/W	0x0	SAMPLE_CLK_PHASE_CTR. Sample Clock Phase Control. The sample clock phase delay is based on the number of source clock that is from 0 to 7.
19:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n). The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:11	/	/	/
10:8	R/W	0x0	OUTPUT_CLK_PHASE_CTR. Output Clock Phase Control. The output clock phase delay is based on the number of source clock that is from 0 to 7.
7:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.5. SDMMC2 Clock Register (Default: 0x00000000)

Offset: 0x0018			Register Name: <b>SDMMC2_CLK_REG</b>
----------------	--	--	--------------------------------------

Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23	/	/	/
22:20	R/W	0x0	SAMPLE_CLK_PHASE_CTR. Sample Clock Phase Control. The sample clock phase delay is based on the number of source clock that is from 0 to 7.
19:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n). The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:11	/	/	/
10:8	R/W	0x0	OUTPUT_CLK_PHASE_CTR. Output Clock Phase Control. The output clock phase delay is based on the number of source clock that is from 0 to 7.
7:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.6. TS Clock Register (Default: 0x00000000)

Offset: 0x0028			Register Name: TS_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL.

			Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.7. SS Clock Register (Default: 0x00000000)

Offset: 0x002C			Register Name: <b>SS_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. 1101: PLL_PERIPH1. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.8. SPI0 Clock Register (Default: 0x00000000)

Offset: 0x0030			Register Name: <b>SPI0_CLK_REG</b>
----------------	--	--	------------------------------------



Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.9. SPI1 Clock Register (Default: 0x00000000)

Offset: 0x0034			Register Name: SPI1_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M.

			Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.
--	--	--	---

#### 3.4.4.10. SPI2 Clock Register (Default: 0x00000000)

Offset: 0x0038			Register Name: <b>SPI2_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.11. SPI3 Clock Register (Default: 0x00000000)

Offset: 0x003C			Register Name: <b>SPI3_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0001: PLL_PERIPH0.

			Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.12. DE Clock Register (Default: 0x00000000)

Offset: 0x0090			Register Name: <b>DE_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source PLL_DE/Divider M.
30:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.13. MP Clock Register (Default: 0x0B000000)

Offset: 0x0098			Register Name: <b>MP_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M.
30:28	/	/	/
27:24	R/W	0xB	CLK_SRC_SEL. Clock Source Select. 1001: PLL_VIDEO1. 1010: PLL_GPU. 1011: PLL_DE. Others: /
23:4	/	/	/

3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.
-----	-----	-----	---

#### 3.4.4.14. LCD0 Clock Register (Default: 0x08000000)

Offset: 0x009C			Register Name: LCD0_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M.
30:28	/	/	/
27:24	R/W	0x8	CLK_SRC_SEL. Clock Source Select. 1000: PLL_VIDEO0. 1001: PLL_VIDEO1. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.15. LCD1 Clock Register (Default: 0x09000000)

Offset: 0x00A0			Register Name: LCD1_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M.
30:28	/	/	/
27:24	R/W	0x9	CLK_SRC_SEL. Clock Source Select. 1000: PLL_VIDEO0. 1001: PLL_VIDEO1. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M.

			Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.
--	--	--	---

#### 3.4.4.16. CSI ISP Clock Register (Default: 0x00000000)

Offset: 0x00C0			Register Name: CSI_ISP_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source PLL_ISP/Divider M.
30:17	/	/	/
16	R/W	0x0	CSI_MISC_CLK_GATING. 0: Clock is OFF. 1: Clock is ON. This clock is OSC24M.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.17. CSIO Master Clock Register (Default: 0x00000000)

Offset: 0x00C4			Register Name: CSIO_MCLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	MCLK_GATING. Gating of MCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M. This MCLK is transmitted to relevant pad.
30:28	/	/	/
27:24	R/W	0x9	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 1001: PLL_VIDEO1. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m)

			The pre-divided clock is divided by (m+1). The divider is from 1 to 16.
--	--	--	---

#### 3.4.4.18. CSI1 Master Clock Register (Default: 0x00000000)

Offset: 0x00C8			Register Name: CSI1_MCLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	MCLK_GATING. Gating of MCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M.
30:28	/	/	/
27:24	R/W	0x9	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 1001: PLL_VIDEO1. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.19. FD Clock Register (Default: 0x0C000000)

Offset: 0x00CC			Register Name: FD_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of Face Detect SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider M.
30:28	/	/	/
27:24	R/W	0xC	CLK_SRC_SEL. Clock Source Select. 0001: PLL_PERIPH0. 1100: PLL_ISP. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

**3.4.4.20. VE Clock Register (Default: 0x00000000)**

Offset: 0x00D0			Register Name: <b>VE_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source PLL_VE/Divider N.
30:19	/	/	/
18:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (N). The select clock source is pre-divided by n+1. The divider is from 1 to 8.
15:0	/	/	/

**3.4.4.21. AVS Clock Register (Default: 0x00000000)**

Offset: 0x00D4			Register Name: <b>AVS_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. The clock is OSC24M.
30:0	/	/	/

**3.4.4.22. GPU Core Clock Register (Default: 0x00000000)**

Offset: 0x00F0			Register Name: <b>GPU_CORE_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source PLL_GPU/Divider M.
30:3	/	/	/
2:0	R/W	0x0	CLK_DIV_RATIO_M. Master Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 8.

**3.4.4.23. GPU Memory Clock Register (Default: 0x00000000)**

Offset: 0x00F4			Register Name: <b>GPU_MEM_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source PLL_GPU/Divider M.
30:3	/	/	/
2:0	R/W	0x0	CLK_DIV_RATIO_M. Master Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 8.

**3.4.4.24. GPU AXI Clock Register (Default: 0x01000000)**

Offset: 0x00F8			Register Name: <b>GPU_AXI_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source /Divider M.
30:28	/	/	/
27:24	R/W	0x1	CLK_SRC_SEL. Clock Source Select. 0001: PLL_PERIPH0. 1010: PLL_GPU. Others: /
23:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Master Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

**3.4.4.25. GPADC Clock Register (Default: 0x00000000)**

Offset: 0x010C			Register Name: <b>GPADC_CLK_REG</b>
Bit	R/W	Default/Hex	Description



31	R/W	0x0	SCLK_GATING. Gating of SCLK. 0: Clock is OFF. 1: Clock is ON. SCLK = Clock Source/Divider N/Divider M.
30:28	/	/	/
27:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select. 0000: OSC24M. 0100: PLL_AUDIO. 0111: X32KI. Others: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n) The select clock source is pre-divided by 2^n. The divider is 1/2/4/8.
15:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m) The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

#### 3.4.4.26. Bus Clock Gating Register 0 (Default: 0x00000000)

Offset: 0x0180			Register Name: <b>BUS_CLK_GATING_REG0</b>
Bit	R/W	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	SPI3_GATING. Gating Clock for SPI3. 0: Mask. 1: Pass.
22	R/W	0x0	SPI2_GATING. Gating Clock for SPI2. 0: Mask. 1: Pass.
21	R/W	0x0	SPI1_GATING. Gating Clock for SPI1. 0: Mask. 1: Pass.
20	R/W	0x0	SPIO_GATING. Gating Clock for SPIO. 0: Mask. 1: Pass.

19	/	/	/
18	R/W	0x0	TS_GATING. Gating Clock for TS. 0: Mask. 1: Pass.
17	/	/	/
16	R/W	0x0	/
15	R/W	0x0	/
14	R/W	0x0	/
13	R/W	0x0	NDFC0_GATING. Gating Clock for NDFC0. 0: Mask. 1: Pass.
12	R/W	0x0	NDFC1_GATING. Gating Clock for NDFC1. 0: Mask. 1: Pass.
11:9	/	/	/
8	R/W	0x0	SD_GATING. Gating Clock for SD. 0: Mask. 1: Pass.
7:6	/	/	/
5	R/W	0x0	SS_GATING. Gating Clock for SS. 0: Mask. 1: Pass.
4	/	/	/
3	R/W	0x0	GPU_CTRL_GATING. Gating Clock for GPU_CTRL 0: Mask. 1: Pass.
2	/	/	/
1	R/W	0x0	/
0	R/W	0x0	FD_GATING. Gating Clock for FD. 0: Mask. 1: Pass.

#### 3.4.4.27. Bus Clock Gating Register 1 (Default: 0x00000000)

Offset: 0x0184	Register Name: <b>BUS_CLK_GATING_REG1</b>
----------------	---

Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DMA_GATING. Gating Clock for DMA. 0: Mask. 1: Pass.
23	R/W	0x0	HSTIMER_GATING. Gating Clock for HSTIMER. 0: Mask. 1: Pass.
22	R/W	0x0	SPINLOCK_GATING. Gating Clock for SPINLOCK. 0: Mask. 1: Pass.
21	R/W	0x0	MSGBOX_GATING. Gating Clock for MSGBOX. 0: Mask. 1: Pass.
20:18	/	/	/
17	R/W	0x0	/
16:2	/	/	/
1	R/W	0x0	USB_HOST_GATING. Gating Clock for USB HOST. 0: Mask. 1: Pass.
0	R/W	0x0	/

#### 3.4.4.28. Bus Clock Gating Register 2 (Default: 0x00000000)

Offset: 0x0188			Register Name: <b>BUS_CLK_GATING_REG2</b>
Bit	R/W	Default/Hex	Description
31:12	/	/	/
11	R/W	0x0	/
10:9	/	/	/
8	R/W	0x0	MP_GATING. Gating Clock for MP. 0: Mask. 1: Pass.
7	R/W	0x0	DE_GATING. Gating Clock for DE. 0: Mask. 1: Pass.

6	/	/	/
5	R/W	0x0	/
4	R/W	0x0	CSI_GATING. Gating Clock for CSI. 0: Mask. 1: Pass.
3	/	/	/
2	R/W	0x0	/
1	R/W	0x0	LCD1_GATING. Gating Clock for LCD1. 0: Mask. 1: Pass.
0	R/W	0x0	LCD0_GATING. Gating Clock for LCD0. 0: Mask. 1: Pass.

#### 3.4.4.29. Bus Clock Gating Register 3 (Default: 0x00040000)

Offset: 0x0190			Register Name: <b>BUS_CLK_GATING_REG3</b>
Bit	R/W	Default/Hex	Description
31:19	/	/	/
18	/	/	/
17	R/W	0x0	GPADC_GATING. Gating Clock for GPADC. 0: Mask. 1: Pass.
16	/	/	/
15	R/W	0x0	KEYADC_GATING. Gating Clock for KEYADC. 0: Mask. 1: Pass.
14:6	/	/	/
5	R/W	0x0	GPIO_GATING. Gating Clock for GPIO. 0: Mask. 1: Pass.
4:0	/	/	/

**3.4.4.30. Bus Clock Gating Register 4 (Default: 0x00000000)**

Offset: 0x0194			Register Name: <b>BUS_CLK_GATING_REG4</b>
Bit	R/W	Default/Hex	Description
31:22	/	/	/
21	R/W	0x0	UART5_GATING. Gating Clock for UART5. 0: Mask. 1: Pass.
20	R/W	0x0	UART4_GATING. Gating Clock for UART4. 0: Mask. 1: Pass.
19	R/W	0x0	UART3_GATING. Gating Clock for UART3. 0: Mask. 1: Pass.
18	R/W	0x0	UART2_GATING. Gating Clock for UART2. 0: Mask. 1: Pass.
17	R/W	0x0	UART1_GATING. Gating Clock for UART1. 0: Mask. 1: Pass.
16	R/W	0x0	UART0_GATING. Gating Clock for UART0. 0: Mask. 1: Pass.
15:5	/	/	/
4	R/W	0x0	TWI4_GATING. Gating Clock for TWI4. 0: Mask. 1: Pass.
3	R/W	0x0	TWI3_GATING. Gating Clock for TWI3. 0: Mask. 1: Pass.
2	R/W	0x0	TWI2_GATING. Gating Clock for TWI2. 0: Mask. 1: Pass.
1	R/W	0x0	TWI1_GATING.

			Gating Clock for TWI1. 0: Mask. 1: Pass.
0	R/W	0x0	TWIO_GATING. Gating Clock for TWIO. 0: Mask. 1: Pass.

### 3.4.4.31. Bus Software Reset Register 0 (Default: 0x00000000)

Offset: 0x01A0			Register Name: <b>BUS_SOFT_RST_REG0</b>
Bit	R/W	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	SPI3_RESET. Reset signal for SPI3. 0: Assert. 1: De-assert.
22	R/W	0x0	SPI2_RESET. Reset signal for SPI2. 0: Assert. 1: De-assert.
21	R/W	0x0	SPI1_RESET. Reset signal for SPI1. 0: Assert. 1: De-assert.
20	R/W	0x0	SPIO_RESET. Reset signal for SPIO. 0: Assert. 1: De-assert.
19	/	/	/
18	R/W	0x0	TS_RESET. Reset signal for TS. 0: Assert. 1: De-assert.
17:15	/	/	/
14	R/W	0x0	/
13	R/W	0x0	NDFC0_RESET. Reset signal for NDFC0. 0: Assert. 1: De-assert.
12	R/W	0x0	NDFC1_RESET. Reset signal for NDFC1.

			0: Assert. 1: De-assert.
11:9	/	/	/
8	R/W	0x0	SD_RESET. Reset signal for SDMMC. 0: Assert. 1: De-assert.
7:6	/	/	/
5	R/W	0x0	SS_RESET. Reset signal for SS. 0: Assert. 1: De-assert.
4	/	/	/
3	R/W	0x0	GPU_CTRL_RESET. Reset signal for GPU_CTRL. 0: Assert. 1: De-assert.
2:1	/	/	/
0	R/W	0x0	FD_RESET. Reset signal for FD. 0: Assert. 1: De-assert.

#### 3.4.4.32. Bus Software Reset Register 1 (Default: 0x00000000)

Offset: 0x01A4			Register Name: <b>BUS_SOFT_RST_REG1</b>
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DMA_RESET. Reset signal for DMA. 0: Assert. 1: De-assert.
23	R/W	0x0	HSTIMER_RESET. Reset signal for HSTIMER. 0: Assert. 1: De-assert.
22	R/W	0x0	SPINLOCK_RESET. Reset signal for SPINLOCK. 0: Assert. 1: De-assert.
21	R/W	0x0	MSGBOX_RESET. Reset signal for MSGBOX.

			0: Assert. 1: De-assert.
20:2	/	/	/
1	R/W	0x0	USBDRD_PHY_RESET. Reset signal for USBDRD PHY. 0: Assert. 1: De-assert.
0	R/W	0x0	/

### 3.4.4.33. Bus Software Reset Register 2 (Default: 0x00000000)

Offset: 0x01A8			Register Name: <b>BUS_SOFT_RST_REG2</b>
Bit	R/W	Default/Hex	Description
31:10	/	/	/
9	R/W	0x0	GPU_RESET. Reset signal for GPU. 0: Assert. 1: De-assert.
8	R/W	0x0	MP_RESET. Reset signal for MP. 0: Assert. 1: De-assert.
7	R/W	0x0	DE_RESET. Reset signal for DE. 0: Assert. 1: De-assert.
6:5	R/W	0x0	/
4	R/W	0x0	CSI_RESET. Reset signal for CSI. 0: Assert. 1: De-assert.
3:2	R/W	0x0	/
1	R/W	0x0	LCD1_RESET. Reset signal for LCD1. 0: Assert. 1: De-assert.
0	R/W	0x0	LCD0_RESET. Reset signal for LCD0. 0: Assert. 1: De-assert.



#### 3.4.4.34. Bus Software Reset Register 3 (Default: 0x00000000)

Offset: 0x01B0			Register Name: <b>BUS_SOFT_RST_REG3</b>
Bit	R/W	Default/Hex	Description
31:19	/	/	/
18	/	/	/
17	R/W	0x0	GPADC_RESET. Reset signal for GPADC. 0: Assert. 1: De-assert.
16	/	/	/
15	R/W	0x0	KEYADC_RESET. Reset signal for KEYADC. 0: Assert. 1: De-assert.
14:0	/	/	/

#### 3.4.4.35. Bus Software Reset Register 4 (Default: 0x00000000)

Offset: 0x01B4			Register Name: <b>BUS_SOFT_RST_REG4</b>
Bit	R/W	Default/Hex	Description
31:22	/	/	/
21	R/W	0x0	UART5_RESET. Reset signal for UART5. 0: Assert. 1: De-assert.
20	R/W	0x0	UART4_RESET. Reset signal for UART4. 0: Assert. 1: De-assert.
19	R/W	0x0	UART3_RESET. Reset signal for UART3. 0: Assert. 1: De-assert.
18	R/W	0x0	UART2_RESET. Reset signal for UART2. 0: Assert. 1: De-assert.
17	R/W	0x0	UART1_RESET. Reset signal for UART1.

			0: Assert. 1: De-assert.
16	R/W	0x0	UART0_RESET. Reset signal for UART0. 0: Assert. 1: De-assert.
15:5	/	/	/
4	R/W	0x0	TWI4_RESET. Reset signal for TWI4. 0: Assert. 1: De-assert.
3	R/W	0x0	TWI3_RESET. Reset signal for TWI3. 0: Assert. 1: De-assert.
2	R/W	0x0	TWI2_RESET. Reset signal for TWI2. 0: Assert. 1: De-assert.
1	R/W	0x0	TWI1_RESET. Reset signal for TWI1. 0: Assert. 1: De-assert.
0	R/W	0x0	TWI0_RESET. Reset signal for TWI0. 0: Assert. 1: De-assert.

### **3.4.5. Programming Guidelines**

#### **3.4.5.1. Clock Switch**

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

#### **3.4.5.2. Gating and reset**

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

## 3.5. CPU Configuration

### 3.5.1. Overview

CPUCFG module is used to configure related CPU control of the two clusters, including power, reset, cache, debug, CPU status, etc; It will be used when you want to disable/enable the CPU, cluster switch, CPU status check, and debug, etc.

It features:

- Capable of CPU reset, including core reset, debug circuit rest, etc
- Capable of Cache control, including cache reset, idle control, etc
- Capable of other CPU-related control, including interface control, CP15 control, and power control, etc
- Capable of checking CPU status, including idle status, SMP status, and interrupt status, etc

### 3.5.2. Functionalities Description

#### 3.5.2.1. Module Operation description

##### 3.5.2.1.1. L2 Idle Mode

When the L2 of Cluster0 needs to enter WFI mode, firstly make sure the CPU0/1/2/3 of Cluster 0 all enter WFI mode, which can be checked in Cluster0 CPU status Register, and then pull the ACINACTM of Cluster 0 high by writing related register bit to 1, and then check whether L2 enters idle status by checking whether the STANDBYWFI2 is high. Remember to set the ACINACTM to low when exiting the L2 idle mode;

When the L2 of Cluster 1 needs to enter WFI mode, firstly make sure that the CPU0/1/2/3 of cluster 1 all enter WFI mode. Notice that you need to disable Branch prediction before CPU0/1/2/3 of cluster 1 all enter WFI mode, and by checking Cluster1 CPU status Register to know whether they enter WFI mode, and then pull Cluster 1 ACINACTM to high by setting related register bit to 1, finally, you can check whether the STANDBYWFI2 is high to know whether L2 enters idle status. Remember to set the ACINACTM to low when exiting the L2 idle mode;

##### 3.5.2.1.2. Cache Idle Mode

L1/L2 cache of each cluster have Sleep and ShutDown low power mode control mechanism, which can be realized by

setting related register values;

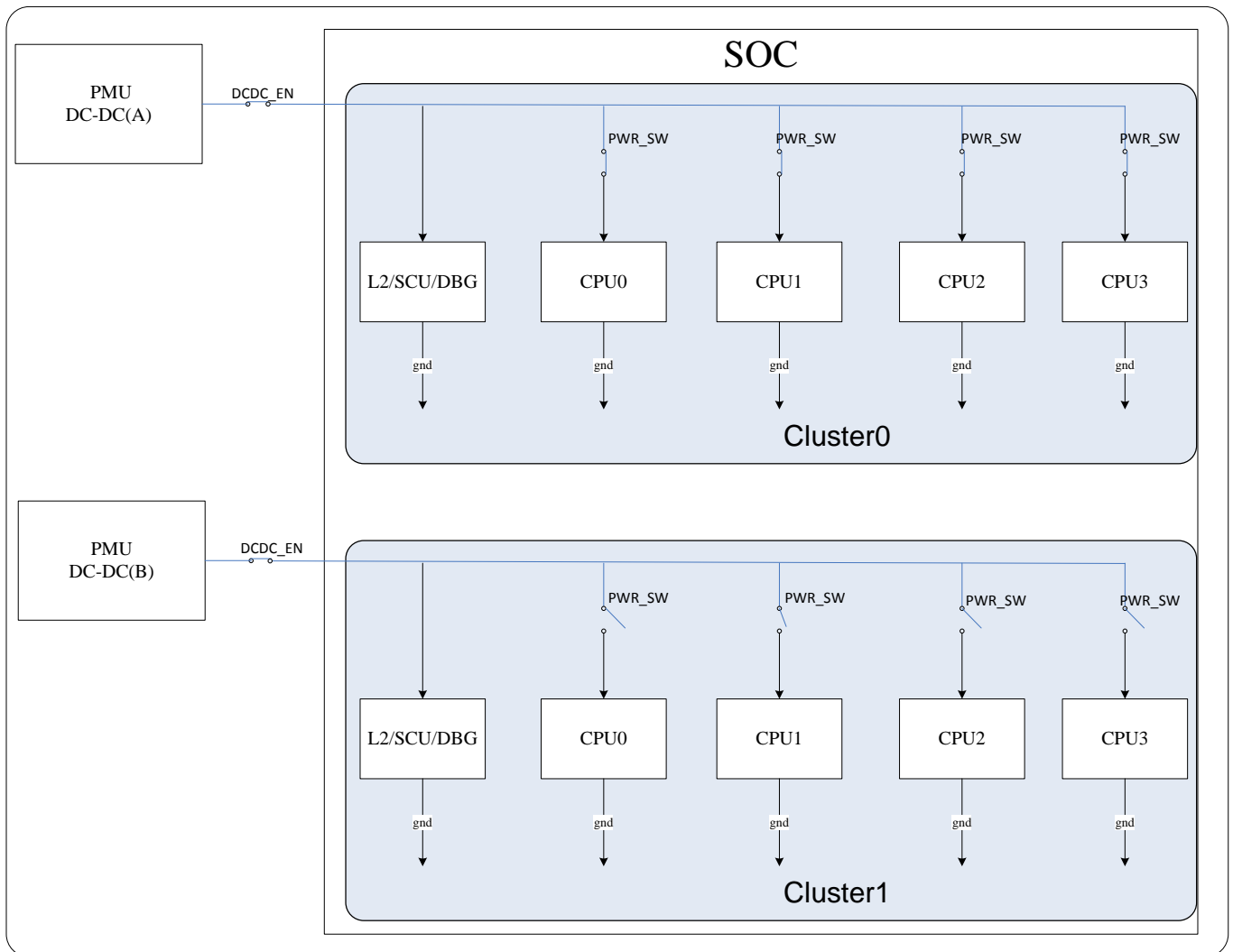
### 3.5.2.1.3. CPU Reset System

The CPU reset includes core reset, power on reset and H\_reset. And their scopes rank: Core Reset < Power-On Reset < H\_Reset

### 3.5.2.1.4. Trace Clock

When the trace clock is to be changed, trace clock in CCU register should be configured, and a software pulse operation is needed as well. The register operation is detailed in General Control Register1.

### 3.5.3. Block Diagram



CPU Power Domain Diagram

The figure above lists the CPU reset power domain. Since each CPU and its appended circuits have the same power domain, the processor and related L1 cache, neon, and vfp should be taken as a whole when it comes to the CPU core enable/disable.

#### **3.5.4. Operation Principle**

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

### 3.5.5. Register List

Module Name	Base Address
CPUCFG	0x01700000

Register Name	Offset	Description
C0_CTRL_REG0	0x0000	Cluster 0 Control Register0
C0_CTRL_REG1	0x0004	Cluster 0 Control Register1
C0_ADB400_PWRDNREQN_REG	0x0008	Cluster 0 adb400 pwrnreqn Register
C1_CTRL_REG0	0x0010	Cluster 1 Control Register0
C1_CTRL_REG1	0x0014	Cluster 1 Control Register1
C1_ADB400_PWRDNREQN_REG	0x0018	Cluster 1 adb400 pwrnreqn Register
GENER_CTRL_REG0	0x0028	General Control Register0
GENER_CTRL_REG1	0x002C	General Control Register1
C0_CPU_STATUS	0x0030	Cluster0 CPU Status Register
C1_CPU_STATUS	0x0034	Cluster1 CPU Status Register
IRQ_FIQ_STATUS	0x003C	Cluster CPU Irq and Fiq Status Register
C0_RST_CTRL	0x0080	Cluster 0 Reset Control Register
C1_RST_CTRL	0x0084	Cluster 1 Reset Control Register
GIC_JTAG_RST_CTRL	0x0088	GIC and Jtag reset control Register

### 3.5.6. Register Description

#### 3.5.6.1. Cluster 0 Control Register0(Default :0x60000000)

Offset: 0x00			Register Name: CO_CTRL_REG0
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SYSBAR_DISABLE.  Disable broadcasting of barriers onto system bus:  0: Barriers are broadcast onto system bus,this requires an AMBA4 interconnect.  1: Barriers are not broadcast onto the system bus.This is compatible with an AXI3 interconnect.
30	R/W	0x1	BROADCAST_INNER.  Enable broadcasting of Inner Shareable transactions:  0: Inner shareable transactions are not broadcasted externally.  1: Inner shareable transactions are broadcasted externally.
29	R/W	0x1	BROADCAST_OUTER.  Enable broadcasting of outer shareable transactions:  0: Outer Shareable transactions are not broadcasted externally.  0: Outer Shareable transactions are broadcasted externally.
28	R/W	0x0	BROADCAST_CACHE_MAINT  Enable broadcasting of cache maintenance operations to downstream caches:  0: Cache maintenance operations are not broadcasted to downstream caches.  1: Cache maintenance operations are broadcasted to downstream caches.
27:12	/	/	/
11:8	R/W	0x0	CP15S_DISABLE.  Disable write access to some secure CP15 register.
7:6	/	/	/



5	R/W	0x0	CFGS_DISABLE.  Disable write access to some secure GIC register.
4	R/W	0x0	L2_RST_DISABLE.  Disable automatic L2 cache invalidate at reset:  0: L2 cache is reset by hardware.  1: L2 cache is not reset by hardware.
3:0	R/W	0x0	L1_RST_DISABLE.  Disable automatic Cluster0 CPU0/1/2/3 L1 cache invalidate at reset:  0: L1 cache is reset by hardware.  1: L1 cache is not reset by hardware.

### 3.5.6.2. Cluster 0 Control Register1(Default :0x00000000)

Offset: 0x04			Register Name: CO_CTRL_REG1
Bit	R/W	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	SLEEP.  Cluster 0 CPU Memory Sleep Mode Control  0: Normal  1: Sleeping.
15:9	/	/	/
8	R/W	0x0	SHUT_DOWN.  Cluster 0 CPU Memory(All L1 and L2 cache) Shut Down Control (The data in cache will be lost)  0:Normal  1:Shut Down.
7:1	/	/	/
0	R/W	0x0	ACINACTM.  Snoop interface is inactive and no longer accepting requests.

**3.5.6.3. Cluster 0 adb400 pwrnreqn Register(Default :0x00000001)**

Offset: 0x08			Register Name: CO_ADB400_PWRDNREQN_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	ADB400_PWRDNREQN.

**3.5.6.4. Cluster 1 Control Register0(Default :0x70000000)**

Offset: 0x10			Register Name: C1_CTRL_REG0
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SYSBAR_DISABLE.  Disable broadcasting of barriers onto system bus:  0: Barriers are broadcast onto system bus,this requires an AMBA4 interconnect.  1: Barriers are not broadcast onto the system bus.This is compatible with an AXI3 interconnect.
30	R/W	0x1	BROADCAST_INNER.  Enable broadcasting of Inner Shareable transactions:  0: Inner shareable transactions are not broadcasted externally.  1: Inner shareable transactions are broadcasted externally.
29	R/W	0x1	BROADCAST_OUTER.  Enable broadcasting of outer shareable transactions:  0: Outer Shareable transactions are not broadcasted externally.  1: Outer Shareable transactions are broadcasted externally.
28	R/W	0x1	BROADCAST_CACHE_MAINT  Enable broadcasting of cache maintenance operations to downstream caches:  0: Cache maintenance operations are not broadcasted to downstream caches.  1: Cache maintenance operations are broadcasted to downstream caches.

27:17	/	/	/
16	R/W	0x0	IMINLN.  Individual processor control of the instruction cache minimum line size at reset. It sets the initial value of the IminLine Field in the CP15 Cache Type Register (CTR):  0: 32-bytes.  1: 64-bytes.
15:12	/	/	/
11:8	R/W	0x0	CP15S_DISABLE.  Disable write access to some secure CP15 register.
7:1	/	/	/
0	R/W	0x0	L2_RST_DISABLE.  L2 cache hardware reset disable:  0: L2 cache is reset by hardware.  1: L2 cache is not reset by hardware.

### 3.5.6.5. Cluster 1 Control Register1(Default :0x80000000)

Offset: 0x14			Register Name: C1_CTRL_REG1
Bit	R/W	Default/Hex	Description
31	R/W	0x1	CPU_CLK_EN.  0: Disable  1: Enable
30:20	/	/	/
19:16	R/W	0x0	CPU_CLK_OFF.  Individual processor clock disable.  0: ON  1: OFF
15:10	/	/	/
9	R/W	0x0	SLEEP.

			Cluster 1 CPU Memory Sleep Mode Control  0: Normal  1: Sleeping.
8	R/W	0x0	SHUT_DOWN.  Cluster 1 CPU Memory Shut Down Control  0: Normal  1: Shut Down.
7:1	/	/	/
0	R/W	0x0	ACINACTM.  Snoop interface is inactive and no longer accepting requests.

### 3.5.6.6. Cluster 1 adb400 pwrdrnreqn Register(Default :0x00000001)

Offset: 0x18			Register Name: C1_ADB400_PWRDNREQN_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	ADB400_PWRDNREQN.

### 3.5.6.7. General Control Register0(Default :0x00000000)

Offset: 0x28			Register Name: GENER_CTRL_REG0
Bit	R/W	Default/Hex	Description
31	R/W	0x0	CSETB_SD.  CSETB Shut Down.
30	R/W	0x0	CSETB_SLP.  CSETB Sleep.
29:1	/	/	/
0	R/W	0x0	GIC_CFGSDISABLE.  Disables write access to some secure GIC registers.

**3.5.6.8. General Control Register1(Default :0x00000000)**

Offset: 0x2C			Register Name: GENER_CTRL_REG1
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	TSCLKCHANGE  When trace clock changed, this bit should set 1 then clear 0 to valid the trace clock.

**3.5.6.9. Cluster0 CPU Status Register(Default : 0x00000000)**

Offset: 0x30			Register Name: CO_CPU_STATUS
Bit	R/W	Default/Hex	Description
31:28	/	/	/
27:24	R	0x0	SMP_AMP  A CPU is in Symmetric Multiprocessing mode or Asymmetric Multiprocessing mode.  0: AMP mode  1: SMP mode
23:20	/	/	/
19:16	R	0x0	STANDBYWFI.  Indicates if Cluster0 CPU0/1/2/3 is in WFI standby mode:  0: Processor not in WFI standby mode.  1: Processor in WFI standby mode
15:12	/	/	/
11:8	R	0x0	STANDBYWFE.  Indicates if Cluster0 CPU0/1/2/3 is in the WFE standby mode:  0: Processor not in WFE standby mode  1: Processor in WFE standby mode
7:1	/	/	/
0	R	0x0	STANDBYWFI2.

			Indicates if the Cluster0 L2 memory system is in WFI standby mode.  0:active  1:idle
--	--	--	--

### 3.5.6.10. Cluster1 CPU Status Register(Default : 0x00000000)

Offset: 0x34			Register Name: C1_CPU_STATUS
Bit	R/W	Default/Hex	Description
31:20	/	/	/
19:16	R	0x0	STANDBYWFI.  Indicates if Cluster1 CPU0/1/2/3 is in WFI standby mode:  0: Processor not in WFI standby mode.  1: Processor in WFI standby mode  Note: The CPU need to disable Branch prediction before enter idle mode
15:12	/	/	/
11:8	R	0x0	STANDBYWFE.  Indicates if Cluster1 CPU0/1/2/3 is in the WFE standby mode:  0: Processor not in WFE standby mode  1: Processor in WFE standby mode  Note: The CPU need to disable Branch prediction before enter idle mode
7:1	/	/	/
0	R	0x0	STANDBYWFIL2.  Indicates if the Cluster1 L2 memory system is in WFI standby mode.  0:active  1:idle

### 3.5.6.11. Cluster CPU IRQ and FIQ Status Register(Default : 0x00000000)

Offset: 0x3C		Register Name: IRQ_FIQ_STATUS
--------------	--	-------------------------------

Bit	R/W	Default/Hex	Description
31:16	/	/	/
15:12	R	0x0	C1_FIQ_OUT[3:0].  Cluster1 CPU FIQ wakeup output  0: Normal  1: FIQ happen
11:8	R	0x0	C0_FIQ_OUT[3:0].  Cluster0 CPU FIQ wakeup output  0: Normal  1: FIQ happen
7:4	R	0x0	C1_IRQ_OUT[3:0].  Cluster1 CPU IRQ wakeup output  0: Normal  1: IRQ happen
3:0	R	0x0	C0_IRQ_OUT[3:0].  Cluster0 CPU IRQ wakeup output  0: Normal  1: IRQ happen

### 3.5.6.12. Cluster 0 Reset Control Register(Default: 0x01FF1101)

Offset: 0x80			Register Name: CO_RST_CTRL
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R/W	0x1	SOC_DBG_RST.  Cluster0 SOC Debug Reset  0: assert  1: de-assert.

23:20	R/W	0xF	ETM_RST.  Cluster0 ETM Reset Assert.  0: assert  1: de-assert.
19:16	R/W	0xF	DBG_RST.  Cluster0 Debug Reset Assert.  0: assert  1: de-assert.
15:13	/	/	/
12	R/W	0x1	HRESET.  Cluster0 H_Reset. Reset all the Cluster0 Logic and Cluster0 Interface Logic.  0: assert  1: de-assert.
11:9	/	/	/
8	R/W	0x1	L2_RST.  Cluster0 L2 Cache Reset  0: assert  1: de-assert.
7:4	/	/	/
3:0	R/W	0x1	CORE_RESET.  Cluster0 CPU0/1/2/3 Reset Assert.  0: assert  1: de-assert.

### 3.5.6.13. Cluster 1 Reset Control Register(Default: 0x010F11F0)

Offset: 0x84			Register Name: C1_RST_CTRL
Bit	R/W	Default/Hex	Description
31:25	/	/	/



24	R/W	0x1	DBG_PRST.  Cluster1 Debug P Reset. Just reset the D-APB,CTI/CTM(PCLKDBG)  0: assert  1: de-assert.
23:20	/	/	/
19:16	R/W	0xF	DBG_RST.  Cluster1 Debug Reset Assert.  0: assert  1: de-assert.
15:13	/	/	/
12	R/W	0x1	HRESET.  Cluster1 H_Reset. Reset all the Cluster1 Logic and Cluster1 Interface Logic.  0: assert  1: de-assert.
11:9	/	/	/
8	R/W	0x1	L2_RST.  Cluster1 L2 Cache Reset  0: assert  1: de-assert.
7:4	R/W	0xF	CX_RESET.  Cluster1 CX Reset Assert.(Neon Reset)  0: assert  1: de-assert.
3:0	R/W	0x0	CORE_RESET.  Cluster1 CPU Reset Assert.  0: assert  1: de-assert.

**3.5.6.14. GIC and Jtag Reset Control Register(Default: 0x00000F01)**

Offset: 0x88			Register Name: GIC_JTAG_RST_CTRL
Bit	R/W	Default/Hex	Description
31:12	/	/	/
11	R/W	0x1	CS_RST  CoreSight Reset.  0: assert  1: de-assert.
10	R/W	0x1	DAP_RST  DAP Reset.  0: assert  1: de-assert.
9	R/W	0x1	PORTRST  Jtag portrst.  0: assert  1: de-assert.
8	R/W	0x1	TRST.  Jtag trst.  0: assert  1: de-assert.
7:1	/	/	/
0	R/W	0x1	GIC_RESET.  Gic_reset_cpu_reg  0: assert  1: de-assert.

## 3.6. TimeStamp

### 3.6.1. Overview

The timestamp module generates and distributes a consistent timestamp value for multiple processors and other SOC IP .

It features:

The Timestamp uses a timing reference with a fixed frequency clock OSC24M.

The Timestamp functions as the clock source of the CPU local timer only.

### 3.6.2. Register List

Module Name	Base Address
TIMESTAMP_STA	0x01710000

Note: Timestamp status Register

Register Name	Offset	Description
CNT_LOW_REG	0x0000	Counter low register[31:0] value
CNT_HI_REG	0x0004	Counter high register[63:32] value

Module Name	Base Address
TIMESTAMP_CTRL	0x01720000

Note: Timestamp Control Registers

Register Name	Offset	Description
TSTAMP_CTRL_REG	0x0000	Timestamp control register
CNT_LOW_REG	0x0008	Counter low register[31:0] value
CNT_HI_REG	0x000C	Counter high register[63:32] value
CNT_FREQID_REG	0x0020	The Counter Base Frequency ID Register

### 3.6.3. Timestamp Status Register Description

#### 3.6.3.1. Counter Low Register (Default: 0x00000000)

Offset: 0x0000			Register Name: CNT_LOW_REG
Bit	R/W	Default/Hex	Description
31:0	R	0x0	CNT_LOW_REG. Current value of counter[31:0]

#### 3.6.3.2. Counter High Register (Default: 0x00000000)

Offset: 0x0004			Register Name: CNT_HI_REG
Bit	R/W	Default/Hex	Description
31:0	R	0x0	CNT_HI_REG. Current value of counter[64:32]

### 3.6.4. Timestamp Control Register Description

#### 3.6.4.1. Timestamp Control Register (Default: 0x00000000)

Offset: 0x0000			Register Name: TSTAMP_CTRL_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/. Timestamp Counter Enable. 0: Disable 1: Enable
0	R/W	0x0	

#### 3.6.4.2. Counter Low Register (Default: 0x00000000)

Offset: 0x0008			Register Name: CNT_LOW_REG
Bit	R/W	Default/Hex	Description
31:0	R	0x0	CNT_LOW_REG. Current value of counter[31:0]

#### 3.6.4.3. Counter High Register (Default: 0x00000000)

Offset: 0x000C			Register Name: CNT_HI_REG
Bit	R/W	Default/Hex	Description
31:0	R	0x0	CNT_HI_REG. Current value of counter[64:32]

#### 3.6.4.4. Cluster 0 Control Register1(Default :0x00000000)

Offset: 0x0020			Register Name: CNT_FREQID_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	FREQ.

			This register must be programmed to match the clock frequency of the timestamp generator, in ticks per second. For example, in our soc for a 24 MHz clock, must program 0x16E3600.
--	--	--	--

Note: This register just records the fixed frequency value.

### 3.6.5. Programming Guidelines

- 1) since timestamp counter is used to provide time baseline for the local timer of processor, so the counter should be enabled before the OS runs.
- 2) Timestamp Counter Status Registers(TIMESTAMP STA) are for read only, indicating the current 64-bit counter value;
- 3) Timestamp Counter Control Registers(TIMESTAMP CTRL) is used to enable/disable the counter; Base Frequency ID Register should define the timestamp generator frequency so that other software interface can access the timestamp clock source information through this register.
- 4) Timestamp Counter Control Registers(TIMESTAMP CTRL) is for R/W, indicating the current 64-bit counter value. Writing to this register is not recommended, however, if writing is required, you must follow steps below:

- Disable the counter;
- The first write access must be to the lower 32bits;(Counter Low Register)
- The second write access can be the high 32bits;(Counter High Register)
- The counter initial value then will be updated.



## 3.7. System Control

### 3.7.1. Overview

Area	Address	Size(Bytes)
A1	0x00010000--0x00019FFF	40K
A2	0x08100000--0x08127FFF	160K
B(Secure RAM)	0x00020000--0x0005FFFF	256K
C0_CPUX I-Cache		32K (X=0,1,2,3)
C0_CPUX D-Cache		32K (X=0,1,2,3)
Cluster 0 L2 Cache		512K
C1_CPUX I-Cache		32K (X=0,1,2,3)
C1_CPUX D-Cache		32K (X=0,1,2,3)
Cluster 1 L2 Cache		2048K
Total		3528K

### 3.7.2. System Control Register List

Module Name	Base Address
System Control	0x00800000

Register Name	Offset	Description
VER_REG	0x24	Version Register
EMAC_CLK_REG	0x30	EMAC Clock Register
DISP_MUX_CTRL_REG	0x38	Display Mux Control Register

### 3.7.3. System Control Register Description

#### 3.7.3.1. Version Register

Offset:0x24			Register Name: VER_REG
Bit	R/W	Default/Hex	Description
31:11	/	/	/
10	R	x	UBOOT_SEL_PAD_STA. U_boot Select Pin Status. 0: U_Boot, 1: R_Boot.
9:8	R	x	BOOT_SEL_PAD_STA. Boot Select Pin Status 00: SPI0 Boot 01: eMMC2 Boot 10: SDC2 Boot 11: NAND Flash Boot.
7:0	R	0x0	VER_BITS. This read-only bit field always reads back the mask revision level of the chip.

Note:

The UBOOT\_SEL pin is used for control whether system should jump to USB boot, and the pin is pull up by internal 50K resistor in R\_state. Two pins of BOOT\_SEL are used for selecting which media should be booting firstly, and they are also pull up by internal 50K resistor.

#### 3.7.3.2. EMAC Clock Register (Default: 0x00000000)

Offset:0x30			Register Name: EMAC_CLK_REG
Bit	R/W	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	TXC_DIV_CFG. Clock pre-divide ratio. External transmit clock (125MHz) is pre-divided by as follows for RGMII. 0: /1, generate 125MHz, 1: /5, generate 25MHz.
14:13	/	/	/
12:10	R/W	0x0	ETXDC. Configure EMAC Transmit Clock Delay Chain.
9:5	R/W	0x0	ERXDC. Configure EMAC Receive Clock Delay Chain.
4	R/W	0x0	ERXIE

			Enable EMAC Receive Clock Invertor. 0: Disable, 1: Enable.
3	R/W	0x0	ETXIE Enable EMAC Transmit Clock Invertor. 0: Disable, 1: Enable.
2	R/W	0x0	EPIT EMAC PHY Interface Type 0: GMII/MII, 1: RGMII.
1:0	R/W	0x0	ETCS. EMAC Transmit Clock Source 00: Transmit clock source for MII 01: External transmit clock source for GMII and RGMII 10: Internal transmit clock source for GMII and RGMII 11: Reserved.

### 3.7.3.3. Display Mux Control Register (Default: 0x00000000)

Offset:0x38			Register Name: DISP_MUX_CTRL_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	DISP_MUX_CTRL. Display Mux Control.

## 3.8. Timer

### 3.8.1. Overview

Timer 0/1/2/3 can take their inputs from 32K or OSC24M. They provide the operating system's scheduler interrupt. It is designed to offer maximum accuracy and efficient management, even for systems with long or short response time. They provide 32-bit programmable down-counter and work in auto-reload mode or no-reload mode. When the current value in Current Value Register is counting down to zero, the timer will generate interrupt if set interrupt enable bit.

Timer 4 and Timer 5 have one more clock source named External Clock. They not only can count using OSC24M or 32K like other Timers, but also can be used for calculating external devices' frequency.

The watchdog is used to resume the controller operation when it has been disturbed by malfunctions such as noise and system errors. It features a down counter that allows a watchdog period of up to 16 seconds (512000 cycles). It can generate a general reset or request.

AVS counter is used to synchronize video and audio in the player.

The Timer module includes the following features:

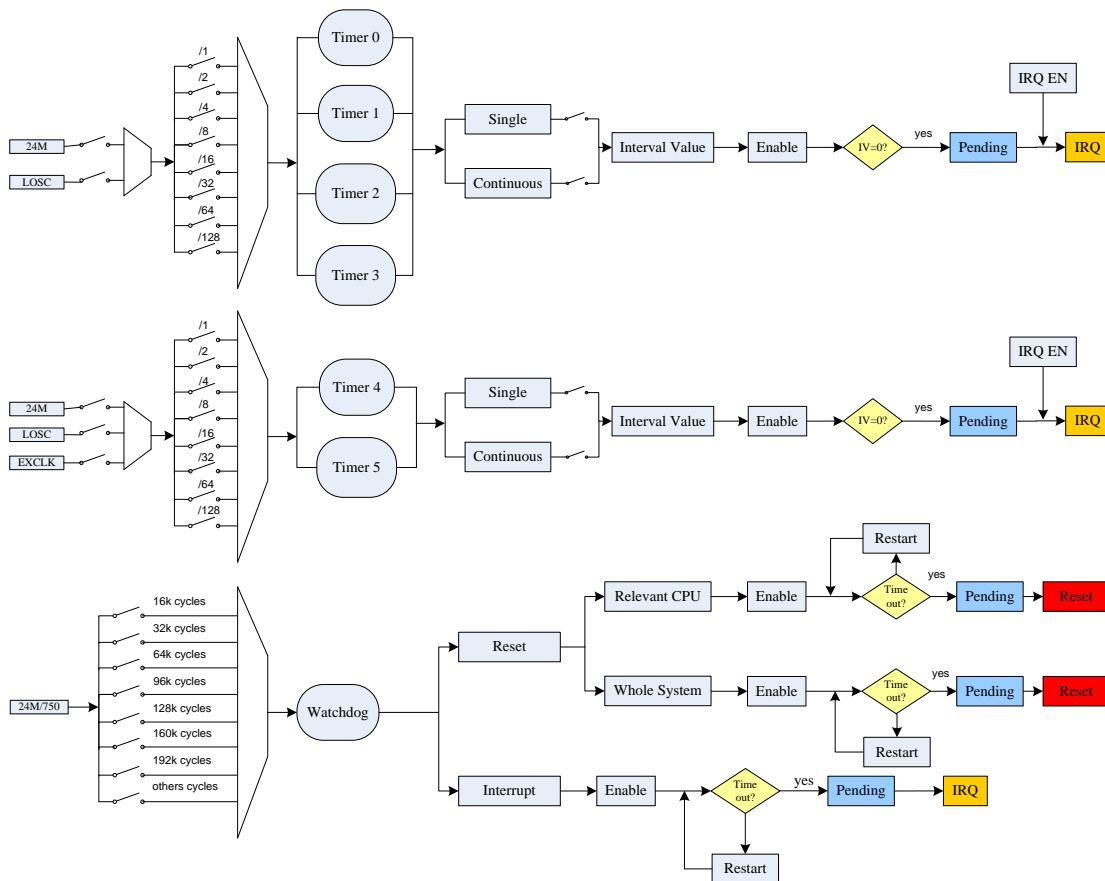
- 6 Timers for system scheduler counting using 24MHz or 32KHz clock, Timer 4/5 also calculates External Clock especially
- Each Timer could generate individual interrupt
- 1 Watchdog for resetting whole system or interrupt
- 2 AVS counters used for synchronizing video and audio in the player

### 3.8.2. Functionalities Description

#### 3.8.2.1. Typical Applications

- Timer provides scheduler interrupt and delay or calculates external devices' frequency
- The watchdog supports reset function for system operation' malfunction
- AVS is used to synchronize video and audio in the player

#### 3.8.2.2. Functional Block Diagram



Timer function structure and work flow

Timer counter' clock input comes from one of the three clock sources that could be pre-scaled up to 128 division. In single mode, when current value is counted down to 0, enable bit would be cleared automatically and Timer stops working. But in continuous mode, Interval Value will be auto-reloaded into *Current Value Register* and then counter counts from the new interval value again when current value is counted down to 0. Every time current value is counted down to 0, a pending will be generated. Pending could be sent to GIC or R\_NVIC only if IRQ enable bit is set.

For Timer4 or Timer5' calculating External Clock, the *Current Value Register* is an up-counter counting form 0. In single

mode, Timer stops working when counter overflows. In continuous mode, counter works from 0 when it overflows. The counter's overflow generates a pending every time. The reload operation makes the counter count from 0 again before it overflows.

Generally watchdog could not count down to 0 because it would be restart inside Interval Value. Otherwise the malfunction makes the watchdog counts down to 0 and a pending will be generated, which causes a reset(*Watchdog Configuration Register* is configured to To whole system) or an interrupt(*Watchdog Configuration Register* is configured to Only interrupt).

AVS has two counters which are both up-counted. The counter's clock source comes from 24MHz/Divisor\_N and Divisor\_N is set in *AVS Counter Divisor Register*. AVS counter could be changed to pause or enable at any time, so are the Interval Value set in AVS Counter 0 Register or AVS Counter 1 Register and Divisor Value set in AVS Counter Divisor Register. When you enable the AVS counter, it counts up from Interval Value until you pause it. It doesn't generate any pending.

### 3.8.2.3. Operation Principle

#### ➤ Timer reload and enable bit

Generally the operation of setting both reload bit and enable bit 1 and writing them into timer control register moreover could cause a risk. It had better to enable Timer after Interval Value has been loaded into Current Value Register totally. Only in timer pause time, when you hope that counter starts working from a new interval value, reload bit and enable bit should be set 1 and written into timer control register at the same time.

#### ➤ Timing requirement for Timer command

For reload and enable operation of Timer, it is necessary to wait some cycles between the same continuous operations. It indicates that from pause state to start or from start state to pause it has to wait for 2 cycles at least. And to reload operation, it could not be implemented immediately again until the reload bit is cleared automatically of the last operation.

#### ➤ Watchdog restart

Watchdog restart function should be enabled inside Interval Value. Writing watchdog restart bit 1 and watchdog key field 0xA57 at the same time make a restart, but writing watchdog key field other values will be ignored.

### 3.8.3. Timer Register List

Module Name	Base Address
TIMER	0x06000C00

Register Name	Offset	Description
TMR_IRQ_EN_REG	0x00	Timer IRQ Enable Register
TMR_IRQ_STA_REG	0x04	Timer Status Register
TMR0_CTRL_REG	0x10	Timer 0 Control
TMR0_INTV_VALUE_REG	0x14	Timer 0 Interval Value Register
TMR0_CUR_VALUE_REG	0x18	Timer 0 Current Value Register
TMR1_CTRL_REG	0x20	Timer 1 Control Register
TMR1_INTV_VALUE_REG	0x24	Timer 1 Interval Value Register
TMR1_CUR_VALUE_REG	0x28	Timer 1 Current Value Register
TMR2_CTRL_REG	0x30	Timer 2 Control Register
TMR2_INTV_VALUE_REG	0x34	Timer 2 Interval Value Register
TMR2_CUR_VALUE_REG	0x38	Timer 2 Current Value Register
TMR3_CTRL_REG	0x40	Timer 3 Control Register
TMR3_INTV_VALUE_REG	0x44	Timer 3 Interval Value Register
TMR3_CUR_VALUE_REG	0x48	Timer 3 Current Value Register
TMR4_CTRL_REG	0x50	Timer 4 Control Register
TMR4_INTV_VALUE_REG	0x54	Timer 4 Interval Value Register
TMR4_CUR_VALUE_REG	0x58	Timer 4 Current Value Register
TMR5_CTRL_REG	0x60	Timer 5 Control Register
TMR5_INTV_VALUE_REG	0x64	Timer 5 Interval Value Register
TMR5_CUR_VALUE_REG	0x68	Timer 5 Current Value Register
AVS_CNT_CTL_REG	0x80	AVS Control Register
AVS_CNT0_REG	0x84	AVS Counter 0 Register
AVS_CNT1_REG	0x88	AVS Counter 1 Register
AVS_CNT_DIV_REG	0x8C	AVS Divisor
WDOG_IRQ_EN_REG	0xA0	Watchdog IRQ Enable Register
WDOG_IRQ_STA_REG	0xA4	Watchdog Status Register
WDOG_CTRL_REG	0xB0	Watchdog Control Register
WDOG_CFG_REG	0xB4	Watchdog Configuration Register
WDOG_MODE_REG	0xB8	Watchdog Mode Register



### 3.8.4. Timer Register Description

#### 3.8.4.1. Timer IRQ Enable Register (Default: 0x00000000)

Offset:0x0			Register Name: TMR_IRQ_EN_REG
Bit	R/W	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	TMR5_IRQ_EN. Timer 5 Interrupt Enable. 0: No effect; 1: Timer 5 Interval Value reached interrupt enable.
4	R/W	0x0	TMR4_IRQ_EN. Timer 4 Interrupt Enable. 0: No effect; 1: Timer 4 Interval Value reached interrupt enable.
3	R/W	0x0	TMR3_IRQ_EN. Timer 3 Interrupt Enable. 0: No effect; 1: Timer 3 Interval Value reached interrupt enable.
2	R/W	0x0	TMR2_IRQ_EN. Timer 2 Interrupt Enable. 0: No effect; 1: Timer 2 Interval Value reached interrupt enable.
1	R/W	0x0	TMR1_IRQ_EN. Timer 1 Interrupt Enable. 0: No effect; 1: Timer 1 Interval Value reached interrupt enable.
0	R/W	0x0	TMR0_IRQ_EN. Timer 0 Interrupt Enable. 0: No effect; 1: Timer 0 Interval Value reached interrupt enable.

#### 3.8.4.2. Timer IRQ Status Register (Default: 0x00000000)

Offset:0x04			Register Name: TMR_IRQ_STA_REG
Bit	R/W	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	TMR5_IRQ_PEND. Timer 5 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 5 counter value is reached.

4	R/W	0x0	TMR4_IRQ_PEND. Timer 4 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 4 counter value is reached.
3	R/W	0x0	TMR3_IRQ_PEND. Timer 3 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 3 counter value is reached.
2	R/W	0x0	TMR2_IRQ_PEND. Timer 2 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 2 counter value is reached.
1	R/W	0x0	TMR1_IRQ_PEND. Timer 1 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 1 interval value is reached.
0	R/W	0x0	TMR0_IRQ_PEND. Timer 0 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, timer 0 interval value is reached.

### 3.8.4.3. Timer 0 Control Register (Default: 0x00000004)

Offset:0x10			Register Name: TMR0_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR0_MODE. Timer 0 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR0_CLK_PRES. Select the pre-scale of timer 0 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR0_CLK_SRC.

			Timer 0 Clock Source. 00: 32K 01: OSC24M. 10: / 11: /
1	R/W	0x0	TMRO_RELOAD. Timer 0 Reload. 0: No effect, 1: Reload timer 0 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMRO_EN. Timer 0 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

#### 3.8.4.4. Timer 0 Interval Value Register

Offset:0x14			Register Name: TMRO_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMRO_INTV_VALUE. Timer 0 Interval Value.

**Note:**

The value setting should consider the system clock and the timer clock source.

#### 3.8.4.5. Timer 0 Current Value Register

Offset:0x18			Register Name: TMRO_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMRO_CUR_VALUE. Timer 0 Current Value.

**Note:**

Timer 0 current value is a 32-bit down-counter (from interval value to 0).

**3.8.4.6. Timer 1 Control Register (Default: 0x00000004)**

Offset:0x20			Register Name: TMR1_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR1_MODE. Timer 1 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR1_CLK_PRES. Select the pre-scale of timer 1 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR1_CLK_SRC. Timer 1 Clock Source. 00: 32K 01: OSC24M. 10: / 11: /.
1	R/W	0x0	TMR1_RELOAD. Timer 1 Reload. 0: No effect, 1: Reload timer 1 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR1_EN. Timer 1 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to

			down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.
--	--	--	--

### 3.8.4.7. Timer 1 Interval Value Register

Offset:0x24			Register Name: TMR1_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR1_INTV_VALUE. Timer 1 Interval Value.

**Note:**

the value setting should consider the system clock and the timer clock source.

### 3.8.4.8. Timer 1 Interval Value Register

Offset:0x28			Register Name: TMR1_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR1_CUR_VALUE. Timer 1 Current Value.

**Note:**

Timer 1 current value is a 32-bit down-counter (from interval value to 0).

### 3.8.4.9. Timer 2 Control Register (Default: 0x00000004)

Offset:0x30			Register Name: TMR2_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR2_MODE. Timer 2 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR2_CLK_PRE. Select the pre-scale of timer 2 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128

3:2	R/W	0x1	TMR2_CLK_SRC. Timer 2 Clock Source. 00: 32K 01: OSC24M. 1x: /.
1	R/W	0x0	TMR2_RELOAD. Timer 2 Reload. 0: No effect, 1: Reload timer 2 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR2_EN. Timer 2 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

#### 3.8.4.10. Timer 2 Interval Value Register

Offset:0x34			Register Name: TMR2_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR2_INTV_VALUE. Timer 2 Interval Value.

**Note:**

the value setting should consider the system clock and the timer clock source.

#### 3.8.4.11. Timer 2 Current Value Register

Offset:0x38			Register Name: TMR2_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR2_CUR_VALUE. Timer 2 Current Value.

**Note:**

Timer 2 current value is a 32-bit down-counter (from interval value to 0).

### 3.8.4.12. Timer 3 Control Register (Default: 0x00000004)

Offset:0x40			Register Name: TMR3_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR3_MODE. Timer 3 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR3_CLK_PRES. Select the pre-scale of timer 3 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR3_CLK_SRC. Timer 3 Clock Source. 00: 32K 01: OSC24M. 1x: /
1	R/W	0x0	TMR3_RELOAD. Timer 3 Reload. 0: No effect, 1: Reload timer 3 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR3_EN. Timer 3 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit

			should be set to 1 at the same time.
--	--	--	--------------------------------------

### 3.8.4.13. Timer 3 Interval Value Register

Offset:0x44			Register Name: TMR3_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR3_INTV_VALUE. Timer 3 Interval Value.

**Note:**

the value setting should consider the system clock and the timer clock source.

### 3.8.4.14. Timer 3 Current Value Register

Offset:0x48			Register Name: TMR3_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR3_CUR_VALUE. Timer 3 Current Value.

**Note:**

Timer 3 current value is a 32-bit down-counter (from interval value to 0).

### 3.8.4.15. Timer 4 Control Register (Default: 0x00000004)

Offset:0x50			Register Name: TMR4_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR4_MODE. Timer 4 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR4_CLK_PRES. Select the pre-scale of timer 4 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR4_CLK_SRC.



			Timer 4 Clock Source. 00: 32K 01: OSC24M. 10: External CLKIN0 11: /
1	R/W	0x0	TMR4_RELOAD. Timer 4 Reload. 0: No effect, 1: Reload timer 4 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR4_EN. Timer 4 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

**Note:**

If the clock source is External CLKIN, the interval value register is not used, the current value register is an up counter that counting from 0.

**3.8.4.16. Timer 4 Interval Value Register**

Offset:0x54			Register Name: TMR4_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR4_INTV_VALUE. Timer 4 Interval Value.

**Note:**

the value setting should consider the system clock and the timer clock source.

**3.8.4.17. Timer 4 Current Value Register**

Offset:0x58			Register Name: TMR4_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR4_CUR_VALUE. Timer 4 Current Value.

**Note:**

Timer 4 current value is a 32-bit down-counter (from interval value to 0).

### 3.8.4.18. Timer 5 Control Register (Default: 0x00000004)

Offset:0x60			Register Name: TMR5_CTRL_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR5_MODE. Timer 5 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR5_CLK_PRES. Select the pre-scale of timer 5 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR5_CLK_SRC. Timer 5 Clock Source. 00: 32K 01: OSC24M. 10: External CLKIN1 11: /.
1	R/W	0x0	TMR5_RELOAD. Timer 5 Reload. 0: No effect, 1: Reload timer 5 Interval value. After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR5_EN. Timer 5 Enable. 0: Stop/Pause, 1: Start. When the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to

			1. In timer pause state; the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.
--	--	--	---

**Note:**

If the clock source is External CLKIN, the interval value register is not used, the current value register is an up counter that counting from 0.

**3.8.4.19. Timer 5 Interval Value Register**

Offset:0x64			Register Name: TMR5_INTV_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR5_INTV_VALUE. Timer 5 Interval Value.

**Note:**

the value setting should consider the system clock and the timer clock source.

**3.8.4.20. Timer 5 Current Value Register**

Offset:0x68			Register Name: TMR5_CUR_VALUE_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TMR5_CUR_VALUE. Timer 5 Current Value.

**Note:**

Timer 5 current value is a 32-bit down-counter (from interval value to 0).

**3.8.4.21. AVS Counter Control Register (Default: 0x00000000)**

Offset:0x80			Register Name: AVS_CNT_CTL_REG
Bit	R/W	Default/Hex	Description
31:10	/	/	/
9	R/W	0x0	AVS_CNT1_PS. Audio/Video Sync Counter 1 Pause Control 0: Not pause, 1: Pause Counter 1.
8	R/W	0x0	AVS_CNT0_PS. Audio/Video Sync Counter 0 Pause Control 0: Not pause, 1: Pause Counter 0.
7:2	/	/	/
1	R/W	0x0	AVS_CNT1_EN. Audio/Video Sync Counter 1 Enable/ Disable. The counter source is OSC24M.

			0: Disable, 1: Enable.
0	R/W	0x0	AVS_CNT0_EN. Audio/Video Sync Counter 1 Enable/ Disable. The counter source is OSC24M. 0: Disable, 1: Enable.

#### 3.8.4.22. AVS Counter 0 Register (Default: 0x00000000)

Offset:0x84			Register Name: AVS_CNT0_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	AVS_CNT0. Counter 0 for Audio/ Video Sync Application The high 32 bits of the internal 33-bits counter register. The initial value of the internal 33-bits counter register can be set by software. The LSB bit of the 33-bits counter register should be zero when the initial value is updated. It will count from the initial value. The initial value can be updated at any time. It can also be paused by setting AVS_CNT0_PS to '1'. When it is paused, the counter won't increase.

#### 3.8.4.23. AVS Counter 1 Register (Default: 0x00000000)

Offset:0x88			Register Name: AVS_CNT1_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	AVS_CNT1. Counter 1 for Audio/ Video Sync Application The high 32 bits of the internal 33-bits counter register. The initial value of the internal 33-bits counter register can be set by software. The LSB bit of the 33-bits counter register should be zero when the initial value is updated. It will count from the initial value. The initial value can be updated at any time. It can also be paused by setting AVS_CNT1_PS to '1'. When it is paused, the counter won't increase.

#### 3.8.4.24. AVS Counter Divisor Register (Default: 0x05DB05DB)

Offset:0x8C			Register Name: AVS_CNT_DIV_REG
Bit	R/W	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x5DB	AVS_CNT1_D. Divisor N for AVS Counter 1 AVS CN1 CLK=24MHz/Divisor_N1. Divisor N1 = Bit [27:16] + 1.

			The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bits counter engine will maintain another 12-bits counter. The 12-bits counter is used for counting the cycle number of one 24Mhz clock. When the 12-bits counter reaches ( $\geq N$ ) the divisor value, the internal 33-bits counter register will increase 1 and the 12-bits counter will reset to zero and restart again.
15:12	/	/	/
11:0	R/W	0x5DB	AVS_CNT0_D. Divisor N for AVS Counter 0 AVS CNO CLK=24MHz/Divisor_NO. Divisor NO = Bit [11:0] + 1 The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bits counter engine will maintain another 12-bits counter. The 12-bits counter is used for counting the cycle number of one 24Mhz clock. When the 12-bits counter reaches ( $\geq N$ ) the divisor value, the internal 33-bits counter register will increase 1 and the 12-bits counter will reset to zero and restart again.

**Note:**

Divisor N can be configured by software at any time.

**3.8.4.25. Watchdog IRQ Enable Register (Default: 0x00000000)**

Offset:0xA0			Register Name: WDOG_IRQ_EN_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	WDOG_IRQ_EN. Watchdog Interrupt Enable. 0: No effect, 1: Watchdog interrupt enable.

**3.8.4.26. Watchdog Status Register (Default: 0x00000000)**

Offset:0xA4			Register Name: WDOG_IRQ_STA_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	WDOG_IRQ_PEND. Watchdog n IRQ Pending. Set 1 to the bit will clear it. 0: No effect, 1: Pending, watchdog interval value is reached.

**3.8.4.27. Watchdog Control Register (Default: 0x00000000)**

Offset:0xB0			Register Name: WDOG_CTRL_REG
Bit	R/W	Default/Hex	Description
31:13	/	/	/
12:1	R/W	0x0	WDOG_KEY_FIELD. Watchdog Key Field. Should be written at value 0xA57. Writing any other value in this field aborts the write operation.
0	R/W	0x0	WDOG_RSTART. Watchdog Restart. 0: No effect, 1: Restart watchdog.

**3.8.4.28. Watchdog Configuration Register (Default: 0x00000001)**

Offset:0xB4			Register Name: WDOG_CFG_REG
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x1	WDOG_CONFIG. Watchdog generates a reset signal 00: / 01: To whole system 10: Only interrupt 11: /

**3.8.4.29. Watchdog Mode Register (Default: 0x00000000)**

Offset:0xB8			Register Name: WDOG_MODE_REG
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:4	R/W	0x0	WDOG_INTV_VALUE. Watchdog Interval Value Watchdog clock source is <i>OSC24M / 750</i> . If the clock source is turned off, Watchdog 1 will not work. 0000: 16000 cycles (0.5s) 0001: 32000 cycles (1s) 0010: 64000 cycles (2s) 0011: 96000 cycles (3s) 0100: 128000 cycles (4s) 0101: 160000 cycles (5s) 0110: 192000 cycles (6s)

			0111: 256000 cycles (8s) 1000: 320000 cycles (10s) 1001: 384000 cycles (12s) 1010: 448000 cycles (14s) 1011: 512000 cycles (16s) others: /
3:1	/	/	/
0	R/W	0x0	WDOG_EN. Watchdog Enable. 0: No effect; 1: Enable watchdog.

### 3.8.5. Programming Guidelines

#### 3.8.5.1. Timer

Take making a 1ms delay for an example, 24M clock source, single mode and 2 pre-scale will be selected in the instance.

```
writel(0x2EE0,TMR_0_INTV);           //Set interval value
writel(0x94, TMR_0_CTRL);           //Select Single mode,24MHz clock source,2 pre-scale
writel(readl(TMR_0_CTRL)|(1<<1), TMR_0_CTRL); //Set Reload bit
while((readl(TMR_0_CTRL)>>1)&1);     //Waiting Reload bit turns to 0
writel(readl(TMR_0_CTRL)|(1<<0), TMR_0_CTRL); //Enable Timer0
```

#### 3.8.5.2. Watchdog

##### 3.8.5.2.1. Watchdog Reset

In the following instance making configurations for Watchdog: configurate clock source as 24M/750, configurate Interval Value as 1s and configurate Watchdog Configuration as To whole system. This instance indicates that reset system after 1s.

```
writel(0x1, WDOG_CONFIG);           //To whole system
writel(0x10, WDOG_MODE);           //Interval Value set 1s
writel(readl(WDOG_MODE)|(1<<0), WDOG_MODE); //Enable Watchdog
```

##### 3.8.5.2.2. Watchdog Restart

In the following instance making configurations for Watchdog: configurate clock source as 24M/750, configurate Interval Value as 1s and configurate Watchdog Configuration as To whole system. In the following instance, if the time of other codes is larger than 1s, watchdog will reset the whole system. If the sentence of restart watchdog is implemented inside 1s, watchdog will be restarted.

```
writel(0x1, WDOG_CONFIG);           //To whole system
writel(0x10, WDOG_MODE);           //Interval Value set 1s
writel(readl(WDOG_MODE)|(1<<0), WDOG_MODE); //Enable Watchdog
----other codes---
writel(readl(WDOG_CTRL)|(0xA57<<1)|(1<<0), WDOG_CTRL); //Writel 0xA57 at Key Field and Restart Watchdog
```



## 3.9. Trusted Watchdog

### 3.9.1. Overview

The trusted watchdog is primarily used to protect the trusted world operations from denial of service when secure services are dependent to the RichOS scheduler. For example, if the trusted world is not entered after a defined time limit the SoC is re-started to perform an authentication of the system.

The trusted watchdog can also be used to mask the real cause of a security error thanks to the delayed warm reset it generates.

The TWD includes the following features:

- Support 64-bit counter
- Support to generate warm reset to system
- Support to be restarted by software
- Contains 4 ***Synchronize Data Counter Registers***

### 3.9.2. Functionalities Description

The trusted watchdog must always be running when the SoC wakes up from cold reset and can be refreshed, suspended, or reset only by secure accesses. And a clock of at least 32 kHz is used when the device is not a power saving cycle.

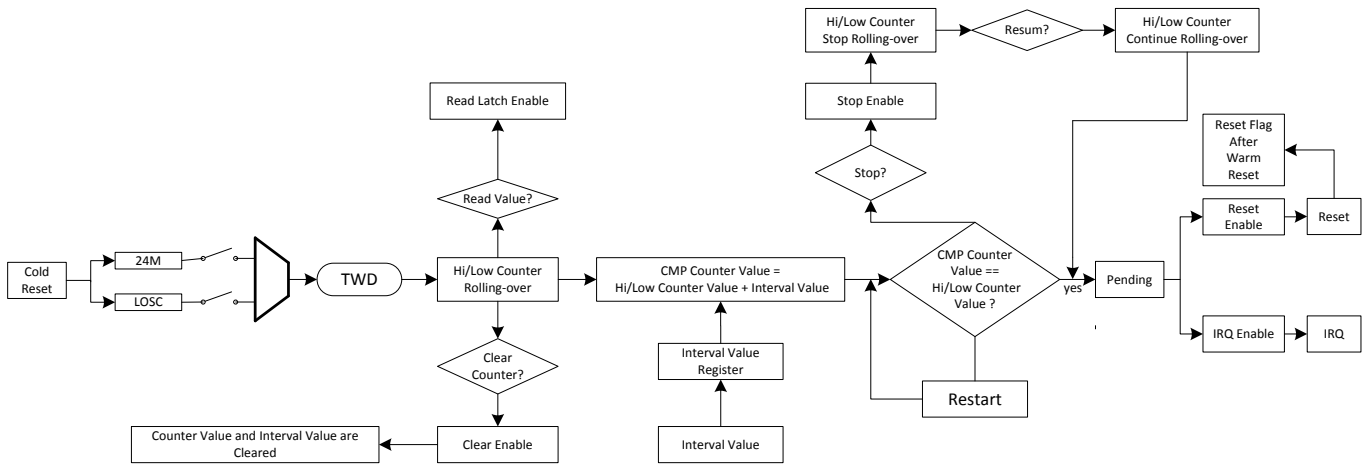
#### 3.9.2.1. Typical Applications

##### 3.9.2.1.1. TWD reset

The trusted watchdog is able to generate a SoC warm reset after a duration programmed into the timer or set by default in hardware. And the flag indicating the occurrence of a watchdog triggered warm reset has occurred since the last cold reset.

Clock sources driving the watchdog timer must be controlled or managed by a trusted entity. This means that non-trusted world accesses are not permitted to turn on, turn off or modify the characteristics of clock source. The ***Clear Enable*** will reset relevant bits in the watchdog registers, except the reset flag.

### 3.9.2.2. Functional Block Diagram



### 3.9.3. Operation Modes

#### 3.9.3.1. Clock

Trusted watchdog clock is LOSC by default, and can be switched to OSC24M. Its clock gating is invalid since the 64-bit counter of TWD will be overturned after power on by default. If TWD is not used, its clock gating can be cleared to 0 to reduce system power consumption.

#### 3.9.3.2. Interrupt

After writing a value to TWD Interval Value Register, TWD Low/High Counter Compare Registers will be updated automatically. When the value of TWD Low/High Counter Registers is identical to value of TWD Low/High Counter Compare Registers, the interrupt pending will be suspended, and corresponding interrupt will be generated if interrupts are enabled, and system will reset if reset is enabled.

### 3.9.4. TWD Register List

Module Name	Base Address
TWD	0x08001800

Register Name	Offset	Description
TWD_STATUS_REG	0x0000	TWD Status Register
TWD_CTRL_REG	0x0010	TWD Control Register
TWD_RESTART_REG	0x0014	TWD Restart Register
TWD_LOW_CNT_REG	0x0020	TWD Low Counter Register
TWD_HIGH_CNT_REG	0x0024	TWD High Counter Register
TWD_INTV_VAL_REG	0x0030	TWD Interval Value Register
TWD_LOW_CNT_CMP_REG	0x0040	TWD Low Counter Compare Register
TWD_HIGH_CNT_CMP_REG	0x0044	TWD High Counter Compare Register
SST_NV_CNT_REG	0x0100	Secure Storage NV-Counter Register
SYN_DATA_CNT_REG0	0x0110	Synchronize Data Counter Register 0
SYN_DATA_CNT_REG1	0x0114	Synchronize Data Counter Register 1
SYN_DATA_CNT_REG2	0x0118	Synchronize Data Counter Register 2
SYN_DATA_CNT_REG3	0x011C	Synchronize Data Counter Register 3

### 3.9.5. TWD Register Description

#### 3.9.5.1. TWD Status Register (Default: 0x00000000)

Offset: 0x0000			Register Name: <b>TWD_STATUS_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	TWD_PEND_FLAG. Interrupt pending. Set 1 to the bit will clear it. 0: No effect. 1: Pending.

#### 3.9.5.2. TWD Control Register (Default: 0x00000000)

Offset: 0x0010			Register Name: <b>TWD_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	CNT64_CLK_SRC_SEL. 64-bit counter clock source select. 0: LOSC. 1: OSC24M.
30:10	/	/	/
9	R/W	0x0	TWD_RESET_EN. TWD reset enable. 0: Reset disable. 1: Reset enable.
8	R/W	0x0	TWD_INT_EN. TWD Interrupt Enable. 0: Interrupt disable. 1: Interrupt enable.
7:2	/	/	/
1	R/W	0x0	TWD_STOP_EN. TWD stop enable. 0: Resume rolling-over. 1: Stop rolling-over.
0	R/W	0x0	TWD_CLR_EN. TWD clear enable. 0: No effect. 1: To clear relevant registers and it will change to zero after the registers are cleared.

### 3.9.5.3. TWD Restart Register (Default: 0x00000000)

Offset: 0x0014			Register Name: <b>TWD_RESTART_REG</b>
Bit	R/W	Default/Hex	Description
31:28	/	/	/
27:16	WO	0x0	TWD_RESTART_KEYFILED. Should be written at value 0xD14. Writing any other value in this field aborts the write operation.
15:1	/	/	/
0	WO	0x0	TWD_RESTART_EN. If writing '1' in this bit, the value of <i>Counter Compare Registers</i> would change. 0: No effect. 1: Restart enable.

### 3.9.5.4. TWD Low Counter Register (Default: 0x00000000)

Offset: 0x0020			Register Name: <b>TWD_LOW_CNT_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	TWD_LOW_CNT. The TWD low 32-bit counter.

### 3.9.5.5. TWD High Counter Register (Default: 0x00000000)

Offset: 0x0024			Register Name: <b>TWD_HIGH_CNT_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	TWD_HIGH_CNT. The TWD high 32-bit counter.

### 3.9.5.6. TWD Interval Value Register (Default: 0x00000000)

Offset: 0x0030			Register Name: <b>TWD_INTV_VAL_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	TWD_INTV_VAL. The TWD interval value.

**3.9.5.7. TWD Low Counter Compare Register (Default: 0x00000000)**

Offset: 0x0040			Register Name: <b>TWD_LOW_CNT_CMP_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	TWD_LOW_CNT. The TWD low 32-bit compare counter.

**3.9.5.8. TWD High Counter Compare Register (Default: 0x00000000)**

Offset: 0x0044			Register Name: <b>TWD_HIGH_CNT_CMP_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	TWD_HIGH_CNT. The TWD high 32-bit compare counter.

**3.9.5.9. Secure Storage NV-Counter Register (Default: 0x00000000)**

Offset: 0x0100			Register Name: <b>SST_NV_CNT_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SST_NV_CNT. This counter protects the trusted world Secure Storage file from replay attacks.

**3.9.5.10. Synchronize Data Counter Register 0 (Default: 0x00000000)**

Offset: 0x0110			Register Name: <b>SYN_DATA_CNT_REG0</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SYN_DATA_CNT0. This counter is used for synchronizing data stores against replay attacks.

**3.9.5.11. Synchronize Data Counter Register 1 (Default: 0x00000000)**

Offset: 0x0114			Register Name: <b>SYN_DATA_CNT_REG1</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SYN_DATA_CNT1. This counter is used for synchronizing data stores against replay attacks.

### 3.9.5.12. Synchronize Data Counter Register 2 (Default: 0x00000000)

Offset: 0x0118			Register Name: <b>SYN_DATA_CNT_REG2</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SYN_DATA_CNT2. This counter is used for synchronizing data stores against replay attacks.

### 3.9.5.13. Synchronize Data Counter Register 3 (Default: 0x00000000)

Offset: 0x011C			Register Name: <b>SYN_DATA_CNT_REG3</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SYN_DATA_CNT3. This counter is used for synchronizing data stores against replay attacks.

## 3.9.6. Programming Guidelines

When TWD chooses LOSC as the clock source, operations on TWD-related registers will become valid after a period of delay.

After TWD restart, the value of **TWD Low/High Counter Compare Registers** will be auto updated to the added value of **TWD Interval Value Register** and **TWD Low/High Counter Registers**.

## 3.10. High-speed Timer

### 3.10.1. Overview

High Speed Timer' clock source is fixed to AHBCLK, which is much higher than OSC24M. Compared with other timers, High Speed Timer calculates much more accurately. When the relevant bit in the Control Register is set 1, HSTimer goes into the test mode, which is used to System Simulation. While the current value in both LO and HI Current Value Register are counting down to zero, the timer will generate interrupt if set interrupt enable bit.

The HSTimer includes the following features:

- 5 HSTimers with individual 56-bit counter each
- Each HSTimer could generate a pending
- Clock source is synchronized with AHB clock, which means calculating much more accurate than other timers
- Support Test Mode for System Simulation

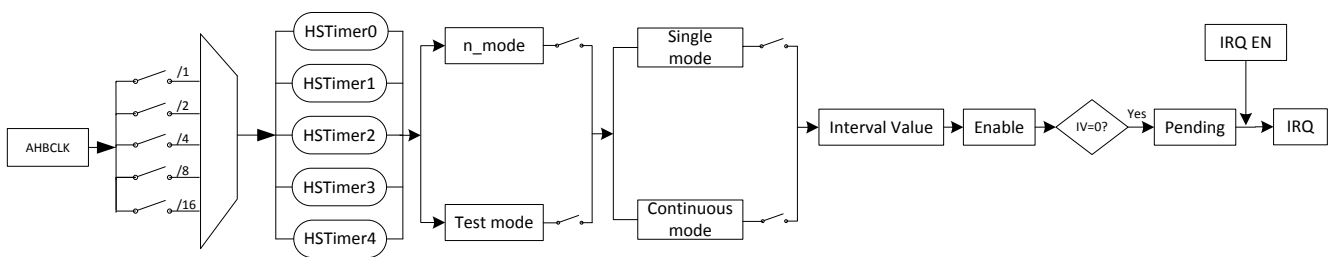


### 3.10.2. Functionalities Description

#### 3.10.2.1. Typical Applications

HSTimer provides much more accurate delay and scheduler interrupt

#### 3.10.2.2. Functional Block Diagram



HSTimer function structure and work flow

HSTimer has two work modes and two count modes. n\_mode is used for normal counting and Test mode is used in System Simulation. Each work mode has the two count mode: Single mode and Continuous mode. These two count modes have the same principle with Timers', which means when Current Value counts down to 0, HSTimer will be disable in Single mode ,but HSTimer will not be disable and counts from Interval value again in Continuous mode. About HSTimer 56-bit counter, it is combined with a high 24-bit counter(*HS Timer Current Value Hi Register*) and a low 32-bit counter(*HS Timer Current Value Lo Register*).

### 3.10.3. Operation Principle

#### 3.3.3.1. HSTimer clock gating and software reset

By default the HSTimer clock gating is mask. When it is necessary to use HSTimer, it's clock gating should be open in *AHB1 Module Clock Gating Register* and then de-assert the software reset in *AHB1 Module Software Reset Register* on CCU module. If it is no need to use HSTimer, both the gating bit and software reset bit should be set 0.

#### 3.3.3.2. HSTimer reload bit

Differing from the reload of Timer, when interval value is reloaded into current value register, the reload bit would not turn to 0 automatically until you clear it. If software hopes the current value register to down-count from the new interval value in pause status, the reload bit and the enable bit should be written 1 at the same time.

### 3.10.4. HSTimer Register List

Module Name	Base Address
High Speed Timer	0x00801000

Register Name	Offset	Description
HS_TMR_IRQ_EN_REG	0x00	HS Timer IRQ Enable Register
HS_TMR_IRQ_STAS_REG	0x04	HS Timer Status Register
HS_TMR_CTRL_REG	$0x10+N*0x20(N=0,1,2,3,4)$	HS Timer N Control Register
HS_TMR_INTV_LO_REG	$0x14+N*0x20(N=0,1,2,3,4)$	HS Timer N Interval Value Low Register
HS_TMR_INTV_HI_REG	$0x18+N*0x20(N=0,1,2,3,4)$	HS Timer N Interval Value High Register
HS_TMR_CURNT_LO_REG	$0x1C+N*0x20(N=0,1,2,3,4)$	HS Timer N Current Value Low Register
HS_TMR_CURNT_HI_REG	$0x20+N*0x20(N=0,1,2,3,4)$	HS Timer N Current Value High Register

### 3.10.5. HSTimer Register Description

#### 3.10.5.1. HS Timer IRQ Enable Register (Default: 0x00000000)

Offset:0x0			Register Name: HS_TMR_IRQ_EN_REG
Bit	R/W	Default/Hex	Description
31:5	/	/	/
4	R/W	0x0	HS_TMR4_INT_EN. High Speed Timer 4 Interrupt Enable. 0: No effect; 1: High Speed Timer 4 Interval Value reached interrupt enable.
3	R/W	0x0	HS_TMR3_INT_EN. High Speed Timer 3 Interrupt Enable. 0: No effect; 1: High Speed Timer 3 Interval Value reached interrupt enable.
2	R/W	0x0	HS_TMR2_INT_EN. High Speed Timer 2 Interrupt Enable. 0: No effect; 1: High Speed Timer 2 Interval Value reached interrupt enable.
1	R/W	0x0	HS_TMR1_INT_EN. High Speed Timer 1 Interrupt Enable. 0: No effect; 1: High Speed Timer 1 Interval Value reached interrupt enable.
0	R/W	0x0	HS_TMR0_INT_EN. High Speed Timer 0 Interrupt Enable. 0: No effect; 1: High Speed Timer 0 Interval Value reached interrupt enable.

#### 3.10.5.2. HS Timer IRQ Status Register (Default: 0x00000000)

Offset:0x4			Register Name: HS_TMR_IRQ_STAS_REG
Bit	R/W	Default/Hex	Description
31:5	/	/	/
4	R/W	0x0	HS_TMR4_IRQ_PEND. High Speed Timer 4 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, High speed timer 4 interval value is reached.
3	R/W	0x0	HS_TMR3_IRQ_PEND. High Speed Timer 3 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, High speed timer 3 interval value is reached.

2	R/W	0x0	HS_TMR2_IRQ_PEND. High Speed Timer 2 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, High speed timer 2 interval value is reached.
1	R/W	0x0	HS_TMR1_IRQ_PEND. High Speed Timer 1 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, High speed timer 1 interval value is reached.
0	R/W	0x0	HS_TMR0_IRQ_PEND. High Speed Timer 0 IRQ Pending. Set 1 to the bit will clear it. 0: No effect; 1: Pending, High speed timer 0 interval value is reached.

### 3.10.5.3. HS Timer N Control Register (Default: 0x00000000)

Offset:0x10+N*0x20 (N=0,1,2,3,4)			Register Name: HS_TMR_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	HS_TMR_TEST. High speed timer test mode. In test mode, the low register should be set to 0x1, the high register will down counter. The counter needs to be reloaded. 0: n_mode; 1: test mode.
30:8	/	/	/
7	R/W	0x0	HS_TMR_MODE. High Speed Timer mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	HS_TMR_CLK Select the pre-scale of the high speed timer clock sources. 000: /1 001: /2 010: /4 011: /8 100: /16 Others: /
3:2	/	/	/
1	R/W	0x0	HS_TMR_RELOAD. High Speed Timer Reload. 0: No effect 1: Reload High Speed Timer Interval Value.

0	R/W	0x0	<p>HS_TMRO_EN. High Speed Timer Enable.</p> <p>0: Stop/Pause, 1: Start.</p> <p>If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0.</p> <p>If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1.</p> <p>In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.</p>
---	-----	-----	--

#### 3.10.5.4. HS Timer N Interval Value Lo Register

Offset:0x14+N*0x20 (N=0,1,2,3,4)			Register Name: HS_TMR_INTV_LO_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	x	HS_TMR_INTV_VALUE_LO. High Speed Timer Interval Value [31:0].

#### 3.10.5.5. HS Timer N Interval Value Hi Register

Offset:0x18+N*0x20 (N=0,1,2,3,4)			Register Name: HS_TMR_INTV_HI_REG
Bit	R/W	Default/Hex	Description
31:24	/	/	/
23:0	R/W	x	HS_TMR_INTV_VALUE_HI. High Speed Timer Interval Value [55:32].

**Note:**

The interval value register is a 56-bit register. When read or write the interval value, the Lo register should be read or write first. And the Hi register should be written after the Lo register.

#### 3.10.5.6. HS Timer N Current Value Lo Register

Offset:0x1C+N*0x20 (N=0,1,2,3,4)			Register Name: HS_TMR_CURNT_LO_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	x	HS_TMR_CUR_VALUE_LO. High Speed Timer Current Value [31:0].

**3.10.5.7. HS Timer N Current Value Hi Register**

Offset:0x20+N*0x20 (N=0,1,2,3,4)			Register Name: HS_TMR_CURNT_HI_REG
Bit	R/W	Default/Hex	Description
31:24	/	/	/
23:0	R/W	x	HS_TMR_CUR_VALUE_HI. High Speed Timer Current Value [55:32].

**Note:**

- HSTimer current value is a 56-bit down-counter (from interval value to 0).
- The current value register is a 56-bit register. When read or write the current value, the Lo register should be read or write first.

### 3.10.6. Programming Guidelines

Take making a 1us delay using HSTimer0 for an instance as follow, AHB1CLK will be configurated as 100MHz and n\_mode,Single mode and 2 pre-scale will be selected in this instance.

```
writel(0x0, HS_TMR0_INTV_HI); //Set interval value Hi 0x0
writel(0x32, HS_TMR0_INTV_LO); //Set interval value Lo 0x32
writel(0x90, HS_TMR0_CTRL); //Select n_mode,2 pre-scale,single mode
writel(readl(HS_TMR0_CTRL)|(1<<1), HS_TMR0_CTRL); //Set Reload bit
writel(readl(HS_TMR0_CTRL)|(1<<0), HS_TMR0_CTRL); //Enable HSTimer0
While(!(readl(HS_TMR_IRQ_STAT)&1)); //Wait for HSTimer0 to generate pending
Writel(1,HS_TMR_IRQ_STAT); //Clear HSTimer0 pending
```

## 3.11. PWM

### 3.11.1. Overview

The output of the PWM is a toggling signal whose frequency and duty cycle can be modulated by its programmable registers. Each channel has a dedicated internal 16-bit up counter. If the counter reaches the value stored in the channel period register, it resets. At the beginning of a count period cycle, the PWMOUT is set to active state and count from 0x0.

The Special Mode of PWM 0 is used for the 1-wire digital interface of some LED ICs. If PWM 0 is working in the Special Mode, there are totally 4 kinds of Steps that could be set by programmer according to the LED IC's feature, and the default Step is 32 steps. When control the brightness of backlight, programmer should calculate a proper clock divider, entire cycle number and active cycle number, so that PWM 0 can output a suitable duty cycle waveform. If the IC is from shutdown to power on, it is necessary that the output is holding on in high status more than 40us, which is also called ready time; and in order to shut down the IC, the output is holding on in low status more than 3ms.

The PWM includes the following features:

- Support 4 PWM channels and 2 output pins for PWM1/2/3 channel each
- Could be used for some 1-wire digital interface of some LED ICs
- Support outputting 2 kinds of waveform: continuous waveform and pulse waveform
- 0% to 100% adjustable duty cycle
- 0Hz~12MHz output frequency

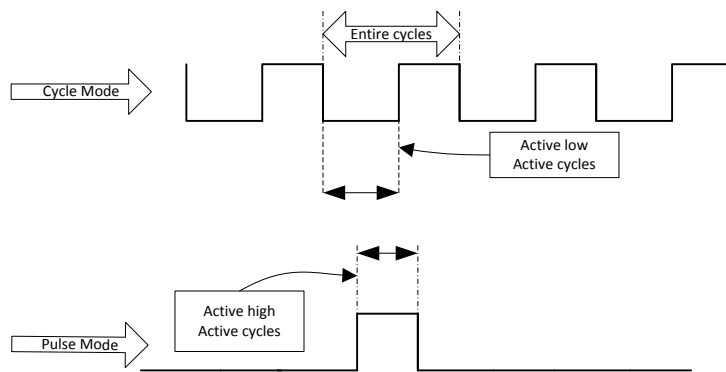


### 3.11.2. Functionalities Description

#### 3.11.2.1. Typical Applications

- Suitable for display device, such as LVDS, HV, MIPI DSI.
- Could be used for adjusting brightness for some 1-wire digital interface of some LED ICs

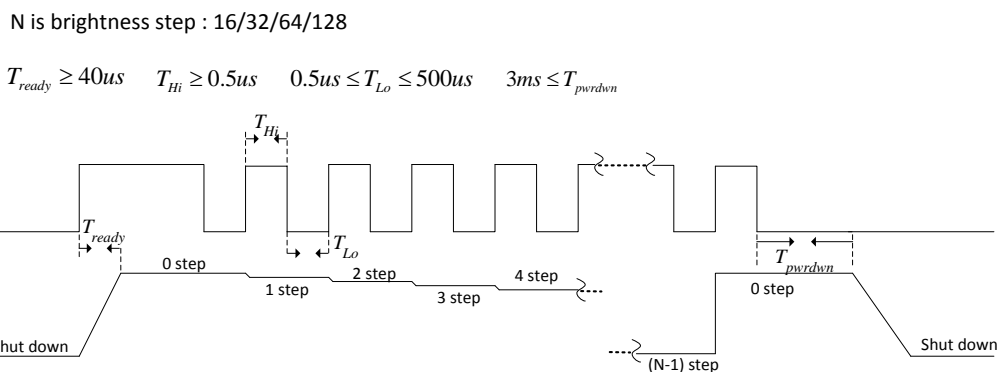
#### 3.11.2.2. Functional Block Diagram



PWM output waveform

When PWM is enabling, the PWM can output two signals, which are reversed on two pins. And when PWM is disabling, the PWM can control the status of two pins. The PWM divider divides the clock (24MHz) by 1-64 according to the pre-scalar bits in the PWM control register. The PWM output Frequency can be divided by 65536 at most. In PWM cycle mode, the output will be a square waveform; the frequency is set to the period register. In PWM pulse mode, the output will be a positive pulse or a negative pulse.

In Continuous Mode PWM Outputs continuous waveform and in Pulse Mode it outputs a pulse waveform. Each PWM channel has 2 16-bit up counters exiting in corresponding Period Register, whose [31:16] bits indicate one 16-bit up counter for counting Entire Cycle and [15:0] bits indicate the other 16-bit up counter for counting Active Cycle.



## PWM0 Special Mode

An application example of PWM 0 in Special Mode: assuming the brightness step is 32, and the regulation step is 16. If programmer wants the backlight darker, then changing the regulation step number, which should be less than 31, such as 20, and then PWM 0 would generate 4 pulses continuously. Or if programmer wants the backlight lighter, then changing the regulation step number, such as 1, and then PWM 0 would generate 17 pulses continuously. Since the regulation is one-way linear cycle, the regulation step would increase from 16 to 31, then rolling over into 0, and become 1 finally.

### 3.11.3. Operation Principle

#### 3.11.3.1. PWM output pins

Except for PWM0 channel, the other 3 PWM channels have a couple of output pins, which have 180° phase difference. Before using PWM, it is necessary to configure PWM output at the corresponding pin. Take configuring PWM1 output for an example, selecting PWM1\_P function at *PH Configuration Register 1* bit[2:0] indicates PWM1 will be outputted from the pin. For the other pin, it outputs the 180° phase difference PWM1 waveform and PWM1\_N function should be selected for it at *PH Configuration Register 1* bit[6:4].

### 3.11.4. PWM Register List

Module Name	Base Address
PWM	0x06001400

Register Name	Offset	Description
PWM_CH0_CTRL	0x00	PWM Channel 0 Control Register
PWM_CH0_PERIOD	0x04	PWM Channel 0 Period Register
PWM_CH1_CTRL	0x10	PWM Channel 1 Control Register
PWM_CH1_PERIOD	0x14	PWM Channel 1 Period Register
PWM_CH2_CTRL	0x20	PWM Channel 2 Control Register
PWM_CH2_PERIOD	0x24	PWM Channel 2 Period Register
PWM_CH3_CTRL	0x30	PWM Channel 3 Control Register
PWM_CH3_PERIOD	0x34	PWM Channel 3 Period Register

### 3.11.5. PWM Register Description

#### 3.11.5.1. PWM Channel 0 Control Register

Offset:0x0			Register Name: PWM_CH0_CTRL
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R	0x0	PWM0_RDY. PWM0 period register ready. 0: PWM0 period register is ready to write; 1: PWM0 period register is busy.
27:20	R/W	0x0	REGULATE_STEP. Regulation steps. If the brightness step is N, the regulation step is M, then N >= M. When writing a number, which is larger than brightness step, in [27:20], the write operation is invalid.
19:18	R/W	0x1	STEP_BRIGHT_SLT. Select brightness steps. 00: 16 Steps, 01: 32 Steps, 10: 64 Steps, 11: 128 Steps
17	/	/	/
16	R/W	0x0	PWM0_WORK_MODE PWM 0 works mode. 0: Normal Mode, 1: Special Mode. Special mode is used for 1-wire digital interface in some ICs. And when selecting the Special mode, the cycle mode or pulse mode will be invalid.
15:12	/	/	/
11:10	R/W	0x0	PWM0_PIN_STATUS. When PWM is disable, two output pins' status can be changed. 00: Both low 01: Pin 0 is high, pin 1 is low 10: Pin 0 is low, pin 1 is high 11: Both high
9	/	/	/
8	R/W	0x0	PWM_CH0_PUL_START. PWM Channel 0 pulse output start. 0: No effect, 1: Output 1 pulse.

			The pulse width should be according to the period 0 register [15:0], and the pulse state should be according to the active state. After the pulse is finished, the bit will be cleared automatically.
7	R/W	0x0	PWM_CHANNELO_MODE. 0: Cycle Mode, 1: Pulse Mode.
6	R/W	0x0	SCLK_CHO_GATING. Gating the Special Clock for PWM0 (0: mask, 1: pass).
5	R/W	0x0	PWM_CHO_ACT_STA. PWM Channel 0 Active State. 0: Low Level, 1: High Level.
4	R/W	0x0	PWM_CHO_EN. PWM Channel 0 Enable. 0: Disable, 1: Enable.
3:0	R/W	0x0	PWM_CHO_PRESCAL. PWM Channel 0 Pre-scalar. These bits should be setting before the PWM Channel 0 clock gate on. 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 others: /

### 3.11.5.2. PWM Channel 0 Period Register

Offset:0x4			Register Name: PWM_CHO_PERIOD
Bit	R/W	Default/Hex	Description
31:16	R/W	x	PWM_CHO_ENTIRE_CYS Number of the entire cycles in the PWM clock. 0 = 1 cycle 1 = 2 cycles ..... N = N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK (PWM CLK = 24MHz/pre-scale).
15:0	R/W	x	PWM_CHO_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock.

			0 = 0 cycle 1 = 1 cycles ..... N = N cycles
--	--	--	--

**Note:**

When the active cycles are larger than the period cycles, the duty cycle is 100%.

**3.11.5.3. PWM Channel 1 Control Register (Default: 0x00000000)**

Offset:0x10			Register Name: PWM_CH1_CTRL
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R	0x0	PWM1_RDY. PWM1 period register ready. 0: PWM1 period register is ready to write; 1: PWM1 period register is busy.
27:12	/	/	/
11:10	R/W	0x0	PWM1_PIN_STATUS. When PWM is disable, two output pins' status can be changed. 00: Both low 01: Pin 0 is high, pin 1 is low 10: Pin 0 is low, pin 1 is high 11: Both high
9	/	/	/
8	R/W	0x0	PWM_CH1_PUL_START. PWM Channel 1 pulse output start. 0: No effect, 1: Output 1 pulse. The pulse width should be according to the period 1 register [15:0], and the pulse state should be according to the active state. After the pulse is finished, the bit will be cleared automatically.
7	R/W	0x0	PWM_CHANNEL1_MODE. 0: Cycle Mode, 1: Pulse Mode.
6	R/W	0x0	SCLK_CH1_GATING. Gating the Special Clock for PWM1 (0: mask, 1: pass).
5	R/W	0x0	PWM_CH1_ACT_STA. PWM Channel 1 Active State. 0: Low Level, 1: High Level.
4	R/W	0x0	PWM_CH1_EN. PWM Channel 1 Enable. 0: Disable,

			1: Enable.
3:0	R/W	0x0	PWM_CH1_PRESCAL. PWM Channel 1 Pre-scalar. These bits should be setting before the PWM Channel 1 clock gate on. 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 others: /

#### 3.11.5.4. PWM Channel 1 Period Register

Offset:0x14			Register Name: PWM_CH1_PERIOD
Bit	R/W	Default/Hex	Description
31:16	R/W	x	PWM_CH1_ENTIRE_CYS Number of the entire cycles in the PWM clock. 0 = 1 cycle 1 = 2 cycles ..... N = N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK (PWM CLK = 24MHz/pre-scale).
15:0	R/W	x	PWM_CH1_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock. 0 = 0 cycle 1 = 1 cycles ..... N = N cycles

**Note:**

When the active cycles are larger than the period cycles, the duty cycle is 100%.

#### 3.11.5.5. PWM Channel 2 Control Register (Default: 0x00000000)

Offset:0x20			Register Name: PWM_CH2_CTRL
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R	0x0	PWM2_RDY. PWM2 period register ready. 0: PWM2 period register is ready to write;

			1: PWM2 period register is busy.
27:12	/	/	/
11:10	R/W	0x0	PWM2_PIN_STATUS. When PWM is disable, two output pins' status can be changed. 00: Both low 01: Pin 0 is high, pin 1 is low 10: Pin 0 is low, pin 1 is high 11: Both high
9	/	/	/
8	R/W	0x0	PWM_CH2_PUL_START. PWM Channel 2 pulse output start. 0: No effect, 1: Output 1 pulse. The pulse width should be according to the period 2 register [15:0], and the pulse state should be according to the active state. After the pulse is finished, the bit will be cleared automatically.
7	R/W	0x0	PWM_CHANNEL2_MODE. 0: Cycle Mode, 1: Pulse Mode.
6	R/W	0x0	SCLK_CH2_GATING. Gating the Special Clock for PWM 2 (0: mask, 1: pass).
5	R/W	0x0	PWM_CH2_ACT_STA. PWM Channel 2 Active State. 0: Low Level, 1: High Level.
4	R/W	0x0	PWM_CH2_EN. PWM Channel 2 Enable. 0: Disable, 1: Enable.
3:0	R/W	0x0	PWM_CH2_PRESCAL. PWM Channel 2 Pre-scalar. These bits should be setting before the PWM Channel 2 clock gate on. 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 others: /



### 3.11.5.6. PWM Channel 2 Period Register

Offset:0x24			Register Name: PWM_CH2_PERIOD
Bit	R/W	Default/Hex	Description
31:16	R/W	x	PWM_CH2_ENTIRE_CYS Number of the entire cycles in the PWM clock. 0 = 1 cycle 1 = 2 cycles ..... N = N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK (PWM CLK = 24MHz/pre-scale).
15:0	R/W	x	PWM_CH2_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock. 0 = 0 cycle 1 = 1 cycles ..... N = N cycles

**Note:**

When the active cycles are larger than the period cycles, the duty cycle is 100%.

### 3.11.5.7. PWM Channel 3 Control Register (Default: 0x00000000)

Offset:0x30			Register Name: PWM_CH3_CTRL
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R	0x0	PWM3_RDY. PWM3 period register ready. 0: PWM3 period register is ready to write; 1: PWM3 period register is busy.
27:12	/	/	/
11:10	R/W	0x0	PWM3_PIN_STATUS. When PWM is disable, two output pins' status can be changed. 00: Both low 01: Pin 0 is high, pin 1 is low 10: Pin 0 is low, pin 1 is high 11: Both high
9	/	/	/
8	R/W	0x0	PWM_CH3_PUL_START. PWM Channel 3 pulse output start. 0: No effect, 1: Output 1 pulse. The pulse width should be according to the period 3 register [15:0], and

			the pulse state should be according to the active state. After the pulse is finished, the bit will be cleared automatically.
7	R/W	0x0	PWM_CHANNEL3_MODE. 0: Cycle Mode, 1: Pulse Mode.
6	R/W	0x0	SCLK_CH3_GATING. Gating the Special Clock for PWM 3 (0: mask, 1: pass).
5	R/W	0x0	PWM_CH3_ACT_STA. PWM Channel 3 Active State. 0: Low Level, 1: High Level.
4	R/W	0x0	PWM_CH3_EN. PWM Channel 3 Enable. 0: Disable, 1: Enable.
3:0	R/W	0x0	PWM_CH3_PRESCAL. PWM Channel 3 Pre-scalar. These bits should be setting before the PWM Channel 3 clock gate on. 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 others: /

### 3.11.5.8. PWM Channel 3 Period Register

Offset:0x34			Register Name: PWM_CH3_PERIOD
Bit	R/W	Default/Hex	Description
31:16	R/W	x	PWM_CH3_ENTIRE_CYS Number of the entire cycles in the PWM clock. 0 = 1 cycle 1 = 2 cycles ..... N = N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK (PWM CLK = 24MHz/pre-scale).
15:0	R/W	x	PWM_CH3_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock. 0 = 0 cycle

			1 = 1 cycles ..... N = N cycles
--	--	--	---------------------------------------

**Note:**

When the active cycles are larger than the period cycles, the duty cycle is 100%.

### 3.11.6. Programming Guidelines

#### 3.11.6.1. PWM0 normal mode

Assuming that PWM0 outputs 50% duty cycle and 10KHz frequency waveform. 2 pre-scale and Low Level will be selected in the following instance.

```
writel((readl(PH_CFG0)&~(7<<24))|(2<<24),PH_CFG0);           //Select PWM0 output
while(readl(PWM_CHO_CTRL)>>28&1);                             //wait for period register ready to write if it is busy
writel(0x04af0258,PWM_CHO_PERIOD);                             //Configure PWM Channel 0 Period Register
writel(0<<16|0<<7|1<<0,PWM_CHO_CTRL);                         //Select normal mode,cycle mode and 2 pre-scale
writel(readl(PWM_CHO_CTRL)|1<<4|1<<6,PWM_CHO_CTRL);         //Gating the Special Clock for PWM0 and enable it
```

Assuming that PWM0 output two 1ms high pulses and High Level and 2 pre-scale will be selected in the following instance.

```
writel((readl(PH_CFG0)&~(7<<24))|(2<<24),PH_CFG0);           //Select PWM0 output
while(readl(PWM_CHO_CTRL)>>28&1);                             //wait for period register ready to write if it is busy
writel(0x00002ee0,PWM_CHO_PERIOD);                             //Configure PWM Channel 0 Period Register
writel(0<<16|1<<7|1<<0,PWM_CHO_CTRL);                         //Select normal mode,pulse mode and 2 pre-scale
writel(readl(PWM_CHO_CTRL)|1<<4|1<<6,PWM_CHO_CTRL);         //Gating the Special Clock for PWM0 and enable it
writel(readl(PWM_CHO_CTRL)|1<<8,PWM_CHO_CTRL);               //Output a pulse
While(readl(PWM_CHO_CTRL)>>8&1);                               //Wait for pulse output finish
delayms(10);                                                  //delay 10ms
writel(readl(PWM_CHO_CTRL)|1<<8,PWM_CHO_CTRL);               //Output a pulse
While(readl(PWM_CHO_CTRL)>>8&1);                               //Wait for pulse output finish
```

In the upper two instances, the first instance indicates PWM0 outputs 50% duty cycle and 10KHz frequency waveform and the second instance indicates PWM0 output two 1ms high pulse. PWM1/2/3 outputs cycle waveform or pulse as same as PWM0 normal mode as the upper instances.

## 3.12. DMA

### 3.12.1. Overview

DMA controller, also called general DMA controller, is used for the data transfer between devices and memory, memory and memory. The data transfer is controlled by the Device DMA request (DRQ) signal. The DMAC has 16 independent DMA channel, each of which has independent FIFO.

It features:

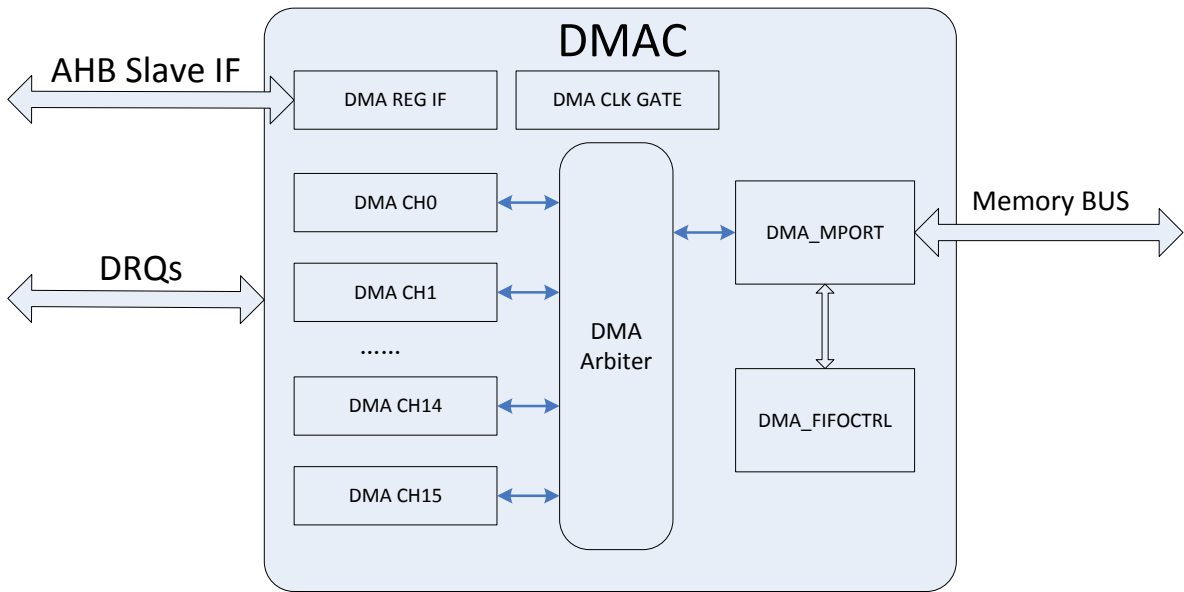
- Single clock synchronization system
- Support AMBA2.0 AHB slave interface
- 16 DMA channels, support Read/Write command priority independent polling mechanism
- Support read/write up to 32 device DRQs
- The data transfer interface supports 64-bit MBUS protocol
- Support script memory (the memory that descriptor located ) and chain transfer
- Support wait mode and handshake mode for DRQ response
- The DRAM type device supports 64-bit burst-6 and burst-16 transfer
- The device type supports 8-bit, 16-bit, 32-bit and 64-bit single transfer
- The single-channel FIFO size is 16x64-bit, and the FIFO of each channel can be merged
- The Memory type device supports non-aligned transfer
- Support trustzone, support DMA channel independent secure mode configuration
- Script memory and device space support 34-bit address

### 3.12.2. Signal Description

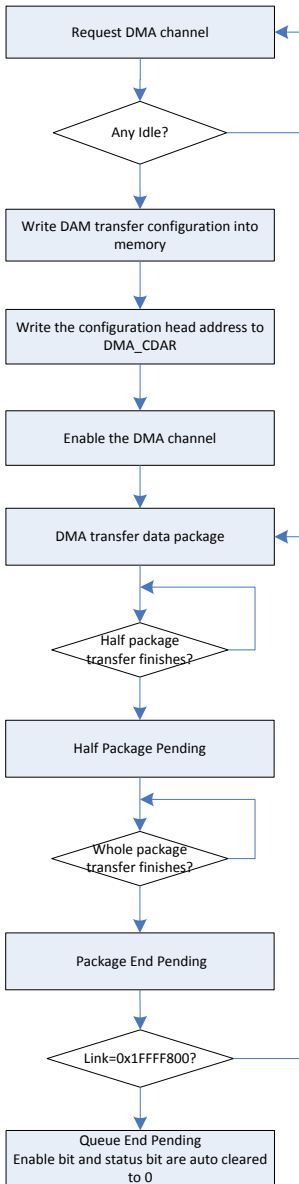
Signal Name	Type	Description
<b>HCLK</b>	Input	DMA main clock input
<b>RST_N</b>	Input	Reset input, async signal, low valid

### 3.12.3. Functionalities Description

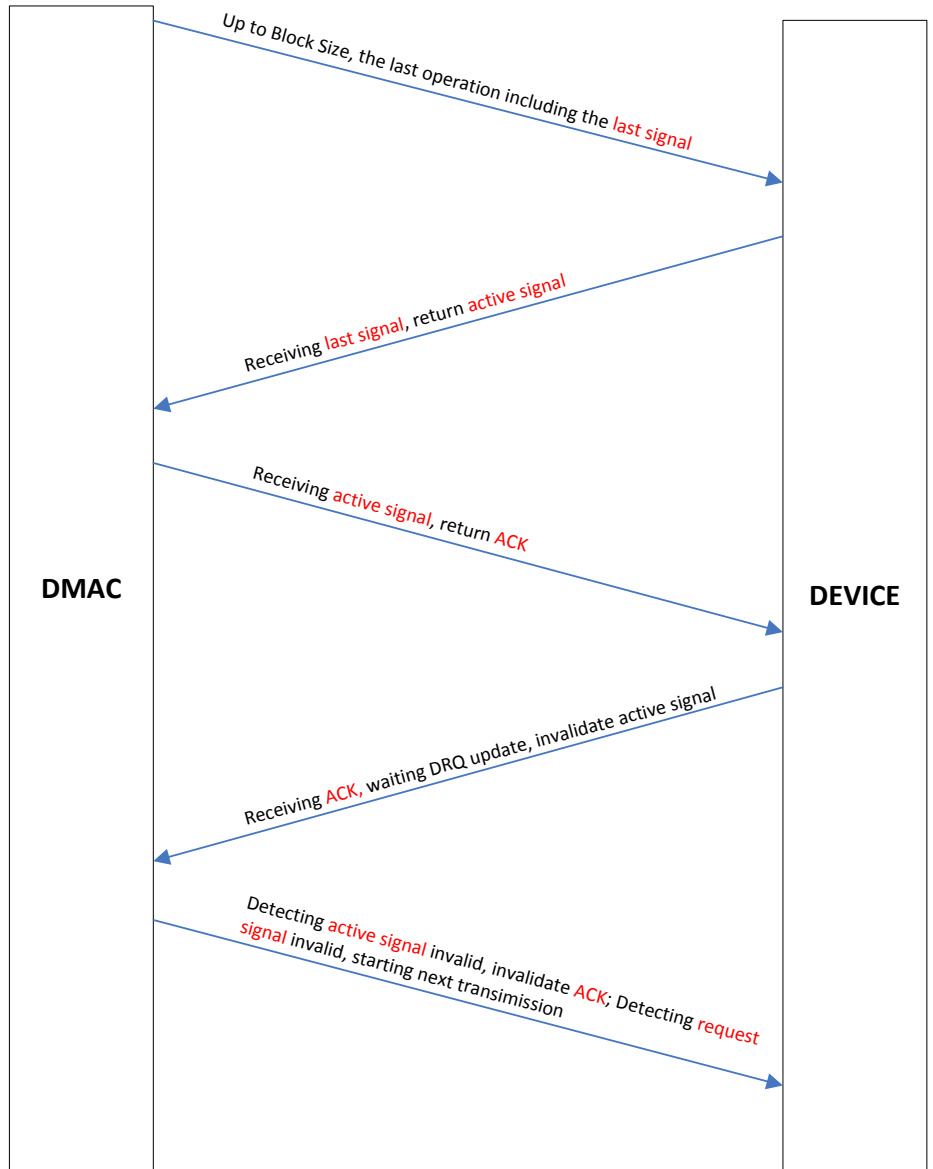
#### 3.12.3.1. Block Diagram



DMA Top Block Diagram



DMA transfer process



DMA Handshake Mode

### 3.12.3.2. DRQ Type and Corresponding Relation

Source DRQ Type		Destination DRQ Type	
Port NO.	Module Name	Port NO.	Module Name
Port 0	Memory	Port 0	Memory
Port 1	/	Port 1	/
Port 2	/	Port 2	/
Port 3	/	Port 3	/
Port 4	/	Port 4	/
Port 6	UART0-RX	Port 6	UART0-TX
Port 7	UART1-RX	Port 7	UART1-TX
Port 8	UART2-RX	Port 8	UART2-TX
Port 9	UART3-RX	Port 9	UART3-TX
Port 10	UART4-RX	Port 10	UART4-TX
Port 15	/	Port 15	
Port 18		Port 18	
Port 22	UART5-RX	Port 22	UART5-TX
Port 23	SPI0_RX	Port 23	SPI0_TX
Port 24	SPI1_RX	Port 24	SPI1_TX
Port 25	SPI2_RX	Port 25	SPI2_TX
Port 26	SPI3_RX	Port 26	SPI3_TX

**DMA DRQ Table**

**Note:**

- SRAM or DRAM DRQ signal is always high.

### 3.12.3.3. DMA Descriptor

The DMA descriptor is the configuration transferred by DMA. Since DMA supports chain-mode multi-packet transfer, the descriptor may contain configuration information of more than one packets. The configuration of each packet consists of six words in fixed order, including Configuration, Source Address, Destination Address, Byte Counter, Parameter and Link: the Configuration is used to configure the DRQ type, r/w mode, data width, data block size of both transfer ends; Source Address is used to configure the source address; Destination Address is used to configure the destination address; Byte Counter is used to configure the data size of a data packet; Parameter is used to configure the interval between each data block; when the link value is 0x1FFFF800, DMAC data transfer will stop after the current data transfer finishes, otherwise, DMAC will take the value of link as the address of the next data packet to be transferred, i.e. the chain-mode multi-packet transfer.

The DMA descriptor can be located in SRAM or DRAM, and its address should be word-aligned. Before the enable of DMA, the descriptor address should be written to **DMA Channel Descriptor Address Register**. After the DMA enable,

DMAC will automatically go to this address to get information of descriptor, analyze it, and then start the data transfer, in the meanwhile, the link value will be displayed in **DMA Channel Descriptor Address Register**. After the data transfer starts, the configuration data will be transferred in Read-Write cycle till the transfer finishes. After the transfer, DMA channel will go back to its initial state, automatically disable the channel enable signal, and the busy status bit will be auto cleared to zero. Please refer to the DMA transfer process diagram for more details.

#### 3.12.3.4. Interrupt

When related interrupts are enabled, the DMA will generate half package interrupt after half package transfer finishes, generate package end interrupt after the whole package transfer finishes, and generate queue end interrupt after the whole chain data package transfer finishes. Notice that when the CPU does not respond to the interrupts timely, or two DMA interrupts are very closed generated, the later interrupt may override the former one.

#### 3.12.3.5. Security

The DMAC supports system trustzone, and supports DMA channel secure mode configuration, in which case each channel property is secure by default. When the system Trustzone function is enabled, the DMAC property is secure by default, that is only secure access is allowable, and all non-secure accesses are invalid. The DMAC security can be configured in SMTA module.

When a DMA channel is configured as non-secure, this channel can only access non-secure memory, and that means, DMA cannot write data to secure memory, and data read from secure memory will be 0.

#### 3.12.3.6. Clock gating

The DMAC clock gating module is used to generate clocks for each sub-module of DMAC and some circuits, including channel clock gating and public clock gating.

For the channel clock gating, when the system visits current DMA channel registers, the DMA clock will auto enabled if the DMA channel is enabled; when the DMA transfer ends, or register access ends, the DMA channel clock will auto disable after 16 HCLKs, and clocks of related channel control, FIFO control circuits will be disabled as well;

For the public clock gating, when all DMA channels are enabled, the public circuit clock will be auto disabled, including clocks of FIFO control module public circuit, MPORT module and related memory bus.

All the functions stated above can be enabled/disabled by software, and reference can be made to **DMA Clock Gating Register**.

#### 3.12.3.7. Transmission mode

The DMAC supports two data transfer modes: wait mode and shakehand mode.

In wait mode, the device request signal after entering DMAC, will be changed into internal DRQ signal by Block and Wait counter; in handshake mode, the DMAC will communicate with device with DMA last, Active and ACK signals. More details are provided in the DMA Device Management section.



### 3.12.3.8. Un-alignment

The DMAC supports address alignment of non-IO devices, that is, when the start address of non-IO device is not 32-byte aligned, DMA will first align the address to 32-byte with burst transfer. This function helps to improve the DRAM access efficiency.

Address alignment is not supported for IO devices, so make sure that the bit width of IO devices matches the address offset.

### 3.12.3.9. Beyond 4G space

The DMAC supports read and write of 4G~16G address space.

When the storage address space of DMA descriptor information exceeds 4G, then the descriptor address bit 0 accessed by DMAC indicates the 32-bit of the address, bit 1 indicates the 33-bit. For example, when the descriptor address accessed is 0x00000001, that means the descriptor information is located at 0x100000000.

When the source address space or destination address space of DMA configuration exceeds 4G, the descriptor Parameters need to be configured. If the source address is 0x140000000, the source address of the descriptor should be configured as 0x40000000, and the bit [17:16] of Parameter should be configured as 0x1; similarly, when the destination address is 0x356314858, the destination address of the descriptor should be configured as 0x56314858, and the Parameter bit [19:18] should be configured as 0x3.

### 3.12.3.10. Clock and reset

#### 3.12.3.10.1. Clocking

The DMAC clock is same with AHB1 Clock. Related gating bit on AHB1 bus should be enabled before visiting the DMAC registers, by writing the 24-bit of 0x06000400+0x184 to 1.

#### 3.12.3.10.2. Reset

The DMAC reset input signal is asynchronous with HCLK, and it's low valid by default. The Reset signal should be invalidated before visiting the DMAC registers, by writing 24-bit of 0x06000400+0x1A4 to 1. It's recommend to first invalidate the reset signal, and then enable the AHB1 gating bit.

### 3.12.4. Operation Principle

#### 3.12.4.1. DMA Device Management

From the port configuration of DMA descriptor, DMA can choose the right device DRQ to drive DMA transfer, and the source address and destination address of the DMA descriptor will determine the data transfer direction.

If the start address of DMA configuration points to some device (must be memory type device), DMA will obtain the DMA descriptor through MBUS; if the DMA source address or destination address points to some device, DMA will Read or write the corresponding device through MBUS. Notice that for single descriptor, DMA will not judge the address, but will visit orderly the device based on the descriptor start addresss sequence.

DMA can differentiate the DMA request source based on the port configuration of descriptors. The port information is only used to define the source of DMA DRQ, and has no business with the DMA transfer destination. If an inexistent devices is pointed due to incorrect device configuration ,DMA will always take its device request as valid.

The request signal of DMA device is used to control the operation of DMA device, such as the DMA transfer start time, stop time, etc. for memory type devices, their request signals are always valid, so the request mechanism of DMAC has no effects on Memory devices.

The DMA request can be used in waiting mode and handshake mode.

### 3.12.5. DMA Register List

Module Name	Base Address
DMAC	0x00802000

Register Name	Offset	Description
DMA_IRQ_ENALBE_REG0	0x0000	DMA IRQ Enable Register 0
DMA_IRQ_ENALBE_REG1	0x0004	DMA IRQ Enable Register 1
DMA_IRQ_PEND_REG0	0x0010	DMA IRQ Pending Register 0
DMA_IRQ_PEND_REG1	0x0014	DMA IRQ Pending Register 1
DMA_SECURE_REG	0x0020	DMA Secure Register
DMA_GATING_REG	0x0028	DMA Gating Register
DMA_STATUS_REG	0x0030	DMA Status Register
DMA_ENABLE_REG	0x0100+N*0x40+0x00 (N=0~15)	DMA Enable Register
DMA_PAUSE_REG	0x0100+N*0x40+0x04 (N=0~15)	DMA Pause Register
DMA_DESCPT_ADDR_REG	0x0100+N*0x40+0x08 (N=0~15)	DMA Descriptor Address Register
DMA_CFG_REG	0x0100+N*0x40+0x0C (N=0~15)	DMA Configuration Register
DMA_CUR_SRCADDR_REG	0x0100+N*0x40+0x10 (N=0~15)	DMA Current Source Address Register
DMA_CUR_DSTADDR_REG	0x0100+N*0x40+0x14 (N=0~15)	DMA Current Destination Address Register
DMA_BCNT_LEFT_REG	0x0100+N*0x40+0x18 (N=0~15)	DMA Byte Counter Left Register
DMA_PARAMETER_REG	0x0100+N*0x40+0x1C (N=0~15)	DMA Parameter Register
DMA_MODE_REG	0x0100+N*0x40+0x28 (N=0~15)	DMA Mode Register
DMA_FDESCPT_ADDR_REG	0x0100+N*0x40+0x2C (N=0~15)	DMA Former Descriptor Address Register
DMA_PKG_NUM_REG	0x0100+N*0x40+0x30 (N=0~15)	DMA Package Number Register

### 3.12.6. DMA Register Description

#### 3.12.6.1. DMA IRQ Enable Register 0 (Default: 0x00000000)

Offset: 0x0000			Register Name: <b>DMA_IRQ_ENALBE_REG0</b>
Bit	R/W	Default/Hex	Description
$4*n + 3$	/	/	/
$4*n + 2$	R/W	0x0	DMA_QUEUE_IRQ_ENABLE. DMA channel n queue end interrupt enable. 0: Disable queue end interrupt. 1: Enable queue end interrupt.
$4*n + 1$	R/W	0x0	DMA_PKG_IRQ_ENABLE. DMA channel n package end interrupt enable. 0: Disable package end interrupt. 1: Enable package end interrupt.
$4*n + 0$	R/W	0x0	DMA_HALF_IRQ_ENABLE. DMA channel n half package interrupt enable. 0: Disable half package interrupt. 1: Enable half package interrupt.

**Note:**  $n = 0\sim7$ .

#### 3.12.6.2. DMA IRQ Enable Register 1 (Default: 0x00000000)

Offset: 0x0004			Register Name: <b>DMA_IRQ_ENALBE_REG1</b>
Bit	R/W	Default/Hex	Description
$4*n + 3$	/	/	/
$4*n + 2$	R/W	0x0	DMA_QUEUE_IRQ_ENABLE. DMA channel (n+8) queue end interrupt enable. 0: Disable queue end interrupt. 1: Enable queue end interrupt.
$4*n + 1$	R/W	0x0	DMA_PKG_IRQ_ENABLE. DMA channel (n+8) package end interrupt enable. 0: Disable package end interrupt. 1: Enable package end interrupt.
$4*n + 0$	R/W	0x0	DMA_HALF_IRQ_ENABLE. DMA channel (n+8) half package interrupt enable. 0: Disable half package interrupt. 1: Enable half package interrupt.

**Note:**  $n = 0\sim7$ .

**3.12.6.3. DMA IRQ Pending Register 0 (Default: 0x00000000)**

Offset: 0x0010			Register Name: <b>DMA_IRQ_PEND_REG0</b>
Bit	R/W	Default/Hex	Description
$4*n + 3$	/	/	/
$4*n + 2$	R/W	0x0	DMA_QUEUE_IRQ_PEND. DMA channel n queue end interrupt pending. Set 1 to clear it. 0: No effect. 1: Half queue end pending.
$4*n + 1$	R/W	0x0	DMA_PKG_IRQ_PEND. DMA channel n package end interrupt pending. Set 1 to clear it. 0: No effect. 1: Package end interrupt pending.
$4*n + 0$	R/W	0x0	DMA_HALF_IRQ_PEND. DMA channel n half package interrupt pending. Set 1 to clear it. 0: No effect. 1: Half package interrupt pending.

**Note:** n = 0~7.

**3.12.6.4. DMA IRQ Pending Register 1 (Default: 0x00000000)**

Offset: 0x0014			Register Name: <b>DMA_IRQ_PEND_REG1</b>
Bit	R/W	Default/Hex	Description
$4*n + 3$	/	/	/
$4*n + 2$	R/W	0x0	DMA_QUEUE_IRQ_PEND. DMA channel (n+8) queue end interrupt pending. Set 1 to clear it. 0: No effect. 1: Half queue end pending.
$4*n + 1$	R/W	0x0	DMA_PKG_IRQ_PEND. DMA channel (n+8) package end interrupt pending. Set 1 to clear it. 0: No effect. 1: Package end interrupt pending.
$4*n + 0$	R/W	0x0	DMA_HALF_IRQ_PEND. DMA channel (n+8) half package interrupt pending. Set 1 to clear it. 0: No effect. 1: Half package interrupt pending.

**Note:** n = 0~7.

**3.12.6.5. DMA Secure Register (Default: 0x00000000)**

Offset: 0x0020			Register Name: <b>DMA_SECURE_REG</b>
Bit	R/W	Default/Hex	Description
31:16	/	/	/
n	R/W	0x0	DMA_SECURE. DMA channel n security (n = 0~15): 0: Secure. 1: Non-secure.

**3.12.6.6. DMA Gating Register (Default: 0x00000000)**

Offset: 0x0028			Register Name: <b>DMA_GATING_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	DMA_CLK_GATING1. This bit is used for Arbiter, MPORT and FIFOCTRL modules clock gating. 0: Enable clock gating. 1: Disable clock gating.
0	R/W	0x0	DMA_CLK_GATING0. This bit is used for DMA channels clock gating. 0: Enable clock gating. 1: Disable clock gating.

**Note:**Please refer to the clock gating section for details.

**3.12.6.7. DMA Status Register (Default: 0x00000000)**

Offset: 0x0030			Register Name: <b>DMA_STATUS_REG</b>
Bit	R/W	Default/Hex	Description
31:16	/	/	/
n	R/W	0x0	DMA_STATUS. DMA channel n status (n = 0~15): 0: Idle. 1: Busy. When a DMA channel is enabled, then it would be busy, even if the channel has been paused.

### 3.12.6.8. DMA Enable Register (Default: 0xC0000000)

Offset: 0x100+N*0x40+0x00 (N=0~15)			Register Name: <b>DMA_ENABLE_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
30	R/W	0x0	DMA_ENABLE. DMA channel enable: 0: Disable. 1: Enable.

### 3.12.6.9. DMA Pause Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x04 (N=0~15)			Register Name: <b>DMA_PAUSE_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
30	R/W	0x0	DMA_PAUSE. Pausing transmission: 0: Resume transmission. 1: Pause transmission.

### 3.12.6.10. DMA Descriptor Address Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x08 (N=0~15)			Register Name: <b>DMA_DESCPT_ADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:8	R/W	0x0	DMA_DESCPT_ADDR. Before enabling the DMA, you need to write word-aligned valid descriptor address.
1:0	R/W	0x0	DMA_DESCPT_ADDR_HIGH. Both bits would be used to storing [33:32] bits of the descriptor address, if it is necessary. Otherwise, they should be 0.

**Note:**When the DMA finishes all data transfer, the register value will be 0x1FFF800.

### 3.12.6.11. DMA Configuration Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x0C (N=0~15)			Register Name: <b>DMA_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:30	/	/	/
26:25	RO	0x0	DMA_DST_DATA_WIDTH. 00: 8-bit.

			01: 16-bit. 10: 32-bit. 11: 64-bit.
24	/	/	/
23:22	RO	0x0	DMA_DST_BLOCK_SIZE. 00: 1. 01: 4. 10: 8. 11: 16.
21	RO	0x0	DMA_DST_DRQ_TYPE. DMA destination address mode. 0: Linear mode. 1: IO mode.
20:16	RO	0x0	DMA_DST_DRQ_TYPE. Refer to the <b>DMA DRQ Table</b> .
15:11	/	/	/
10:9	RO	0x0	DMA_SRC_DATA_WIDTH. 00: 8-bit. 01: 16-bit. 10: 32-bit. 11: 64-bit.
8	/	/	/
7:6	RO	0x0	DMA_SRC_BLOCK_SIZE. 00: 1. 01: 4. 10: 8. 11: 16.
5	RO	0x0	DMA_SRC_DRQ_TYPE. DMA source address mode. 0: Linear mode. 1: IO mode.
4:0	RO	0x0	DMA_SRC_DRQ_TYPE. Refer to the <b>DMA DRQ Table</b> .

### 3.12.6.12. DMA Current Source Address Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x10 (N=0~15)			Register Name: <b>DMA_CUR_SRCADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DMA_CUR_SRC_ADDR. DMA current source address.



**3.12.6.13. DMA Current Destination Register (Default: 0x00000000)**

Offset: 0x100+N*0x40+0x14 (N=0~15)			Register Name: <b>DMA_CUR_DSTADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DMA_CUR_DST_ADDR. DMA current destination address.

**3.12.6.14. DMA Byte Counter Left Register (Default: 0x00000000)**

Offset: 0x100+N*0x40+0x18 (N=0~15)			Register Name: <b>DMA_BCNT_LEFT_REG</b>
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24:0	RO	0x0	DMA_BCNT_LEFT. DMA byte counter left.

**3.12.6.15. DMA Parameter Register (Default: 0x00000000)**

Offset: 0x100+N*0x40+0x1C (N=0~15)			Register Name: <b>DMA_PARAMETER_REG</b>
Bit	R/W	Default/Hex	Description
31:20	/	/	/
19:18	RO	0x0	DMA_DST_HBIT_ADDR.
17:16	RO	0x0	DMA_SRC_HBIT_ADDR.
15:8	/	/	/
7:0	RO	0x0	WAIT_CLK_CYCLE. Wait clock cycle.

**3.12.6.16. DMA Mode Register (Default: 0x00000000)**

Offset: 0x100+N*0x40+0x28 (N=0~15)			Register Name: <b>DMA_MODE_REG</b>
Bit	R/W	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	DMA_DST_MODE.  0: Wait mode. 1: Handshake mode.
2	R/W	0x0	DMA_SRC_MODE.  0: Wait mode. 1: Handshake mode.

1:0	/	/	/
-----	---	---	---

### 3.12.6.17. DMA Former Descriptor Address Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x2C (N=0~15)			Register Name: <b>DMA_FDESCPT_ADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DMA_FDESCPT_ADDR.  This register is used to record the head address of former descriptor.

### 3.12.6.18. DMA Package Number Register (Default: 0x00000000)

Offset: 0x100+N*0x40+0x30 (N=0~15)			Register Name: <b>DMA_PKG_NUM_REG</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DMA_PKG_NUM.  This register is used to record the number of transferred data packages .

### 3.12.7. Programming Guidelines

- The transfer bit width of IO device should matches its start address.
- For devices supporting non-integer word operation, it should be capable of read DMA command based on its FIFO bit width instead of the command bit width;
- When the DMA data transfer is paused before finishes, it's equivalent to the DRQ invalidity.

- DMA Application Example :

```
writel(0x00000000, mem_address + 0x00);           //configuration, mem_address must be word-aligned
writel(0x00001000, mem_address + 0x04);           //configure the source address
writel(0x20000000, mem_address + 0x08);           //configure the destination address
writel(0x00000020, mem_address + 0x0C);           //configure the data package size
writel(0x00000000, mem_address + 0x10);           //configure the Parameter
writel(0x1FFFF800, mem_address + 0x14); //configure the Link, or finish flag, or start address of the next descriptor
writel(mem_address, 0x00802000 + 0x100 + 0x08); // write the start address of DMA0 descriptor
do{
    If(mem_address == readl(0x00802000 + 0x100 + 0x08));
        break;
}while(1); //make sure the write operation is valid
writel(0x00000001, 0x00802000 + 0x100 + 0x00); //enable the DMA 0 transfer
```

- **Add data packages during DMA transfer**

More data packages can be added during the DMA transfer. Notice that:

- if more data packages are to be added to the DMA transfer, first of all, make sure that whether the DMA has got the descriptor of the last data package back: if yes, you will have to restart the DMA transfer; if not, you can modify the stop flag (0x1FFFF800) of the last descriptor to the start address of the next data package's descriptor;
- when the value of **DMA Descriptor Address Register** is 0x1FFFF800, that means the DMA has got the descriptor of the last data package back; after the transfer of this package, the DMA transfer will end;
- you can check whether the data is successfully modified by reading the value of **DMA Descriptor Address Register**: **if it is not** 0x1FFFF800, that means the data package is successfully added; if it is 0x1FFFF800, the package may not be added, as DMA may has taken back the descriptor of the last data package during the add of data packages. In this situation, you can try to read the value of **DMA Current Source Address Register** or **DMA Current Destination Address Register**: if the increasing memory address matches the data package information, the it is successfully added.
- To increase the odd of success, it is recommended to add data package before the half package interrupt of the penultimate data package.

### 3.13. GIC

#### 3.13.1. Interrupt Source

Interrupt source number	Interrupt source name
0	SGI 0
1	SGI 1
2	SGI 2
3	SGI 3
4	SGI 4
5	SGI 5
6	SGI 6
7	SGI 7
8	SGI 8
9	SGI 9
10	SGI 10
11	SGI 11
12	SGI 12
13	SGI 13
14	SGI 14
15	SGI 15
16	PPI 0
17	PPI 1
18	PPI 2
19	PPI 3
20	PPI 4
21	PPI 5
22	PPI 6
23	PPI 7
24	PPI 8
25	PPI 9
26	PPI 10
27	PPI 11
28	PPI 12
29	PPI 13
30	PPI 14
31	PPI 15

32	UART 0
33	UART 1
34	UART 2
35	UART 3
36	UART 4
37	UART 5
38	TWI 0
39	TWI 1
40	TWI 2
41	TWI 3
42	TWI 4
43	PA_EINT
47	PB_EINT
48	PE_EINT
49	PG_EINT
50	Timer 0
51	Timer 1
52	Timer 2
53	Timer 3
54	Timer 4
55	Timer 5
56	Watchdog
57	
58	
59	
60	
61	
62	KEYADC
63	
64	NMI
65	/
66	/
67	/
68	/
69	/
70	/
71	/
72	
73	
74	
75	/
76	/

77	/
78	/
79	/
80	/
81	/
82	DMA
83	HS Timer 0
84	HS Timer 1
85	HS Timer 2
86	HS Timer 3
87	HS Timer 4
88	SMC
89	
90	VE
91	
92	SD/MMC 0
93	SD/MMC 1
94	SD/MMC 2
95	SD/MMC 3
96	
97	SPI 0
98	SPI 1
99	SPI 2
100	SPI 3
101	
102	NAND0
103	USB –DRD
104	USB-EHCI0
105	USB-OHCI0
106	USB-EHCI1
107	/
108	USB-EHCI2
109	USB-OHCI2
110	
111	
112	SS
113	TS
114	EMAC
115	MP
116	CSI-0
117	CSI-1
118	LCD-0

119	LCD-1
120	HDMI
121	MIPI DSI
122	MIPI CSI
123	DRC 0/1
124	DEU 0/1
125	DE_FE0
126	DE_FE1
127	DE_BE0
128	DE_BE1
129	GPU
130	GPU PWR
131	
132	
133	
134	
135	
136	
137	
138	
139	
140	FD
141	GPADC
142	
143	
144	
145	
146	
147	THS
148	DE_BE2
159	DE_FE2
150	eDP
151	
152	PH_EINT
153	
154	CSI0_CCI
155	CSI1_CCI
156	CCI_400
157	
158	
159	
160	

161	
162	
163	
164	
165	
166	
167	
168	
169	
170	
171	
172	
173	
174	
175	
176	
177	
178	
179	



## 3.14. Message Box

### 3.14.1. Overview

Message Box provides an MSGBox-interrupt mechanism for on-chip processors intercommunication. It allows a processor transmit messages to the other one or receive messages from the other through a series of Message Queues, each of which is a four 32-bits depth FIFO. An intercommunication channel could be established by configuring Message Box registers and it works under MSGBox interrupt mechanism.

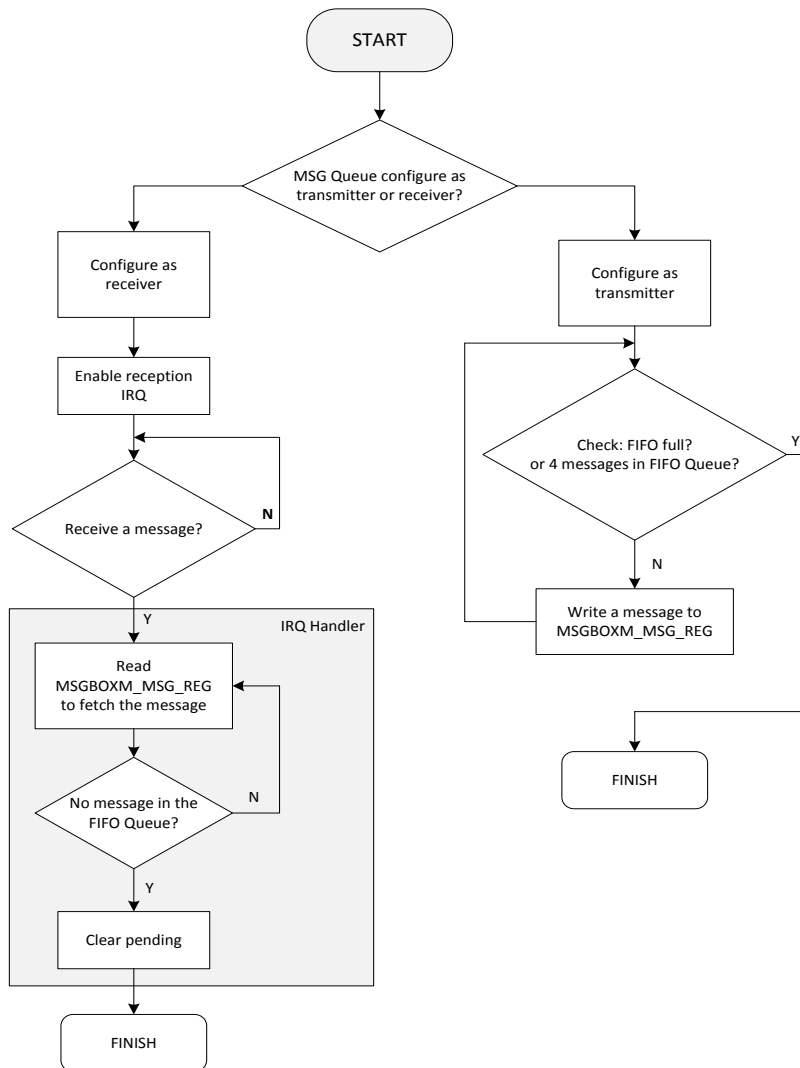
The Message Box includes the following features:

- Two users for Message Box instance(User0 for CPUS and User1 for C0-CPUX/C1-CPUX)
- Eight Message Queues and each of Queues is a four 32-bits depth FIFO for establishing intercommunication channel
- Each of Queues could be configured as transmitter or receiver for user
- Message reception and queue-not-full notification interrupt mechanism

### 3.14.2. Functionalities Description

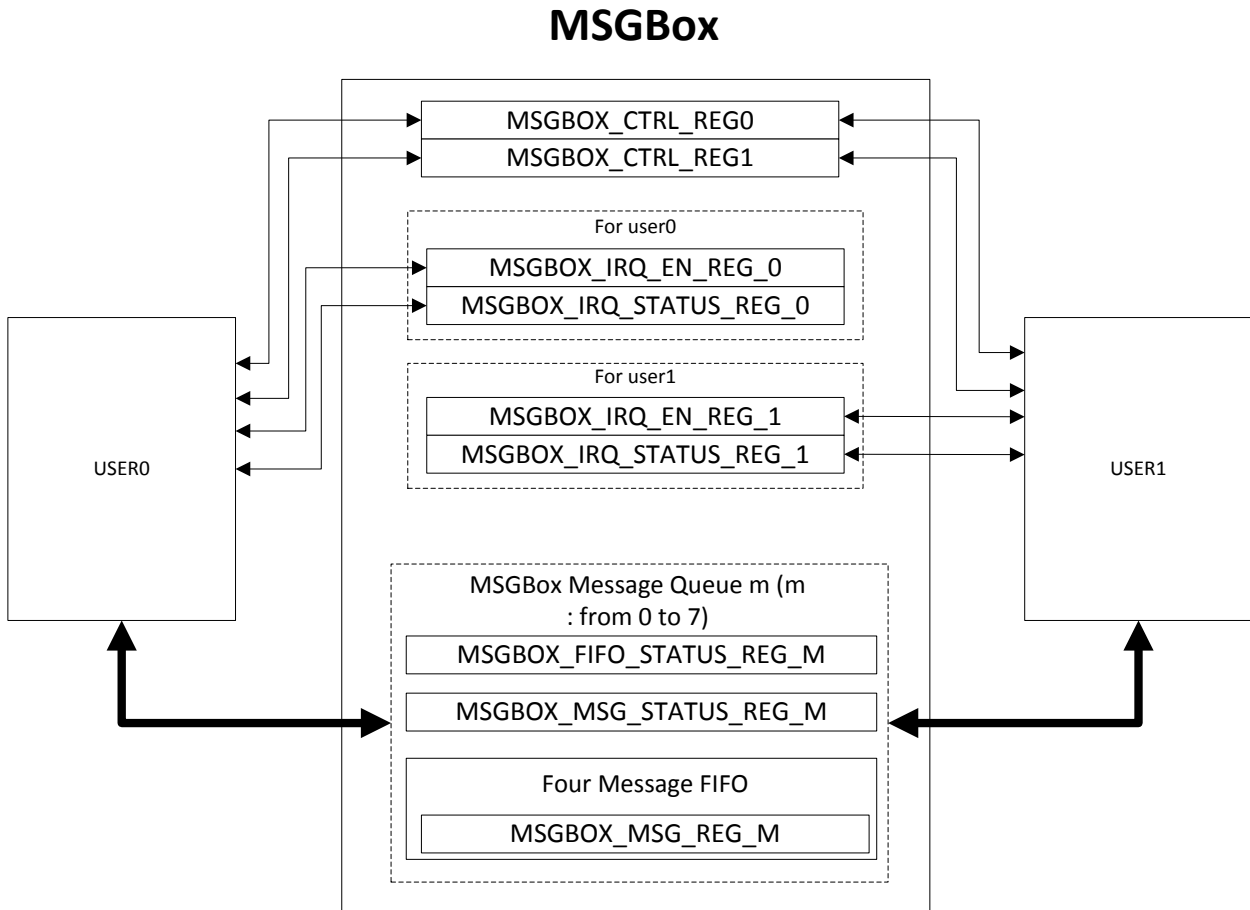
#### 3.14.2.1. Typical Applications

Message Box is typically designed for making the on-chip processors interconnection be true. It could establish an interconnection channel between processors by configuring a set of Message Box registers. Each of Message Queues is bidirectional for users, that means, while a message queue is configured as a receiver for a user, it is a transmitter for the interconnectable user beside. If a processor would like to interconnect with the other processor, it should configure one or more Message Queues firstly. Although Message Box provides two interrupt mechanism to notice user to transmit or receive messages, the way check out queue FIFO full status is usually adopted before transmitting a message and receiving a message is still depended on the reception notification. The Message Box is usually applied as the below flow chart:



Message Box Typical Application Chart

### 3.14.2.2. Functional Block Diagram



Message Box Functional Block Diagram

Message Box supports a set of registers for a processor to establish an interconnection channel with the others. The processor determines message queue numbers for interconnection and the used queues to be transmitter or receiver for itself and the interconnectible one. Every queue has a *MSGBox FIFO Status Register* for processor to check out queue FIFO full status and a *MSGBox Message Status Register* for processor to check out message numbers in queue FIFO. Otherwise, every queue has a corresponding IRQ status bit and a corresponding IRQ enable bit, which used for requesting an interrupt.

### 3.14.3. Operation Principle

#### 3.14.3.1. Message Box clock gating and software reset

By default the Message Box clock gating is mask. When it is necessary to use Message Box, it's clock gating should be open in *AHB1 Module Clock Gating Register* and then de-assert the software reset in *AHB1 Module Software Reset Register* on CCU module. If it is no need to use Message Box, both the gating bit and software reset bit should set 0.

### 3.14.3.2. Message Queue Assignment

When a processor needs to transmit or receive a message from the other one, it should configure the Message Queue assignment for the other one and itself. *MSGBOX\_CTRL\_REG0* and *MSGBOX\_CTRL\_REG1* hold the eight Message Queues assignment. For an instance, *RECEPTION\_MQ0* bit is set to 0 and *TRANSMIT\_MQ0* bit is set to 1, which means, user1 transmits messages and user0 receives them. Or *RECEPTION\_MQ0* bit and *TRANSMIT\_MQ0* bit are both set to 0, which means user0 transmits messages to itself.

### 3.14.3.3. Interrupt request

Message Box provides Message reception and queue-not-full notification interrupt mechanism. For a Message Queue configured as transmitter for a user, this queue transmit pending bit will always be set to 1 of this user if it is not full. For a Message Queue configured as receiver for a user, this queue reception pending bit will be set to 1 for this user only if it receives a new message. For example, Message Queue0 is configured as a transmitter for user0 and a receiver for user1. The thing Message Queue0 is not full always makes *TRANSMIT\_MQ0\_IRQ\_PEND* bit set to 1. If *TRANSMIT\_MQ0\_IRQ\_EN* bit is set to 1, user0 will request a queue-not-full interrupt. When Message Queue0 has received a new message, *RECEPTION\_MQ0\_IRQ\_PEND* bit would be set to 1 and user1 will request a new message reception interrupt if *RECEPTION\_MQ0\_IRQ\_EN* bit is set to 1. *MSGBox IRQ Status Register u (u=0, 1)* hold the IRQ status for user0 and user1. *MSGBox IRQ Enable Register u (u=0, 1)* determine whether the user could request the interrupt or not.

### 3.14.3.4. Transmit and receive messages

Every Message Queue has a couple of private registers for query: *MSGBox Message Status Register* and *MSGBox FIFO Status Register* and a store register bridged to Message Queue FIFO: *MSGBox Message Queue Register*. *MSGBox Message Status Register* records present message number in the Message Queue. *MSGBox FIFO Status Register* indicates whether the Message Queue is full obviously. *MSGBox Message Queue Register* stores the next to be read message of the message FIFO queue or the message to be written into the queue FIFO. The thing that queue is not full usually indicates that you could write messages into the queue FIFO and that there is one or more message in the queue FIFO indicates that you could read messages from the queue FIFO.

Writing a message into the queue FIFO realizes a transmission and reading a message makes a reception. You could transmit messages by writing messages to *MSGBox Message Queue Register* continuously or receive messages by reading *MSGBox Message Queue Register* continuously. The wiring or reading operation could be continuous means it's no need to make a delay between operations.

### 3.14.4. Message Box Register List

Module Name	Base Address
MSGBOX	0x00803000

Register Name	Offset	Description
MSGBOX_CTRL_REG0	0x0000	Message Queue Attribute Control Register 0
MSGBOX_CTRL_REG1	0x0004	Message Queue Attribute Control Register 1
MSGBOXU_IRQ_EN_REG	0x0040+n*0x20	IRQ Enable For User n (n=0,1)
MSGBOXU_IRQ_STATUS_REG	0x0050+n*0x20	IRQ Status For User n (n=0,1)
MSGBOXM_FIFO_STATUS_REG	0x0100+N*0x4	FIFO Status For Message Queue N(N = 0~7)
MSGBOXM_MSG_STATUS_REG	0x0140+N*0x4	Message Status For Message Queue N(N=0~7)
MSGBOXM_MSG_REG	0x0180+N*0x4	Message Register For Message Queue N(N=0~7)

### 3.14.5. Message Box Register Description

#### 3.14.5.1. MSGBox Control Register 0(Default: 0x10101010)

Offset: 0x00			Register Name: MSGBOX_CTRL_REG0
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R/W	0x1	TRANSMIT_MQ3. Message Queue 3 is a Transmitter of user u. 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ3. Message Queue 3 is a Receiver of user u. 0: user0 1: user1
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ2. Message Queue 2 is a Transmitter of user u. 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ2. Message Queue 2 is a Receiver of user u. 0: user0 1: user1
15:13	/	/	/
12	R/W	0x1	TRANSMIT_MQ1 Message Queue 1 is a Transmitter of user u. 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ1. Message Queue 1 is a Receiver of user u. 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ0. Message Queue 0 is a Transmitter of user u. 0: user0 1: user1

3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ0. Message Queue 0 is a Receiver of user u. 0: user0 1: user1

### 3.14.5.2. MSGBox Control Register 1(Default : 0x10101010)

Offset: 0x04			Register Name: MSGBOX_CTRL_REG1
Bit	R/W	Default/Hex	Description
31:29	/	/	/
28	R/W	0x1	TRANSMIT_MQ7. Message Queue 7 is a Transmitter of user u. 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ7. Message Queue 7 is a Receiver of user u. 0: user0 1: user1
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ6. Message Queue 6 is a Transmitter of user u. 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ6. Message Queue 6 is a Receiver of user u. 0: user0 1: user1
15:13	/	/	/
12	R/W	0x1	TRANSMIT_MQ5 Message Queue 5 is a Transmitter of user u. 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ5. Message Queue 5 is a Receiver of user u. 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ4. Message Queue 4 is a Transmitter of user u.

			0: user0 1: user1
3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ4. Message Queue 4 is a Receiver of user u. 0: user0 1: user1

### 3.14.5.3. MSGBox IRQ Enable Register u(u=0,1)(Default : 0x00000000)

Offset:0x40+N*0x20 (N=0,1)			Register Name: MSGBOXU_IRQ_EN_REG
Bit	R/W	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	TRANSMIT_MQ7_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 7 is not full.)
14	R/W	0x0	RECEPTION_MQ7_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 7 has received a new message.)
13	R/W	0x0	TRANSMIT_MQ6_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 6 is not full.)
12	R/W	0x0	RECEPTION_MQ6_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 6 has received a new message.)
11	R/W	0x0	TRANSMIT_MQ5_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 5 is not full.)
10	R/W	0x0	RECEPTION_MQ5_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 5 has received a new message.)
9	R/W	0x0	TRANSMIT_MQ4_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 4 is not full.)
8	R/W	0x0	RECEPTION_MQ4_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 4 has received a new message.)
7	R/W	0x0	TRANSMIT_MQ3_IRQ_EN. 0: Disable



			1: Enable (It will Notify user u by interrupt when Message Queue 3 is not full.)
6	R/W	0x0	RECEPTION_MQ3_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 3 has received a new message.)
5	R/W	0x0	TRANSMIT_MQ2_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 2 is not full.)
4	R/W	0x0	RECEPTION_MQ2_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 2 has received a new message.)
3	R/W	0x0	TRANSMIT_MQ1_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 1 is not full.)
2	R/W	0x0	RECEPTION_MQ1_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 1 has received a new message.)
1	R/W	0x0	TRANSMIT_MQ0_IRQ_EN. 0: Disable 1: Enable (It will Notify user u by interrupt when Message Queue 0 is not full.)
0	R/W	0x0	RECEPTION_MQ0_IRQ_EN. 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 0 has received a new message.)

#### 3.14.5.4. MSGBox IRQ Status Register u(Default : 0x000AAAA)

Offset:0x50+N*0x20 (N=0,1)			Register Name: MSGBOXU_IRQ_STATUS_REG
Bit	R/W	Default/Hex	Description
31:16	/	/	/
15	R/W	0x1	TRANSMIT_MQ7_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 7 is not full. Set one to this bit will clear it.
14	R/W	0x0	RECEPTION_MQ7_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 7 has received a new message. Set one to this bit will clear it.
13	R/W	0x1	TRANSMIT_MQ6_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 6 is not full.

			Set one to this bit will clear it.
12	R/W	0x0	RECEPTION_MQ6_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 6 has received a new message. Set one to this bit will clear it.
11	R/W	0x1	TRANSMIT_MQ5_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 5 is not full. Set one to this bit will clear it.
10	R/W	0x0	RECEPTION_MQ5_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 5 has received a new message. Set one to this bit will clear it.
9	R/W	0x1	TRANSMIT_MQ4_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 4 is not full. Set one to this bit will clear it.
8	R/W	0x0	RECEPTION_MQ4_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 4 has received a new message. Set one to this bit will clear it.
7	R/W	0x1	TRANSMIT_MQ3_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 3 is not full. Set one to this bit will clear it.
6	R/W	0x0	RECEPTION_MQ3_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 3 has received a new message. Set one to this bit will clear it.
5	R/W	0x1	TRANSMIT_MQ2_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 2 is not full. Set one to this bit will clear it.
4	R/W	0x0	RECEPTION_MQ2_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 2 has received a new message. Set one to this bit will clear it.
3	R/W	0x1	TRANSMIT_MQ1_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 1 is not full. Set one to this bit will clear it.
2	R/W	0x0	RECEPTION_MQ1_IRQ_PEND. 0: No effect,

			1: Pending. This bit will be pending for user u when Message Queue 1 has received a new message. Set one to this bit will clear it.
1	R/W	01	TRANSMIT_MQ0_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 0 is not full. Set one to this bit will clear it.
0	R/W	0x0	RECEPTION_MQ0_IRQ_PEND. 0: No effect, 1: Pending. This bit will be pending for user u when Message Queue 0 has received a new message. Set one to this bit will clear it.

### 3.14.5.5. MSGBox FIFO Status Register m(Default : 0x00000000)

Offset:0x100+N*0x4 (N=0~7)			Register Name: MSGBOXM_FIFO_STATUS_REG
Bit	R/W	Default/Hex	Description
31: 1	/	/	/
0	RO	0x0	FIFO_FULL_FLAG. 0: The Message FIFO queue is not full (space is available), 1: The Message FIFO queue is full. This FIFO status register has the status related to the message queue.

### 3.14.5.6. MSGBox Message Status Register m(Default : 0x00000000)

Offset:0x140+N*0x4 (N=0~7)			Register Name: MSGBOXM_MSG_STATUS_REG
Bit	R/W	Default/Hex	Description
31:3	/	/	/
2:0	RO	0x0	MSG_NUM. Number of unread messages in the message queue. Here, limited to four messages per message queue. 000: There is no message in the message FIFO queue. 001: There is 1 message in the message FIFO queue. 010: There are 2 messages in the message FIFO queue. 011: There are 3 messages in the message FIFO queue. 100: There are 4 messages in the message FIFO queue. 101~111:/

### 3.14.5.7. MSGBox Message Queue Register m(Default : 0x00000000)

Offset:0x180+N*0x4 (N=0~7)			Register Name: MSGBOXM_MSG_REG
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	The message register stores the next to be read message of the message FIFO queue. Reads remove the message from the FIFO queue.

### 3.14.6. Programming Guidelines

Assuming that C0-CPU0 establishes an interconnected channel with CPUS using message queue0 and message queue1. The example is as follow:

CPU0 of Cluster 0

-----  
 Step 1: CPU0 establishes an interconnection channel with CPUS

```
writel(readl(AHB1_GATING_REG)|(1<<21), AHB1_GATING_REG); //open Message Box clock gating
writel(readl(AHB1_RST_REG)|(1<<21), AHB1_RST_REG); //software reset Message Box
writel(0x00000110, MSGBOX_CTRL_REG0); //set CPU0 transmits messages to CPUS through Queue0 and
// set CPUS transmits messages to CPU0 through Queue1
Writel(0x4, MSGBOXU_IRQ_EN_REG1); //enable queue1 reception irq of user1
Writel(0x1, MSGBOXU_IRQ_EN_REG0); //enable queue0 reception irq of user0
```

Step 2: CPU0 transmits 4 messages to CPUS

//before transmitting messages, check out queue FIFO full status or message numbers in queue FIFO.

```
for(i=0;i<4;i++)
```

```
writel(queue0_transmit_buf[i], MSGBOXM_MSG_REG0); //write 4 messages stored in
```

```
// queue0_transmit_buf[i](i=0~4) to queue0
```

step 3: CPU0 waits for CPUS to read all messages in queue0

```
while(readl(MSGBOXM_MSG_STATUS_REG 0)); //CPU0 waits for CPUS to read all messages in queue0
```

Step 4: CPU0 waits for CPUS transmits messages for itself and receive them in queue1 irq handler

```
while(!readl(MSGBOXM_FIFO_STATUS_REG1)); //wait for queue1 turns to full status
```

queue1 irq handler:

```
while(readl(MSGBOXM_MSG_STATUS_REG1) != 0) //read all messages from queue1 and store them in
```

```
{ // queue1_receive_buf[i](i=0~4)
```

```
    rdata=readl(MSGBOXM_MSG_REG 1);
```

```
    writel(rdata,queue0_receive_buf[i]);
```

```
}
```

CPUS

-----  
 Step 1: CPUS waits for CPU0 transmits messages for itself and receive them in queue0 irq handler

```
while(!readl(MSGBOXM_FIFO_STATUS_REG0)); //wait for queue0 turns to full status
```

queue0 irq handler:

```
while(readl(MSGBOXM_MSG_STATUS_REG0) != 0) //read all messages from queue0 and store them in
```

```
{ // queue0_receive_buf[i](i=0~4)
```

```
    rdata=readl(MSGBOXM_MSG_REG 0);
```

```
    writel(rdata,queue0_receive_buf[i]);
```

```
}
```

Step 2: CPUS transmits 4 messages to CPU0

```
for(i=0;i<4;i++)
```

```
writel(queue1_transmit_buf[i], MSGBOXM_MSG_REG1); //write 4 messages stored in
```

```
// queue1_transmit_buf[i](i=0~4) to queue1
```

The result of the upper instance is: the messages in the `queue0_transmit_buf[i](i=0~4)` are transmitted to `queue0_receive_buf[i](i=0~4)` and the messages in the `queue1_transmit_buf[i](i=0~4)` are transmitted to `queue1_receive_buf[i](i=0~4)`.

## 3.15. Spinlock

### 3.15.1. Overview

Spinlock provides hardware assistance for synchronizing the processes running on multiple processors in the device. The Spinlock module implements thirty-two 32-bit spinlocks (or hardware semaphores), which provide an efficient way to perform a lock operation of a device resource using a single read access, thus avoiding the need for a 'read-modify-write' bus transfer that not all the programmable cores are capable of.

Spinlocks are present to solve the need for synchronization and mutual exclusion between heterogeneous processors and those not operating under a single, shared operating system. There is no alternative mechanism to accomplish these operations between processors in separate subsystems. However, Spinlocks do not solve all system synchronization issues. They have limited applicability and should be used with care to implement higher level synchronization protocols.

A spinlock is appropriate for mutual exclusion for access to a shared data structure. It should be used only when:

- 1) The time to hold the lock is predictable and small (for example, a maximum hold time of less than 200 CPU cycles may be acceptable).
- 2) The locking task cannot be preempted, suspended, or interrupted while holding the lock (this would make the hold time large and unpredictable).
- 3) The lock is lightly contended, that is the chance of any other process (or processor) trying to acquire the lock while it is held is small.

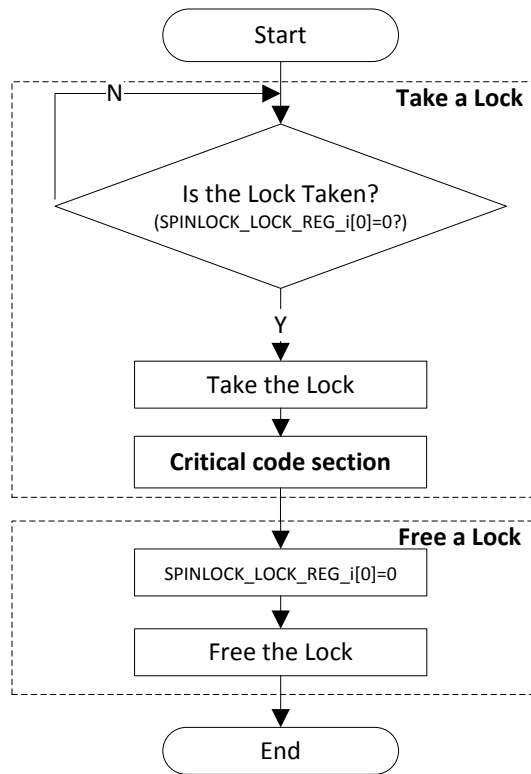
If the conditions are not met, then a spinlock is not a good candidate. One alternative is to use a spinlock for critical section control (engineered to meet the conditions) to implement a higher level semaphore that can support preemption, notification, timeout or other higher level properties.

The Spinlock includes the following features:

- Spinlock module includes 32 spinlocks
- Two kinds of status of lock register: TAKEN and NOT TAKEN

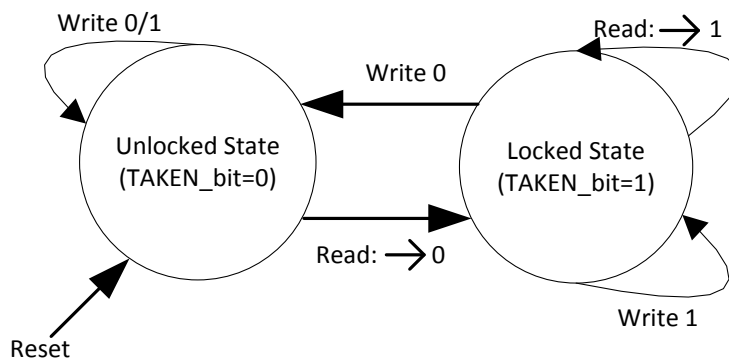
### 3.15.2. Functionalities Description

#### 3.15.2.1. Typical Applications



Spinlock Typical Application Flow Chart

#### 3.15.2.2. Functional Block Diagram



Spinlock Lock Register State Diagram

Every lock register has two kinds of states: TAKEN(locked) or NOT TAKEN(Unlocked). Only read-0-access and write-0-access could change lock register' state and the other accesses has no effect. Just 32-bit reads and writes are supported to access all lock registers.

### 3.15.3. Operation Principle

#### 3.15.3.1. Spinlock clock gating and software reset

Spinlock clock gating should be open before using it. Setting *AHB1 Module Clock Gating Register* bit[22] 1 could activate Spinlock and then de-asserting it's software reset. Setting *AHB1 Module Software Reset Register* bit[22] 1 could de-assert the software reset of Spinlock. If it is no need to use spinlock, both the gating bit and software reset bit should be set 0.

#### 3.15.3.2. Take and free a spinlock

Checking out *SpinLock Register Status* is necessary when a processor would like to take a spinlock. This register stores all 32 lock registers' status: TAKEN or NOT TAKEN(free).

In order to request to take a spinlock, a processor has to do a read-access to the corresponding lock register. If lock register returns 0, the processor takes this spinlock. And if lock register returns 1, the processor must retry.

Writing 0 to a lock register frees the corresponding spinlock. If the lock register is not taken, write-access has no effect. For a taken spinlock, every processor has the privilege to free this spinlock. But it is suggested that the processor which has taken the spinlock free it for strictness.



### 3.15.4. Spinlock Register List

Module Name	Base Address
Spinlock	0x00804000

Register Name	Offset	Description
SPINLOCK_SYSTATUS_REG	0x0000	Spinlock System Status Register
SPINLOCK_STATUS_REG	0x0010	Spinlock Status Register
SPINLOCK_LOCK_REGN	0x100+N*0x4	Spinlock Register N (N=0~31)

### 3.15.5. Spinlock Register Description

#### 3.15.5.1. Spinlock System Status Register (Default: 0x10000000)

Offset: 0x0			Register Name: SPINLOCK_SYSTATUS_REG
Bit	R/W	Default/Hex	Description
31:30	/	/	/
29:28	RO	0x1	LOCKS_NUM. Number of lock registers implemented. 0x1: This instance has 32 lock registers. 0x2: This instance has 64 lock registers. 0x3: This instance has 128 lock registers. 0x4: This instance has 256 lock registers.
27:16	/	/	/
15:9	/	/	/
8	RO	0x0	IU0. In-Use flag0, covering lock register0-31. 0: All lock register 0-31 are in the Not Taken state. 1: At least one of the lock register 0-31 is in the Taken state.
7:0	/	/	/

#### 3.15.5.2. Spinlock Register Status (Default: 0x00000000)

Offset: 0x10			Register Name: SPINLOCK_STATUS_REG
Bit	R/W	Default/Hex	Description
[i] (i=0~31)	RO	0x0	LOCK_REG_STATUS. SpinLock[i] status (i=0~31) 0: The Spinlock is free, 1: The Spinlock is taken.

#### 3.15.5.3. Spinlock Register N (N=0 to 31)(Default :0x00000000)

Offset:0x100+N*0x4 (N=0~31)			Register Name: SPINLOCKN_LOCK_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	TAKEN. Lock State. Read 0x0: The lock was previously Not Taken (free).The requester is granted the lock. Write 0x0: Set the lock to Not Taken (free). Read 0x1: The lock was previously Taken. The requester is not granted

			the lock and must retry. Write 0x1: No update to the lock value.
--	--	--	---

### 3.15.6. Programming Guidelines

Take CO-CPU0's synchronization with CPUS with Spinlock0 for an example, CPU0 takes the spinlock0 firstly in the instance:

CPU0 of Cluster 0

Step 1: CPU0 initializes Spinlock

```
writel(readl(AHB1_GATING_REG)|(1<<22),AHB1_GATING_REG); //open Spinlock clock gating
writel (readl(AHB1_RST_REG)|(1<<22), AHB1_RST_REG); //software reset Spinlock
```

Step 2: CPU0 requests to take spinlock0

```
rdata=readl(SPINLOCK_STATUS_REG); //check lock register0 status, if it is taken, check till
if(rdata != 0) rdata=readl(SPINLOCK_STATUS_REG); // lock register0 is free
    ⋮
rdata=readl(SPINLOCKN_LOCK_REG); //request to take spinlock0, if fail, retry till
if(rdata != 0) rdata=readl(SPINLOCKN_LOCK_REG); // lock register0 is taken
    ⋮
----- CPU0 critical code section -----
```

Step 3: CPU0 free spinlock0

```
writel (0, SPINLOCKN_LOCK_REG); //CPU0 frees spinlock0
```

Step 4: CPU0 waits for CPUS' freeing spinlock0

```
writel (readl(SPINLOCK_STATUS_REG) == 1); // CPU0 waits for CPUS' freeing spinlock0
```

CPUS

Step 1: CPU0 has taken spinlock0, CPUS waits for CPU0' freeing spinlock0

```
while(readl(SPINLOCK_STATUS_REG) == 1); // CPUS waits for CPU0' freeing spinlock0
```

Step 2: CPUS takes spinlock0 and go on

----- CPUS critical code section -----

Step 3: CPUS frees spinlock0

```
writel (0, SPINLOCKN_LOCK_REG); //CPUS frees spinlock0
```

## 3.16. Security System

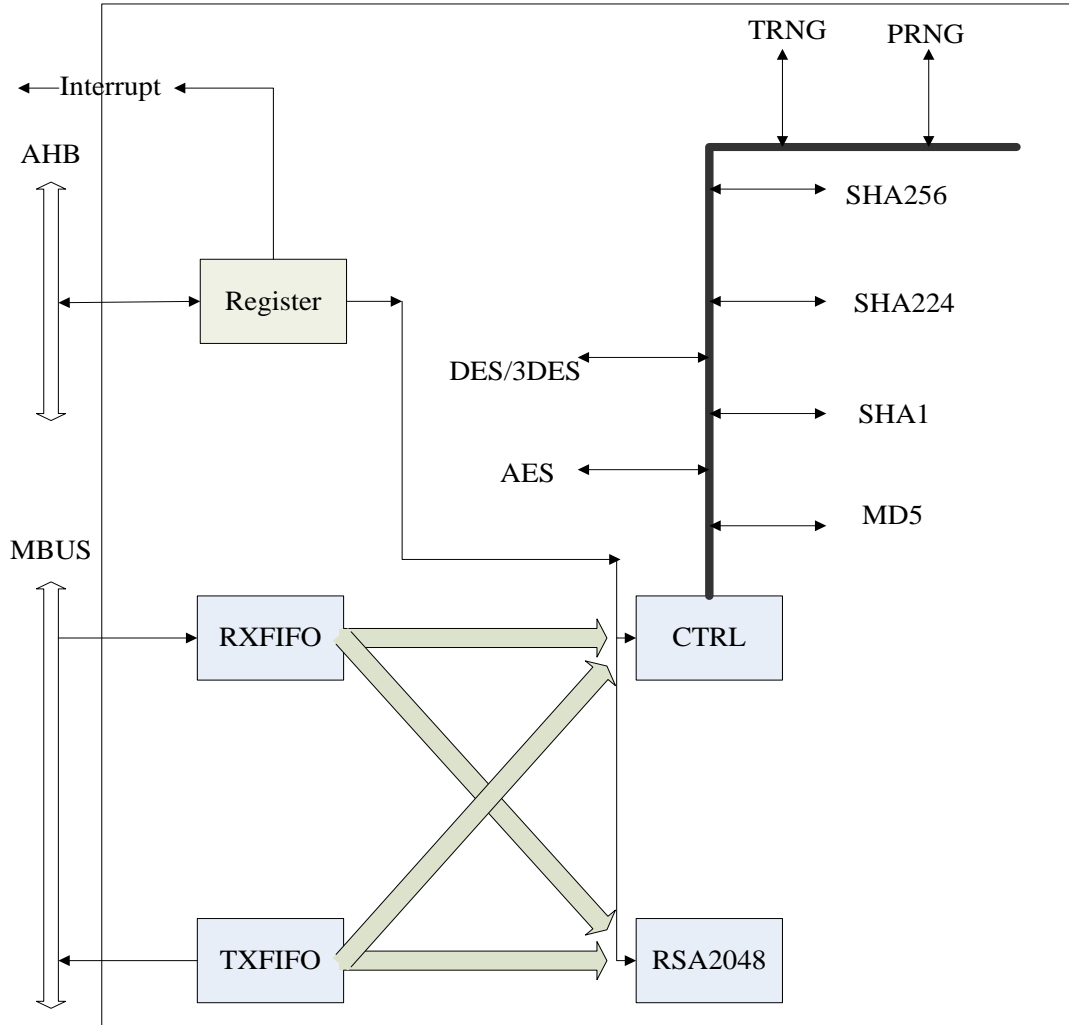
### 3.16.1. Overview

The Security System (SS) is one encrypt/ decrypt function accelerator. It is suitable for a variety of applications. It can support both encryption/decryption and signature/verification, calculate the hash value. Several modes are supported by the security system. SS has an internal DMA(IDMA) controller to transfer data between SS and memory.

It includes the following features:

- Support symmetrical algorithm :AES, DES, 3DES
- Support Secure Hash algorithm: MD5, SHA-1,SHA-224,SHA-256
- Support non-symmetrical algorithm :RSA2048
- Support 160-bits hardware PRNG with 175-bits seed
- Support 256-bits hardware TRNG
- Support ECB, CBC, CTR modes for DES/3DES
- Support ECB, CBC, CTR,CTS modes for AES
- Support 128-bits, 192-bits and 256-bits key size for AES
- Support 16bit/32bit/64bit/128bit wide\_size for AES CTR
- Support 16bit/32bit/64bit wide\_size for DES/3DES CTR
- Support IDMA mode
- Support serial and parallel mode

3.16.2. Block Diagram



### 3.16.3. Security System Register List

Module Name	Base Address
SS	0x01C02000

Register Name	Offset	Description
SS_CTL	0x00	Security Control Register
SS_ICR	0x04	Security Interrupt Control Register
SS_ISR	0x08	Security Interrupt Status Register
SS_Key_Address	0x10	Security Input Key Address Register
SS_IV_Address	0x18	Security Initialization Vector/Preload Counter/Public Modulus Address Register
SS_DataSrc_Address	0x20	Security Source Address Register
SS_DataDst_Address	0x28	Security Destination Address Register
SS_Data_Length	0x30	Security Data Length Register
SS_CTR0	0x34	Security Counter0 Register for Stream0
SS_CTR1	0x38	Security Counter1 Register for Stream0
SS_CTR2	0x3C	Security Counter2 Register for Stream0
SS_CTR3	0x40	Security Counter3 Register for Stream0
SS_CTR4	0x48	Security Counter4 Register for Stream1
SS_CTR5	0x4C	Security Counter5 Register for Stream1
SS_CTR6	0x50	Security Counter6 Register for Stream1
SS_CTR7	0x54	Security Counter7 Register for Stream1

### 3.16.4. Security System Register Description

#### 3.16.4.1. Security System Control Register

Offset: 0x00			Register Name: SS_CTL
Default Value: 0x2000_0000			
Bit	Read/Write	Default	Description
31	R/W	0	Stream1_SELECT This bit is set by software and cleared by hardware. Software write "1" select stream1, when write "0" invalid.
30	R/W	0	Stream0_SELECT This bit is set by software and cleared by hardware. Write "1" select stream0,,when write "0" invalid. <b>Note: When execute a stream, be sure to write bit31 or bit30 to 1; When both bit31 and bit30 written at the same time, the priority of bit30 is higher than bit31; When both bit31 and bit30 are 0, on behalf of no stream in the execution.</b>
29	R	1	SS_IDLE 0: the status of SS is busy 1: the status of SS is idle
28	R/W	0	FLOW_MODE Mode with the last flow 0:non-continue mode 1:continue mode
27	R/W	0	DMA Read/Write Consistent 0:Send end flag after data write-instruction finished 1:Read data when receive response of write-instruction ,if writing is non-finished, waiting until write finished.

			<p>SKEY_SELECT</p> <p>AES/DES/3DES key select</p> <p>0: Select input SS_KEYx (Normal Mode)</p> <p>1:Select {huk }</p> <p>2:Select {rotpk }</p> <p>3:Select {ssk }</p> <p>4:Select {backupkey2, backupkey1}</p> <p>5:Select {backupkey3, backupkey2}</p> <p>6:Select {backupkey4, backupkey3}</p> <p>7:Select {backupkey1, backupkey4}</p> <p>8-15: Select internal Key n (n from 0 to 7)</p> <p>Others: Reserved</p>
26:23	R/W	0	
			DIE_ID
22:20	R	x	Die Bonding ID
19	/	/	/
			<p>TRNG/PRNG_MODE</p> <p>TRNG/PRNG generator mode</p> <p>0: One-shot mode</p> <p>1: Continue mode</p>
18	R/W	0	
			<p>IV_MODE</p> <p>IV Steady of SHA-1/SHA-256/MD5 constants</p> <p>0: Constants</p> <p>1: Arbitrary IV</p> <p>Notes: It is only used for SHA-1/SHA-256/MD5 engine.</p>
17	R/W	0	
			<p>AES_CTS_LAST_PACKAGE_FLAG</p> <p>When writing "1", it means this is the last package for AES-CTS mode.</p>
16	R/W	0	



			(the size of the last package >128bit)
15	/	/	/
			SS_OP_MODE SS Operation Mode 00: Electronic Code Book (ECB) mode 01: Cipher Block Chaining (CBC) mode 10: Counter (CTR) mode
14:13	R/W	0	11: AES Ciphertext Stealing (CTS) mode
			CTR_WIDTH Counter Width for CTR Mode 00: 16-bits Counter 01: 32-bits Counter 10: 64-bits Counter
12:11	R/W	0	11: 128-bits Counter
			RSA Pubic Modulus_Width 10: 2048 bit Other: Reserved The only valid value in this field is 0x10, other value in this value is undefined.
10:9	R/W	0	
			AES_KEY_SIZE Key Size for AES 00: 128-bits 01: 192-bits 10: 256-bits
8:7	R/W	0	11: Reserved
6	R/W	0	SS_OP_DIR

			SS Operation Direction  0: Encryption  1: Decryption
5:2	R/W	0	SS_METHOD  SS Method  0000: AES  0001: DES  0010: Triple DES (3DES)  0011: MD5  0100: PRNG  0101: TRNG  0110: SHA-1  0111: SHA-224  1000: SHA-256  1001: RSA  Others: Reserved
1	R/W	0	PRNG/TRNG_START  PRNG/TRNG start bit  In PRNG/TRNG one-shot mode, write '1' to start PRNG/TRNG. After generating one group random data , this bit is clear to '0' by hardware.
0	R/W	0	SS_Start  Write '1' to start SS .

### 3.16.4.2. Security System Interrupt Control Register

<b>Offset: 0x04</b>	<b>Register Name: SS_ICR</b>
---------------------	------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:2	/	/	/
1	R/W	0	FLOW1_ENCRY/DECRY_END_Enable Flow1 Encry/Decry End Available Interrupt Enable 0: Disable 1: Enable
0	R/W	0	FLOW0_ENCRY/DECRY_END_Enable Flow0 Encry/Decry End Available Interrupt Enable 0: Disable 1: Enable

### 3.16.4.3. Security System Interrupt Status Register

			Register Name: SS_ISR
Offset: 0x08			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:2	/	/	/
1	R/W	0	FLOW1_ENCRY/DECRY_END_PENDING_BIT Flow1 Encry/Decry End Available Pending bit 0: No end pending 1: end pending Notes: Write '1' to clear it
0	R/W	0	FLOW0_ENCRY/DECRY_END_PENDING_BIT Flow0 Encry/Decry End Available Pending bit 0: No end pending 1: end pending Notes: Write '1' to clear it.

#### 3.16.4.4. Security System Key Address Register

<b>Offset: 0x10</b>			<b>Register Name: SS_KEY_Address</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	SS Key Input Address(2words alignment)write in the register

#### 3.16.4.5. Security System Pubic Modulus/IV/ Counter Address Register

<b>Offset: 0x18</b>			<b>Register Name: SS_PM/IV/CNT_Address</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	SS Initialization Vector/Preload Counter/Pubic Modulus 32bit Address(2words alignment)write in the register

#### 3.16.4.6. Security System DataSrc\_Address Register

<b>Offset: 0x20</b>			<b>Register Name: SS_DataSrc_Address</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	SS Data Source Address Source address (2words alignment)write in the register

#### 3.16.4.7. Security System DataDst\_Address Register

<b>Offset: 0x28</b>			<b>Register Name: SS_DataDst_Address</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	SS Data Destination Address Destination address(2words alignment) write in the register

**3.16.4.8. Security System Data Length Register**

<b>Offset: 0x30</b>			<b>Register Name: SS_Data_Length</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	SS Data Length Record the length of plaint/cipher and random data access in Word. 4 words alignment for AES; 2 words alignment for DES/3DES; 16 words alignment for SHA1/MD5/SHA224/SHA256; 5 words alignment for PRNG random data; 8 words alignment for TRNG random data; <b>Note:When it uses in AES-CTS mode , Data Length access in Byte.</b>

**3.16.4.9. Security System Counter[n] Register**

<b>Offset: 0x34+4*n</b>			<b>Register Name: SS_CTR[n]</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R	0	SS_CTR_VALUE Record counter's middle value every block in AES/DES/3DES CTR mode (n=0~3) <b>Note:SS_CTR[0] ~SS_CTR[3] for stream0 only.</b>

**3.16.4.10. Security System Counter[n] Register**

<b>Offset: 0x48+4*(n-4)</b>			<b>Register Name: SS_CTR[n]</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description

31:0	R	0	SS_CTR_VALUE  Record counter's middle value every block in AES/DES/3DES CTR mode (n=4~7)  <b>Note:SS_CTR[4] ~SS_CTR[7] for stream1 only.</b>
------	---	---	--

### 3.16.5. Security System Clock Requirement

Clock Name	Description	Requirement
ahb_clk	AHB bus clock	>=24MHz
ss_clk	SS serial clock	<= 400MHz && >=24MHz

### 3.16.6. Programming Guidelines

- (1) After the selected signal of channel-0 or channel-1 is recognized as low level(the bit31 or bit30 of SS\_CTL register is 0),channel-0 or channel-1 could be requested to calculate.
- (2) The input data sequence of RSA2048 is that the low word store in the low address.
- (3) Before the Hash algorithm is operated, the SS controller need reset, which avoid the influence of other algorithms to the Hash algorithm.
- (4) SHA224 don't support arbitrary iv mode.
- (5) For SHA1/SHA256,It should be noted the sequence of the initial hash value.

For example,the initial hash value of SHA1 in Fips180-2,H<sup>(0)</sup> shall consist of the following five 32-bit words,in hex:

H<sub>0</sub><sup>(0)</sup> = 67452301

H<sub>1</sub><sup>(0)</sup> = efc dab89

H<sub>2</sub><sup>(0)</sup> = 98badcfe

H<sub>3</sub><sup>(0)</sup> = 10325476

H<sub>4</sub><sup>(0)</sup> = c3d2e1f0

**For SHA1**,When we write the initial value in the IV address, according to the following array input sequence:

```
unsigned char iv_sha1[20]={
0x01,0x23,0x45,0x67,0x89,0xab,0xcd,0xef,
0xfe,0xdc,0xba,0x98,0x76,0x54,0x32,0x10,
0xf0,0xe1,0xd2,0xc3};
```

Then the five 32bit words in hex :

IV address : 0x67452301

IV address+0x4: 0xefcdab89

IV address+0x8: 0x98badcfe

IV address+0xc: 0x10325476

IV address+0x10:0xc3d2e1f0

**For SHA256**, according to the following array input sequence:

```
unsigned char iv_sha256[32]={
```

```
0x67,0xe6,0x09,0x6a,0x85,0xae,0x67,0xbb,  
0x72,0xf3,0x6e,0x3c,0x3a,0xf5,0x4f,0xa5,  
0x7f,0x52,0x0e,0x51,0x8c,0x68,0x05,0x9b,  
0xab,0xd9,0x83,0x1f,0x19,0xcd,0xe0,0x5b};
```

- (6) The message digest results of SHA1/SHA256 are stored in DataDst Address, When using SHA1/SHA256 to calculate in continue div-package mode,the initial hash values of the next package are the converted data of the last message digest results.For example, the first 32-bits word of the last message digest is 0x12345678,then the first 32-bits initial hash value of the next package is 0x78563412 by the software converted. Other words will be the same treatment.
- (7) The generation of the true random numbers uses one 16MHz Oscillator source. So when using TRNG algorithm, the 16MHz oscillator need enable, that the bit[0] of **OSC24M\_CTRL\_REG** in R\_PRCM is wrote to 1.After completed, the 16MHz oscillator need disable.

## 3.17. Security ID

### 3.17.1. Overview

There is one on chip EFUSE, which provides 128-bit, 64-bit and one 32-bit electrical fuses for security application. The users can use them as root key, security JTAG key and other applications.

It includes the following features:

- 128-bit electrical fuses for chip ID
- 64-bit electrical fuses for thermal sensor



## 3.18. Secure Memory Controller

### 3.18.1. Overview

The SMC is an Advanced Microcontroller Bus Architecture compliant System-on-Chip peripheral. It is a high-performance, area-optimized address space controller with on-chip AMBA bus interfaces that conform to the AMBA Advanced extensible Interface protocol and the AMBA Advanced Peripheral Bus protocol.

You can configure the SMC to provide the optimum security address region control functions required for your intended application.

The SMC includes the following features:

- Enables you to program security access permissions each address region.
- Permits the transfer of data between master and slave only if the security status of the AXI transaction matches the security settings of the memory region it addresses.
- Support Master DRM access.

### 3.18.2. Signal Description

Signal Name	Direction	Description
<b>smc_int</b>	Output	If this signal is HIGH, then the SMC has denied the AXI master access to a region.

### 3.18.3. Functionalities Description

#### 3.18.3.1. Typical Applications

##### 3.18.3.1.1. Regions

A region is a contiguous area of address space. The SMC provides each region with a programmable security permissions field. The security permissions value is used to enable the SMC to either accept or deny a transaction access to that region.

The SMC always provides two regions, region 0 and region 1, and you can configure it to provide additional regions. With the exception of region 0, the SMC enables you to program the following operating parameters for each region:

- Region enable.
- Security permissions.
- Base address.
- Size, the minimum address size of a region is 32KB.
- Subregion disable. See **Subregions** section.

##### 3.18.3.1.2. Priority

The priority of a region is fixed and is determined by the region number. The higher priority of a region increases with the region number.

When a transaction is received, its address is checked for a match with all the configured regions in turn. The order in which the regions are checked is determined by the priority level, the highest priority level is first. The first region that matches the transaction address match is used as the matching region. The matching regions security permission determines whether the transaction is permitted.

##### 3.18.3.1.3. Subregions

The SMC divides each region into eight equal-sized, non-overlapping subregions.

With the exception of region 0, you can program the SMC to disable any or all of the eight subregions that comprise a region. When a subregion is disabled, the security permissions for its address range are provided by the next highest priority region that overlaps the address range.

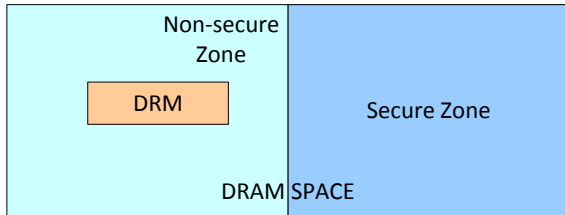
##### 3.18.3.1.4. Region security permissions

The SMC enables you to program the security access permissions for any region that it is configured. A region is assigned a security permissions field, sp<n>, in its **Region Attribute Register** that enables you to have complete control of the permissions for that region.

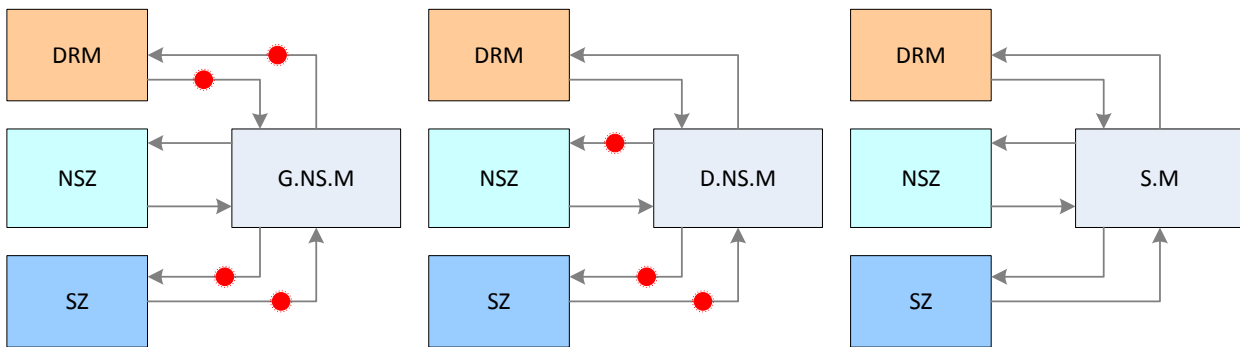
### 3.18.4. Diagram and Table

#### 3.18.4.1. DRM Diagram

G. NS.M stands for General Non-secure Master  
 D. NS.M stands for Non-secure Master appointed by DRM  
 S.M. stands for Secure Mater



G.NS.M only can read data from NSZ and write data into NSZ  
 D.NS.M can read data from NSZ and DRM, but only can write data into DRM  
 S.M can read data from the whole DRAM SPACE



#### 3.18.4.2. Master ID Table

ID	Master	ID	Master
0	CPU_M1	18	SS
1	CPU_M2	19	TS
2		20	DMA
3	USB3	21	NDFC0
4	FE0	22	NDFC1
5	BE1	23	CPUS
6	BE2	24	ATH
7	IEP0	25	EMAC
8	FE1	26	USB0
9	BE0	27	SDMMC0
10	FE2	28	SDMMC1
11	IEP1	29	SDMMC2
12	VED	30	/
13	VEE	31	USB1
14	FD	32	GPU0
15	CSI	33	GPU1

16	MP	34	USB2
17	HSI	35	CPU_M0

### 3.18.4.3. Region Size Table

Size<n>	Size of region<n>	Base address constraints
b000000-b001101	Reserved	-
b001110	32KB	-
b001111	64KB	Bit [15] must be zero
b010000	128KB	Bits [16:15] must be zero
b010001	256KB	Bits [17:15] must be zero
b010010	512KB	Bits [18:15] must be zero
b010011	1MB	Bits [19:15] must be zero
b010100	2MB	Bits [20:15] must be zero
b010101	4MB	Bits [21:15] must be zero
b010110	8MB	Bits [22:15] must be zero
b010111	16MB	Bits [23:15] must be zero
b011000	32MB	Bits [24:15] must be zero
b011001	64MB	Bits [25:15] must be zero
b011010	128MB	Bits [26:15] must be zero
b011011	256MB	Bits [27:15] must be zero
b011100	512MB	Bits [28:15] must be zero
b011101	1GB	Bits [29:15] must be zero
b011110	2GB	Bits [30:15] must be zero
b011111	4GB	Bits [31:15] must be zero
b100000	8GB	Bits [32:15] must be zero

### 3.18.5. Operation Modes

#### 3.18.5.1. Security inversion

There are two modes of operation for the region security permissions, with or without security inversion.

By default, if you program a region to support non-secure accesses, the SMC ensures that region must also support secure accesses. For example, if you program the region permissions for region 3 to be non-secure read only, the SMC permits access to region 3 for secure reads and non-secure reads. If you require that some regions are not accessible to masters in Secure state, but are accessible in Non-secure state, then you must enable security inversion. See **Region security permissions** section and **Security Inversion Enable Register** for more information.

#### 3.18.5.2. Inversion is disabled

By default, security inversion is disabled and therefore the SMC only permits you to program certain combinations of security permissions. These combinations ensure that a master in Secure state is not denied access to a region that is programmed to only accept non-secure accesses. **Table 1** shows the possible security permissions when security inversion is disabled.

sp<n> field	Secure Read	Secure Write	Non-secure Read	Non-secure Write
4b0000	No	No	No	No
4b0100	No	Yes	No	No
4b0001, 4b0101	No	Yes	No	Yes
4b1000	Yes	No	No	No
4b0010, 4b1010	Yes	No	Yes	No
4b1100	Yes	Yes	No	No
4b1001, 4b1101	Yes	Yes	No	Yes
4b0110, 4b1110	Yes	Yes	Yes	No
4b0011-4b1111	Yes	Yes	Yes	Yes

**Table 1** Region security permissions when security inversion is disabled

#### 3.18.5.3. Inversion is enabled

If you enable security inversion, the SMC permits you to program any combination of security permissions as **Table 2** shows.

sp<n> field	Secure Read	Secure Write	Non-secure Read	Non-secure Write
4b0000	No	No	No	No
4b0001	No	No	No	Yes
4b0010	No	No	Yes	No
4b0011	No	No	Yes	Yes
4b0100	No	Yes	No	No
4b0101	No	Yes	No	Yes
4b0110	No	Yes	Yes	No
4b0111	No	Yes	Yes	Yes

4b1000	Yes	No	No	No
4b1001	Yes	No	No	Yes
4b1010	Yes	No	Yes	No
4b1011	Yes	No	Yes	Yes
4b1100	Yes	Yes	No	No
4b1101	Yes	Yes	No	Yes
4b1110	Yes	Yes	Yes	No
4b1111	Yes	Yes	Yes	Yes

**Table 2** Region security permissions when security inversion is enabled

### 3.18.6. DRM

The Master DRM is designed for secure video display channel, mainly for the DE's FE, BE, DRC and VE's decode and encode modules. A partition of DRAM is defined as the DRM area, which features:

- 1) This area can be accessed by secure device, since secure device has the top priority;
- 2) VE's decode and encode cannot access this area when DRM is not enabled;
- 3) DE's FE, BE, and DRC cannot access this area when DRM is not enabled;
- 4) This area cannot be accessed by non-secure devices;
- 5) After enabled, VE decode and encode access the DRM space invalid, and write operation would be invalid if the address is not in the DRM space.
- 6) After enabled, the masters of DE system, such as FE, BE and DRC, access the DRM space invalid, and write operation would be invalid if the address is not in the DRM space.

Invalid access means data cannot be read correctly, only 0 will be returned, and the write operations will be shielded by SMC.

### 3.18.7. SMC Register List

Module Name	Base Address
SMC	0x01C0B000

Register Name	Offset	Description
CONFIG_REG	0x0000	Configuration Register
ACTION_REG	0x0004	Action Register
LKDW_RANGE_REG	0x0008	Lockdown Range Register
LKDW_SELECT_REG	0x000C	Lockdown Select Register
INT_STATUS_REG	0x0010	Interrupt Status Register
INT_CLEAR_REG	0x0014	Interrupt Clear Register
MASTER_BYPASS_REG0	0x0018	Master Bypass Register 0
MASTER_SECURITY_REG0	0x001C	Master Security Register 0
FAIL_ADDR_REG	0x0020	Fail Address Register
FAIL_CTRL_REG	0x0028	Fail Control Register
FAIL_ID_REG	0x002C	Fail ID Register
SPECULATION_CTRL_REG	0x0030	Speculation Control Register
SEC_INVER_EN_REG	0x0034	Security Inversion Enable Register
MASTER_BYPASS_REG1	0x0040	Master Bypass Register 1
MASTER_SECURITY_REG1	0x0044	Master Security Register 1
MASTER_DRM_EN_REG	0x0050	Master DRM Enable Register
DRM_ILLEGAL_ACCESS_REG0	0x0058	DRM Illegal Access Register 0
DRM_ILLEGAL_ACCESS_REG1	0x005C	DRM Illegal Access Register 1
DRM_LOW_SADDR_REG	0x0060	DRM Low Start Address Register
DRM_HIGH_SADDR_REG	0x0064	DRM High Start Address Register
DRM_LOW_EADDR_REG	0x0068	DRM Low End Address Register
DRM_HIGH_EADDR_REG	0x006C	DRM High End Address Register
REGION_SETUP_LOW_REG	0x0100+N*0x10	Region Setup Low Register N (N=0~15)
REGION_SETUP_HIGH_REG	0x0104+N*0x10	Region Setup High Register N (N=0~15)
REGION_ATTRIBUTE_REG	0x0108+N*0x10	Region Attribute Register N (N=0~15)

### 3.18.8. SMC Register Description

#### 3.18.8.1. Configuration Register (Default: 0x00001F0F)

Offset: 0x0000			Register Name: <b>CONFIG_REG</b>
Bit	R/W	Default/Hex	Description
31:4	/	/	/
3:0	R	0xF	REGIONS_RTN. Returns the number of the regions that the SMC provides. 0000: Reserved. 0001: 2 regions. ..... 1111: 16 regions.

#### 3.18.8.2. Action Register (Default: 0x00000001)

Offset: 0x0004			Register Name: <b>ACTION_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x1	SMC_INT_RESP. Control how the SMC uses the bresps[1:0], rresps[1:0], and smc_int signals when a region permission failure occurs: 00: Sets smc_int LOW and issues an OKEY response. 01: Sets smc_int LOW and issues a DECERR response. 10: Sets smc_int HIGH and issues an OKEY response. 11: Sets smc_int HIGH and issues a DECERR response.

#### 3.18.8.3. Lockdown Range Register (Default: 0x00000000)

Offset: 0x0008			Register Name: <b>LKDW_RANGE_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	LOCKDOWN_EN. When set to 1, it enables the lockdown_regions field to control the regions that are to be locked.
30:4	/	/	/
3:0	R/W	0x0	NO_REGIONS_LOCKDOWN. Control the number of regions to lockdown when the enable bit is set to 1. 0000: Region no_of_regions-1 is locked 0001: Region no_of_regions-1 to region no_of_regions-2 are locked .....



			1111: Region no_of_regions-1 to region no_of_regions-16 are locked
--	--	--	--

**Note:**

- No\_of\_regions is the value of the no\_of\_regions field in the configuration register.
- The value programmed in lockdown\_range register must not be greater than no\_of\_regions-1 else all regions are locked.

**3.18.8.4. Lockdown Select Register (Default: 0x00000000)**

Offset: 0x000C			Register Name: <b>LKDW_SELECT_REG</b>
Bit	R/W	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	ACCESS_TYPE_SPECU. Modify the access type of the speculation_control register: 0: No effect. The speculation register remains RW. 1: Speculation_control register is RO
1	R/W	0x0	ACCESS_TYPE_SEC_INV_EN. Modify the access type of the security_inversion_en register. 0: No effect. Security_inversion_en register remains RW. 1: Security_inversion_en register is RO
0	R/W	0x0	ACCESS_TYPE_LOCKDOWN_RANGE. Modify the access type of the lockdown_range register. 0: No effect. Lockdown_range register remains RW 1: Lockdown_range register is RO.

**3.18.8.5. Interrupt Status Register (Default: 0x00000000)**

Offset: 0x0010			Register Name: <b>INT_STATUS_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1	R	0x0	INT_OVERRUN. When set to 1, it indicates the occurrence of two or more region permission failure since the interrupt was last cleared.
0	R	0x0	INT_STATUS. Return the status of the interrupt. 0: Interrupt is inactive. 1: Interrupt is active.

### 3.18.8.6. Interrupt Clear Register (Default: 0x00000000)

Offset: 0x0014			Register Name: <b>INT_CLEAR_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	SMC_CLR_REG. Write any value to the int_clear register sets the : Status bit to 0 in the int_status register Overrun bit to 0 in the int_status register.

**Note:**

It will be auto clear after the write operation.

### 3.18.8.7. Master Bypass Register 0 (Default: 0xFFFFFFFF)

Offset: 0x0018			Register Name: <b>MST_BYPASS_REG0</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0xFFFFFFFF	MASTER_BYPASS. Master n Bypass Enable. If the master n bypass enable is set to 0, the master n access must be through the SMC. 0: Bypass Disable 1: Bypass Enable.

**Note:**

- n = 0~31, see the **MASTER ID Table** for detail.
- Bit[31:0] stand for Master ID [31:0].

### 3.18.8.8. Master Security Register 0 (Default: 0x00000000)

Offset: 0x001C			Register Name: <b>MST_SECURITY_REG0</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	MASTER_SEC0. Master n (except Cluster CPU, CPUS) Secure Configuration. 0: Secure 1: Non-secure.

**Note:**

n = 0~31, see the **MASTER ID Table** for detail.

### 3.18.8.9. Fail Address Register (Default: 0x00000000)

Offset: 0x0020			Register Name: <b>FAIL_ADDRESS_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	<b>FIRST_ACCESS_FAIL.</b> Return the address bits [31:0] of the first access to fail a region permission check after the interrupt was cleared. For external 16-bit DDR2, the address [2:0] is fixed to zero. For external 32-bit DDR2 and 16-bit DDR3, the address [3:0] is fixed to zero. For external 32-bit DDR3, the address [4:0] is fixed to zero.

### 3.18.8.10. Fail Control Register (Default: 0x00000000)

Offset: 0x0028			Register Name: <b>FAIL_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31:25	/	/	/
24	R	0x0	<b>READ_WRITE.</b> This bit indicates whether the first access to fail a region permission check was a write or read as: 0: Read access. 1: Write access.
23:22	/	/	/
21	R	0x0	<b>NON_SECURE.</b> After clearing the interrupt status, this bit indicates whether the first access to fail a region permission check was non-secure. Read as: 0: Secure access. 1: Non-secure access.
20	R	0x0	<b>PRIVILEGED.</b> After clearing the interrupt status, this bit indicates whether the first access to fail a region permission check was privileged. Read as: 0: Unprivileged access. 1: Privileged access.
19:0	/	/	/

### 3.18.8.11. Fail ID Register (Default: 0x00001F00)

Offset: 0x002C			Register Name: <b>FAIL_ID_REG</b>
Bit	R/W	Default/Hex	Description
31:24	/	/	/
23:16	R	0x0	<b>FAIL_BST_LEN.</b>

			Fail burst length. 0000:1 word length ..... 1111:16 words length
15:8	R	0x1F	Reserved to 0x1F.
7:0	R	0x0	FAIL_MASTER_ID. Fail Master ID.

**Note:**

 See the **MASTER ID Table** for detail.

**3.18.8.12. Speculation Control Register (Default: 0x00000000)**

Offset: 0x0030			Register Name: <b>SPECULATION_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	WRITE_SPECULATION. Write speculation. Control the write access speculation: 0: Write access speculation is enabled. 1: Write access speculation is disabled.
0	R/W	0x0	READ_SPECULATION. Read speculation. Control the read access speculation: 0: Read access speculation is enabled. 1: Read access speculation is disabled.

**3.18.8.13. Security Inversion Enable Register (Default: 0x00000000)**

Offset: 0x0034			Register Name: <b>SEC_INVER_EN_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	SECURITY_INVERSION_EN. Controls whether the SMC permits security inversion to occur. 0: Security inversion is not permitted. 1: Security inversion is permitted. This enables a region to be accessible to masters in Non-secure state but not accessible to masters in Secure state.

**3.18.8.14. Master Bypass Register 1 (Default: 0x00000007)**

Offset: 0x0040			Register Name: <b>MST_BYPASS_REG1</b>
Bit	R/W	Default/Hex	Description
31:3	/	/	/
2:0	R/W	0x7	MASTER_BYPASS. Master (n+31) Bypass Enable. If the master (n+31) bypass enable is set to 0, the master n access must be through the SMC. 0: Bypass Disable 1: Bypass Enable.

**Note:**

See the **MASTER ID Table** for detail.

**3.18.8.15. Master Security Register 1 (Default: 0x00000000)**

Offset: 0x0044			Register Name: <b>MST_SECURITY_REG1</b>
Bit	R/W	Default/Hex	Description
31:3	/	/	/
2:0	R/W	0x0	MASTER_SEC1. Master (n+31) (except Cluster CPU, CPUS) Secure Configuration. 0: Secure 1: Non-secure.

**Note:**

See the **MASTER ID Table** for detail.

**3.18.8.16. Master DRM Enable Register (Default: 0x00000000)**

Offset: 0x0050			Register Name: <b>MST_DRM_EN_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	DRM_EN. DRM enable.
30:12	/	/	/
11	R/W	0x0	DRC1_EN. DRC1 enable.
10	R/W	0x0	DRC0_EN. DRC0 enable.

9	R/W	0x0	DEBE2_EN. DEBE2 enable.
8	R/W	0x0	DEBE1_EN. DEBE1 enable.
7	R/W	0x0	DEBE0_EN. DEBE0 enable.
6	R/W	0x0	DEFE2_EN. DEFE2 enable.
5	R/W	0x0	DEFE1_EN. DEFE1 enable.
4	R/W	0x0	DEFE0_EN. DEFE0 enable.
3:2	/	/	/
1	R/W	0x0	VE_DECODE_EN. VE decode enable.
0	R/W	0x0	VE_ENCODE_EN. VE encode enable.

### 3.18.8.17. DRM Illegal Access Register 0 (Default: 0x00000000)

Offset: 0x0058			Register Name: <b>DRM_ILLEGAL_ACCESS_REG0</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DRM_ILLACCE_REG0. When a master, which is non-secure, accesses the DRM space, then the relevant bit would be set up.

### 3.18.8.18. DRM Illegal Access Register 1 (Default: 0x00000000)

Offset: 0x005C			Register Name: <b>DRM_ILLEGAL_ACCESS_REG1</b>
Bit	R/W	Default/Hex	Description
31:0	RO	0x0	DRM_ILLACCE_REG1. When a master, which is non-secure, accesses the DRM space, then the relevant bit would be set up.

### 3.18.8.19. DRM Low Start Address Register (Default: 0x00000000)

Offset: 0x0060			Register Name: <b>DRM_LOW_SADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:15	R/W	0x0	DRM_LOW_SADDR.

14:0	/	/	/
------	---	---	---

### 3.18.8.20. DRM High Start Address Register (Default: 0x00000000)

Offset: 0x0064			Register Name: <b>DRM_HIGH_SADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	DRM_HIGH_SADDR.

### 3.18.8.21. DRM Low End Address Register (Default: 0x00000000)

Offset: 0x0068			Register Name: <b>DRM_LOW_EADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:15	R/W	0x0	DRM_LOW_SADDR.
14:0	/	/	/

### 3.18.8.22. DRM High End Address Register (Default: 0x00000000)

Offset: 0x006C			Register Name: <b>DRM_HIGH_EADDR_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	DRM_HIGH_SADDR.

### 3.18.8.23. Region Setup Low Register (Default: 0x00000000)

Offset: 0x0100+N*0x10 (N=0~15)			Register Name: <b>REGION_SETUP_LOW_REG</b>
Bit	R/W	Default/Hex	Description
31:15	R/W	0x0	<p>BASE_ADDRESS_LOW.</p> <p>Controls the base address [31:15] of region&lt;n&gt;.</p> <p>The SMC only permits a region to start at address 0x0, or at a multiple of its region size.</p> <p>For example, if the size of a region is 512MB, and it is not at address 0x0, the only valid settings for this field are:</p> <p>17'b0010000000000000</p> <p>17'b0100000000000000</p> <p>17'b0110000000000000</p> <p>17'b1000000000000000</p> <p>17'b1010000000000000</p>

			17'b11000000000000000000 17'b11100000000000000000
14:0	/	/	/

**Note:**

- For region 0, this field is Read Only (RO). The SMC sets the base address of region 0 to 0x0.
- The base address should be equal to the DRAM absolutely address.

**3.18.8.24. Region Setup High Register (Default: 0x00000000)**

Offset: 0x0104+N*0x10 (N=0~15)			Register Name: <b>REGION_SETUP_HIGH_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	BASE_ADDRESS_HIGH The SMC only permits a region to start at address 0x0, or at a multiple of its region size. If you program a region size to be 8GB or more, then the SMC might ignore certain bits depending on the region size.

**3.18.8.25. Region Attribute Register (Default: 0x00000000)**

Offset: 0x0108+N*0x10 (N=0~15)			Register Name: <b>REGION_ATTRIBUTE_REG</b>
Bit	R/W	Default/Hex	Description
31:28	R/W	0x0	REGION_ATTR_SPN. SP<n>. Permission setting for region <n>. if an AXI transaction occurs to region n, the value in the sp<n> field controls whether the SMC permits the transaction to proceed. . See <b>Table 1</b> and <b>Table 2</b> .
27:16	/	/	/
15:8	R/W	0x0	SUB_REGION_DISABLE. Subregion_disable. Regions are split into eight equal-sized sub-regions, and each bit enables the corresponding subregion to be disabled. Bit [15] = 1 subregion 7 is disabled. Bit [14] = 1 subregion 6 is disabled. Bit [13] = 1 subregion 5 is disabled. Bit [12] = 1 subregion 4 is disabled. Bit [11] = 1 subregion 3 is disabled. Bit [10] = 1 subregion 2 is disabled. Bit [9] = 1 subregion 1 is disabled. Bit [8] = 1 subregion 0 is disabled.
7	/	/	/
6:1	R/W	0x0	REGION_ATTR_SIZE. Size<n>. Size of region<n>, see <b>Region Size Table</b> for detail.
0	R/W	0x0	REGION_ATTR_EN.



			EN<n>. Enable for region<n>. 0: Region < n> is disabled. 1: Region < n> is enabled.
--	--	--	---

**Note:**

For region 0, this field is reserved except sp<n> field.

## 3.19. SMTA

### 3.19.1. Overview

Secure Memory Touch Arbiter provides a software interface to the protection bits in a secure system in a TrustZone design. It provides system flexibility that enables to configure different areas of memory as secure or non-secure.

The SMTA includes the following features:

- It has protection bits to enable you to program up to 24 areas of memory as secure or non-secure
- Support 256KB secure RAM (SRAM B)
- It has an APB system bus interface

### 3.19.2. Signal Description

Signal Name	Direction	Description
SMTADECPROT0[7:0]	Output	Protection bits: 0: Secure 1: Non-secure
SMTADECPROT1[7:0]	Output	Protection bits: 0: Secure 1: Non-secure
SMTADECPROT2[7:0]	Output	Protection bits: 0: Secure 1: Non-secure

### 3.19.3. Functionalities Description

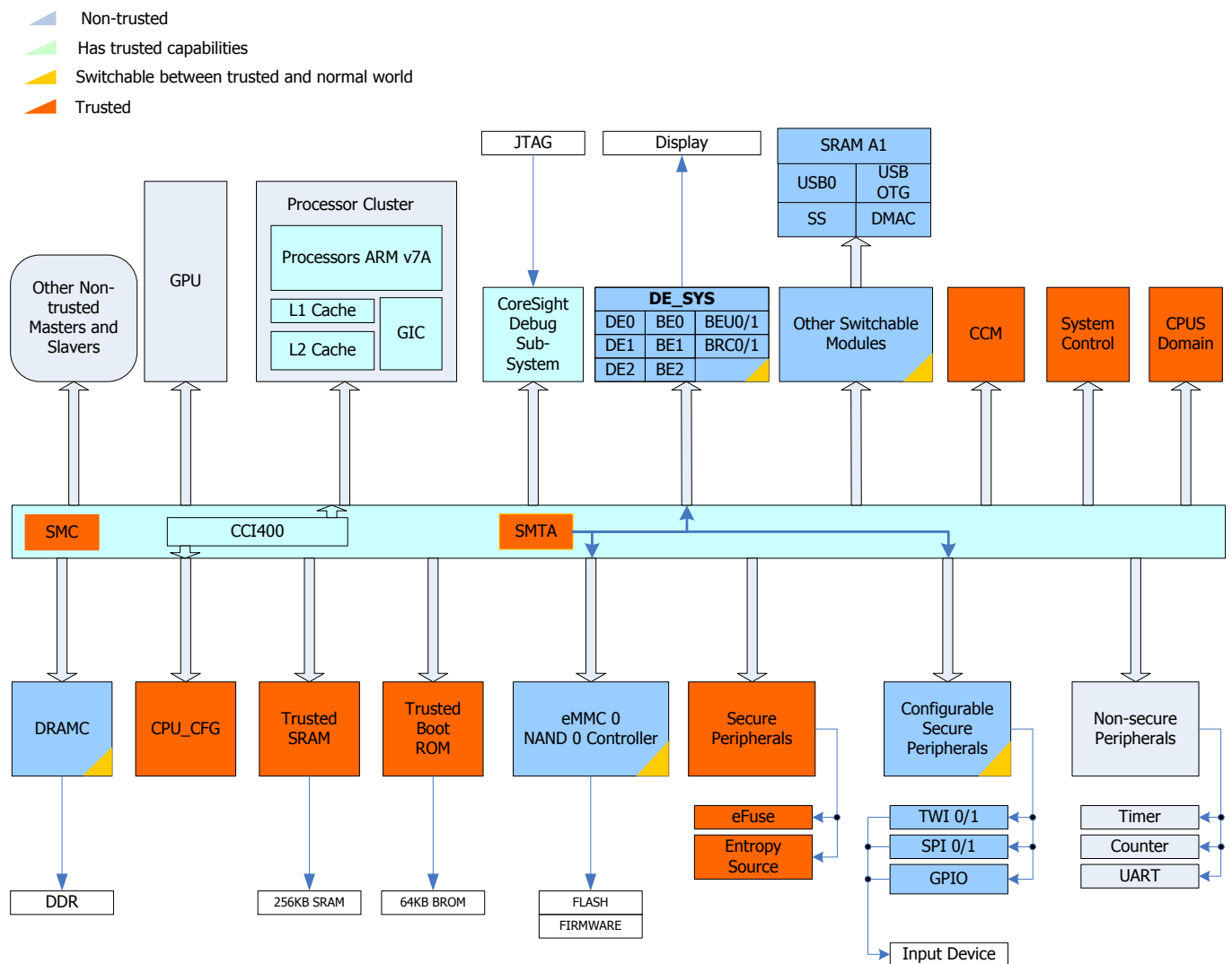
#### 3.19.3.1. Typical Applications

The SMTA provides a software interface to set up memory areas as secure or non-secure. It does this in two ways:

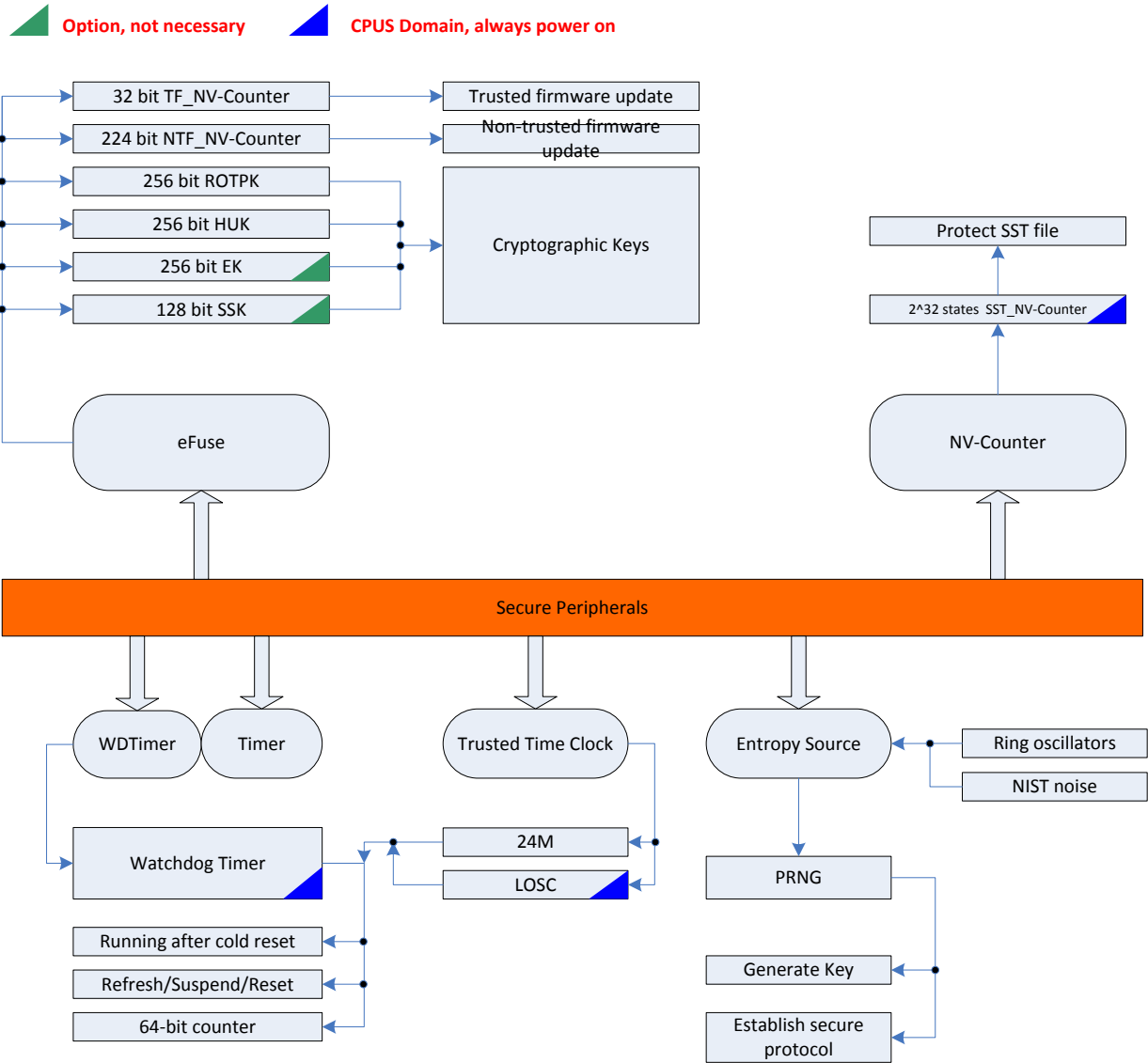
- Programmable protection bits that can be allocated to areas of memory as determined by an external decoder
- Programmable region size value for use by an AXI TrustZone Memory Adapter.

But in the SoC, the secure memory size is 256KB, and you can not use TZMA to split the RAM into two regions, because it is fixed to secure. And the first 4K RAM is used for Cluster CPU0 hot plug, the secure OS can not access the RAM.

#### 3.19.3.2. Functional Block Diagram



Trusted Base System Architecture



Secure Peripherals

### 3.19.3.3. SMTA Configuration Table

Register	Bit	SMTA0	SMTA1	SMTA2
SMTADECPORTx (x=0,1,2)	[0]	/	NDFC0	/
	[1]	TWI0	DMA	/
	[2]	TWI1	SS	/
	[3]	SPI0	SRAM A1	/
	[4]	SPI1	USB_DRD	/
	[5]	GPIO	USB Host	/
	[6]	DE_SYS	DRAMC	/
	[7]	SD/MMC	/	/

### 3.19.4. SMTA Register List

Module Name	Base Address
SMTA	0x06003400

Register Name	Offset	Description
SMTA_DECPORT0_STA_REG	0x0004	SMTA Decode Port0 Status Register
SMTA_DECPORT0_SET_REG	0x0008	SMTA Decode Port0 Set Register
SMTA_DECPORT0_CLR_REG	0x000C	SMTA Decode Port0 Clear Register
SMTA_DECPORT1_STA_REG	0x0010	SMTA Decode Port1 Status Register
SMTA_DECPORT1_SET_REG	0x0014	SMTA Decode Port1 Set Register
SMTA_DECPORT1_CLR_REG	0x0018	SMTA Decode Port1 Clear Register
SMTA_DECPORT2_STA_REG	0x001C	SMTA Decode Port2 Status Register
SMTA_DECPORT2_SET_REG	0x0020	SMTA Decode Port2 Set Register
SMTA_DECPORT2_CLR_REG	0x0024	SMTA Decode Port2 Clear Register

### 3.19.5. SMTA Register Description

#### 3.19.5.1. SMTA Decode Port0 Status Register (Default: 0x00000000)

Offset: 0x0004			Register Name: <b>SMTA_DECPORT0_STA_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	RO	0x0	STA_DEC_PROTO_OUT. Show the status of the decode protection output: 0: Decode region corresponding to the bit is secure. 1: Decode region corresponding to the bit is non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.2. SMTA Decode Port0 Set Register (Default: 0x00000000)

Offset: 0x0008			Register Name: <b>SMTA_DECPORT0_SET_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	SET_DEC_PORT0_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.3. SMTA Decode Port0 Clear Register (Default: 0x00000000)

Offset: 0x000C			Register Name: <b>SMTA_DECPORT0_CLR_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	CLR_DEC_PORT0_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.4. SMTA Decode Port1 Status Register (Default: 0x00000000)

Offset: 0x0010			Register Name: <b>SMTA_DECPORT1_STA_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	RO	0x0	STA_DEC_PROT1_OUT. Show the status of the decode protection output: 0: Decode region corresponding to the bit is secure. 1: Decode region corresponding to the bit is non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.5. SMTA Decode Port1 Set Register (Default: 0x00000000)

Offset: 0x0014			Register Name: <b>SMTA_DECPORT1_SET_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	SET_DEC_PORT1_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.6. SMTA Decode Port1 Clear Register (Default: 0x00000000)

Offset: 0x0018			Register Name: <b>SMTA_DECPORT1_CLR_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	CLR_DEC_PORT1_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

#### 3.19.5.7. SMTA Decode Port2 Status Register (Default: 0x00000000)

Offset: 0x001C			Register Name: <b>SMTA_DECPORT2_STA_REG</b>
----------------	--	--	---



Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	RO	0x0	STA_DEC_PROT2_OUT. Show the status of the decode protection output: 0: Decode region corresponding to the bit is secure. 1: Decode region corresponding to the bit is non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

### 3.19.5.8. SMTA Decode Port2 Set Register (Default: 0x00000000)

Offset: 0x0020			Register Name: <b>SMTA_DECPORT2_SET_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	SET_DEC_PORT2_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to non-secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

### 3.19.5.9. SMTA Decode Port2 Clear Register (Default: 0x00000000)

Offset: 0x0024			Register Name: <b>SMTA_DECPORT2_CLR_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	WO	0x0	CLR_DEC_PORT2_OUT. Sets the corresponding decode protection output: 0: No effect. 1: Set decode region to secure. There is one bit of the register for each protection output (See the <b>SMTA Configuration Table</b> in detail).

### 3.19.6. Programming Guidelines

- The the space address of a module controller is defined as secure, non-secure master cannot visit this area, as a result, only 0 will be returned as a result, and write operations are invalid. Innormal access will not lead to system exception.
- The secure property of SRAM B cannot be modified, fixed to be secure. The first 4K space is used for hot plug function of Cluster CPU0, which cannot be visited by secure system.

## 3.20. GPADC

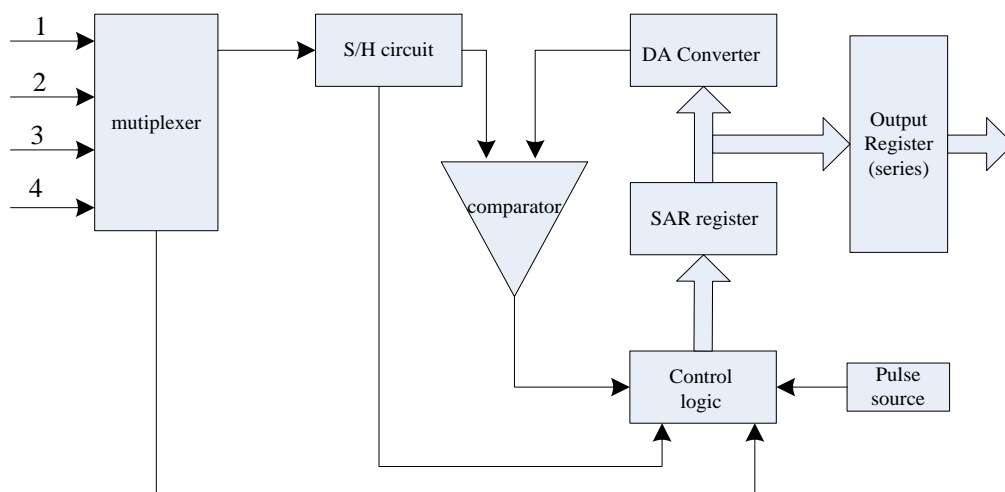
### 3.20.1. Overview

The general purpose ADC is a 12-bit sampling analog-to-digital converter with 4-channel multiplexer which can directly access any of 4-single-end analog signals. The ADC is a type of successive-approximation-register (SAR) converter and it can operate at up to a 1 MSPS conversion rate;

It features:

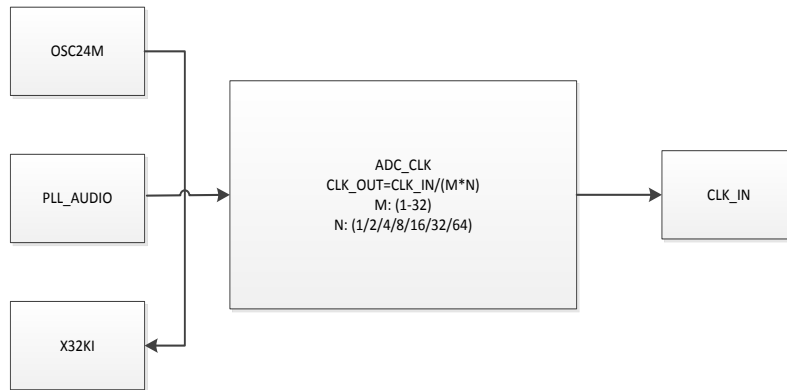
- Supports APB 32-bits bus width
- Interrupt support
- 12 bit SAR type A/D converter
- 4-channel multiplexer
- 32 FIFO depth of data register
- Power Supply Voltage:1.8V
- Analog Input Range:0 to 1.8V
- Maximum Sampling frequency: 1 MHz
- Low power dissipation

### 3.20.2. Block Diagram



### 3.20.3. Clock Tree and ADC Conversion Time

#### 3.20.3.1. Clock Tree



ADC Clock Tree

#### 3.20.3.2. A/D Conversion Time

When the clock source is 24MHz and the scale value M\*N is 6, total 12-bit conversion time is as follows.

$$CLK\_IN = 24MHz/6 = 4MHz$$

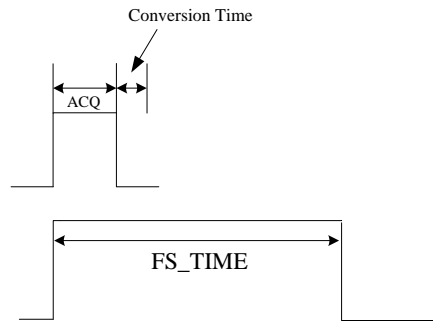
$$Conversion\ Time = 1/(4MHz/14Cycles) = 3.50us$$

$$\text{If ADC acquire time divider N is 5, then } TACQ = 1/(4MHz/6) = 1.50us$$

FS\_TIME (configured by the FS\_DIV register) bases on the summation of Conversion Time and TACQ. The FS\_TIME must be greater or equal than (TACQ + Conversion Time)

$$FS\_TIME \geq TACQ + Conversion\ Time = 5.0us$$

$$\text{Then ADC Sample Frequency Divider } n \geq 5 \times CLK\_IN/1M-1 = 19$$



General ADC Conversion phase

Note: This A/D converter was designed to operate at maximum 24MHz clock, and the conversion rate can go up to 1 MSPS.

#### 3.20.4. Programming Guide

- Configure the GPADC Clock Control Register, set Bit 17 of APB0 Module Clock Gating Register and Bit 17 of APB0 Module Software Reset Register to select the clock source.
- Configure the ADC PORT IO Configure Register to select the function of the ADC PORT
- FS\_DIV and TACQ must meet the following in equation:  
 $FS\_TIME \geq (TACQ + \text{Conversion Time})$ .
- Configure the ADC Interrupt & FIFO Control Register to choose read how many GPADC convert result in one interrupt function or flush the FIFO.
- Then enable the ADC channel and ADC by configure ADC Control Register 1.
- Configure ADC Control Register 1 to enable the ADC Calibration.
- Every pin can be set as ADC input channel、general input、general output, just change the value of ADC\_IO\_CONFIG register.
- When the ADC FIFO Data Available IRQ happened, the CPU can read values from the FIFO .
- When the ADC FIFO Over-Run IRQ happened, the CPU can flush the FIFO.

### 3.20.5. GPADC Register List

Module Name	Base Address
General Purpose ADC	0x06004C00

Register Name	Offset	Description
ADC_CTRL0	0x00	ADC Control Register 0
ADC_CTRL1	0x04	ADC Control Register 1
ADC_INT_FIFOC	0x08	ADC Interrupt FIFO Control Register
ADC_INT_FIFOS	0x0c	ADC Interrupt FIFO Status Register
ADC_DATA	0x10	ADC Data Register
ADC_CDATA	0x14	ADC Common Data Register
ADC_IO_CONFIG	0x18	ADC IO Configuration Register
ADC_PORT_DATA	0x20	ADC IO Port Data Register

### 3.20.6. GPADC Register Description

#### 3.20.6.1. ADC Control Register 0

Offset: 0x00			Register Name: ADC_CTRL0
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0xF	FS_DIV. ADC Sample Frequency Divider CLK_IN/(n+1)
15:0	R/W	0x0	TACQ. ADC acquire time CLK_IN/(N+1)

Notes: This register is configured to control the conversion rate of the SAR ADC. The ADC can work at a maximum up to 1 MSPS. FS\_DIV and TACQ must meet the following in equation: FS\_TIME >= (TACQ + Conversion Time).

#### 3.20.6.2. ADC Control Register 1

Offset: 0x04			Register Name: ADC_CTRL1
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x0	ADC_FIRST_DLY. ADC First Convert Delay setting, ADC conversion of each channel is delayed by N samples
23:22	/	/	/
21:20	R/W	0x0	ADC_OP_BIAS.(Adjust the bandwidth of the ADC amplifier) ADC OP Bias
19:18	/	/	/
17	R/W	0x0	ADC_CALI_EN. ADC Calibration 1: start Calibration, it is clear to 0 after calibration
16	R/W	0x0	ADC_EN. ADC Function Enable 0: Disable 1: Enable
15:4	/	/	/
3	R/W	0x0	ADC_CHAN3_SELECT. Analog input channel 3 Select 0: Disable

			1: Enable
2	R/W	0x0	ADC_CHAN2_SELECT. Analog input channel 2 Select 0: Disable 1: Enable
1	R/W	0x0	ADC_CHAN1_SELECT. Analog input channel 1 Select 0: Disable 1: Enable
0	R/W	0x0	ADC_CHAN0_SELECT. Analog input channel 0 Select 0: Disable 1: Enable

Notes: Channel 0~3 can be selected at the same time, but N channel selected, each channel has 1/N full speed of the ADC. When there are only one channel selected, it has the full conversion rate.

### 3.20.6.3. ADC Interrupt & FIFO Control Register

Offset: 0x08			Register Name: ADC_INT_FIFOC
Bit	Read/Write	Default/Hex	Description
31:18	/	/	
17	R/W	0x0	ADC_OVERRUN_IRQ_EN. ADC FIFO Over Run IRQ Enable 0: Disable 1: Enable
16	R/W	0x0	ADC_DATA_IRQ_EN. ADC FIFO Data Available IRQ Enable 0: Disable 1: Enable
15:13	/	/	/
12:8	R/W	0xF	ADC_FIFO_TRIG_LEVEL. Interrupt and DMA request trigger level for ADC Trigger Level = TXTL + 1
7:5	/	/	/



4	R/W	0x0	ADC_FIFO_FLUSH.  ADC FIFO Flush  Write '1' to flush TX FIFO, self clear to '0'
3:0	/	/	/

#### 3.20.6.4. ADC Interrupt& FIFO Status Register

Offset: 0x0c			Register Name: ADC_INT_FIFOS
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	FIFO_OVERRUN_PENDING.  ADC FIFO Over Run IRQ pending  0: No Pending IRQ  1: FIFO Overrun Pending IRQ  Write '1' to clear this interrupt or automatic clear if interrupt condition fails
16	R/W	0x0	FIFO_DATA_PENDING.  ADC FIFO Data Available pending Bit  0: NO Pending IRQ  1: FIFO Available Pending IRQ  Write '1' to clear this interrupt or automatic clear if interrupt condition fails
15:14	/	/	/
13:8	R	0x0	RXA_CNT.  ADC FIFO available Sample Word Counter
7:0	/	/	/

#### 3.20.6.5. ADC Data Register

Offset: 0x10			Register Name: ADC_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/

11:0	R	0x0	ADC_DATA.  ADC analog input data (FIFO address)
------	---	-----	---

Notes: The data stored in this FIFO bases on ADC\_CHAN(3~0)\_SELECT. If four channels are all enable, FIFO will access the input data in successive turn, first is ADC\_CHAN0 data, then ADC\_CHAN1, ADC\_CHAN2, ADC\_CHAN3 data. If there are only two or three channels selected, such as ADC\_CHAN0 and ADC\_CHAN3, firstly ADC\_CHAN0 input data is accessed, then ADC\_CHAN3 input data.

### 3.20.6.6. ADC Common Data Register

Offset: 0x14			Register Name: ADC_CDAT
Bit	Read/Write	Default/Hex	Description
			Default: 0x0000_0000
31:12	/	/	/
11:0	R/W	0x0	ADC_CDAT.  ADC Common Data

### 3.20.6.7. ADC IO Configure Register

Offset: 0x18			Register Name: ADC_IO_CONFIG
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14:12	R/W	0x2	CHAN3_IO_SELECT  Channel 3 IO Port Function Select:  000: Input            001:Output  010: CHAN3        011:/  100: /                101:/  110: /                111:/
11	/	/	/
10:8	R/W	0x2	CHAN2_IO_SELECT  Channel 2 IO Port Function Select:  000: Input            001:Output

			010: CHAN2      011:/  100: /              101:/  110: /              111:/
7	/	/	/
6:4	R/W	0x2	CHAN1_IO_SELECT  Channel 1 IO Port Function Select:  000: Input              001:Output  010: CHAN1              011:/  100: /                      101:/  110: /                      111:/
3	/	/	/
2:0	R/W	0x2	CHAN0_IO_SELECT  Channel 0 IO Port Function Select:  000: Input              001:Output  010: CHAN0              011:/  100: /                      101:/  110: /                      111:/

### 3.20.6.8. ADC Port Data Register

Offset: 0x20			Register Name: ADC_PORT_DATA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3:0	R/W	0x0	ADC_PORT_DATA  ADC Port Data Value, CHAN3, CHAN2, CHAN1, CHAN0

## 3.21. Thermal Sensor Controller

### 3.21.1. Overview

The thermal sensors have become common elements in wide range of modern system on chip (SOC) platform, which can be used to constantly monitor the temperature on the chip.

A80 embeds four thermal sensors in possible hot spots on the die, which located beside two clusters CPU, one cluster GPU and DRAM interface. The thermal sensor Generates interrupt to SW to lower temperature via DVFS, on reaching a certain thermal threshold. Sensor0 located beside the big CPU, Sensor1 located beside the DRAM, Sensor2 located beside the GPU and Sensor3 located beside the small CPU.

It features:

- Low power dissipation
- Power supply voltage:1.8V
- Support up to four thermal sensors
- Periodic temperature measurement
- Averaging filter for thermal sensor reading
- Support over-temperature protection interrupt and over-temperature alarm interrupt

### 3.21.2. Thermal Sensor Register List

Module Name	Base Address
Thermal Sensor	0x06004C00

Register Name	Offset	Description
THS_CTRL	0x40	THS Control Register
THS_INT_CTRL	0x44	THS Interrupt Control Register
THS_STAT	0x48	THS Status Register
THS_ALARM_CTRL0	0x50	Alarm Threshold Control Register 0
THS_ALARM_CTRL1	0x54	Alarm Threshold Control Register 1
THS_ALARM_CTRL2	0x58	Alarm Threshold Control Register 2
THS_ALARM_CTRL3	0x5c	Alarm Threshold Control Register 3
THP_SHUTDOWN_CTRL0	0x60	Shutdown Threshold Control Register 0
THP_SHUTDOWN_CTRL1	0x64	Shutdown Threshold Control Register 1
THP_SHUTDOWN_CTRL2	0x68	Shutdown Threshold Control Register 2
THP_SHUTDOWN_CTRL3	0x6c	Shutdown Threshold Control Register 3
THS_FILTER	0x70	Average Filter Control Register
THS0_1_CDAT	0x74	Thermal Sensor0_1 Calibration Data
THS2_3_CDAT	0x78	Thermal Sensor2_3 Calibration Data
THS0_DATA	0x80	THS0 Data Register
THS1_DATA	0x84	THS1 Data Register
THS2_DATA	0x88	THS2 Data Register
THS3_DATA	0x8c	THS3 Data Register

### 3.21.3. Thermal Sensor Register Description

#### 3.21.3.1. THS Control Register

Offset: 0x40			Register Name: THS_CTRL
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x4	SENSOR_ACQ. Sensor acquire time $CLK\_IN/(N+1)$
15:4	/	/	/
3	R/W	0x0	SENSE3_EN. Enable temperature measurement on sense point 3 0:Disable 1:Enable
2	R/W	0x0	SENSE2_EN. Enable temperature measurement on sense point 2 0:Disable 1:Enable
1	R/W	0x0	SENSE1_EN. Enable temperature measurement on sense point 1 0:Disable 1:Enable
0	R/W	0x0	SENSE0_EN. Enable temperature measurement on sense point 0 0:Disable 1:Enable

#### 3.21.3.2. THS Interrupt Control Register

Offset: 0x44			Register Name: THS_INT_CTRL
Bit	Read/Write	Default/Hex	Description
31:12	R/W	0x0	THERMAL_PER. $4096*(n+1)/CLK\_IN$
11	R/W	0x0	THS3_DATA_IRQ_EN.  Selects Temperature measurement data of sensing point 3 0: No select

			1: Select
10	R/W	0x0	THS2_DATA_IRQ_EN.  Selects Temperature measurement data of sensing point 2  0: No select 1: Select
9	R/W	0x0	THS1_DATA_IRQ_EN.  Selects Temperature measurement data of sensing point 1  0: No select 1: Select
8	R/W	0x0	THS0_DATA_IRQ_EN.  Selects Temperature measurement data of sensing point 0  0: No select 1: Select
7	R/W	0x0	SHUT_INT3_EN. Selects shutdown interrupt for sensing point 3 0: No select 1: Select
6	R/W	0x0	SHUT_INT2_EN. Selects shutdown interrupt for sensing point 2 0: No select 1: Select
5	R/W	0x0	SHUT_INT1_EN. Selects shutdown interrupt for sensing point 1 0: No select 1: Select
4	R/W	0x0	SHUT_INT0_EN. Selects shutdown interrupt for sensing point 0 0: No select 1: Select
3	R/W	0x0	ALARM_INT3_EN. Selects Alert interrupt for sensing point 3 0: No select 1: Select
2	R/W	0x0	ALARM_INT2_EN. Selects Alert interrupt for sensing point 2 0: No select 1: Select

1	R/W	0x0	ALARM_INT1_EN. Selects Alert interrupt for sensing point 1 0: No select 1: Select
0	R/W	0x0	ALARM_INT0_EN. Selects Alert interrupt for sensing point 0 0: No select 1: Select

### 3.21.3.3. THS Status Register

Offset: 0x48			Register Name: THS_STAT
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11	R/W	0x0	THS3_DATA_IRQ_STS. Data interrupt status for sensing point 3 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
10	R/W	0x0	THS2_DATA_IRQ_STS. Data interrupt status for sensing point2 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
9	R/W	0x0	THS1_DATA_IRQ_STS. Data interrupt status for sensing point 1 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
8	R/W	0x0	THS0_DATA_IRQ_STS. Data interrupt status for sensing point 0 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
7	R/W	0x0	SHUT_INT3_STS. Shutdown interrupt status for sensing point 3 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
6	R/W	0x0	SHUT_INT2_STS. Shutdown interrupt status for sensing point 2 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
5	R/W	0x0	SHUT_INT1_STS. Shutdown interrupt status for sensing point 1



			Write '1' to clear this interrupt or automatic clear if interrupt condition fails
4	R/W	0x0	SHUT_INT0_STS. Shutdown interrupt status for sensing point 0 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
3	R/W	0x0	ALARM_INT3_STS. Alert interrupt status for sensing point 3 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
2	R/W	0x0	ALARM_INT2_STS. Selects Alert interrupt for sensing point 2 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
1	R/W	0x0	ALARM_INT1_STS. Selects Alert interrupt for sensing point 1 Write '1' to clear this interrupt or automatic clear if interrupt condition fails
0	R/W	0x0	ALARM_INT0_STS. Selects Alert interrupt for sensing point 0 Write '1' to clear this interrupt or automatic clear if interrupt condition fails

#### 3.21.3.4. Alarm Threshold Control Register 0

Offset: 0x50			Register Name: THS_ALARM_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x51A	ALARM0_T_HOT. Thermal sensor 0 Alarm Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFF	Reserved to 0xFFF

#### 3.21.3.5. Alarm Threshold Control Register 1

Offset: 0x54			Register Name: THS_ALARM_CTRL
Bit	Read/Write	Default/Hex	Description

31:28	/	/	/
27:16	R/W	0x51A	ALARM1_T_HOT. Thermal sensor 1 Alarm Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.6. Alarm Threshold Control Register 2

<b>Offset: 0x58</b>			<b>Register Name: THS_ALARM_CTRL</b>
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x51A	ALARM2_T_HOT. Thermal sensor 2 Alarm Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.7. Alarm Threshold Control Register 3

<b>Offset: 0x5c</b>			<b>Register Name: THS_ALARM_CTRL</b>
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x51A	ALARM3_T_HOT. Thermal sensor 3 Alarm Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.8. Shutdown Threshold Control Register 0

<b>Offset: 0x60</b>			<b>Register Name: THP_SHUTDOWN_CTRL</b>
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

27:16	R/W	0x3F0	SHUT0_T_HOT. Thermal sensor 0 Shutdown Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.9. Shutdown Threshold Control Register 1

Offset: 0x64			Register Name: THP_SHUTDOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x3F0	SHUT1_T_HOT. Thermal sensor 1 Shutdown Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.10. Shutdown Threshold Control Register 2

Offset: 0x68			Register Name: THP_SHUTDOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x3F0	SHUT2_T_HOT. Thermal sensor 2 Shutdown Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFFF	Reserved to 0xFFFF

### 3.21.3.11. Shutdown Threshold Control Register 3

Offset: 0x6c			Register Name: THP_SHUTDOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x3F0	SHUT3_T_HOT.

			Thermal sensor 3 Shutdown Threshold for hot temperature
15:12	/	/	/
11:0	R	0xFFF	Reserved to 0xFFF

### 3.21.3.12. Average Filter Control Register

Offset: 0x70			Register Name: THS_FILTER
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	FILTER_EN.  Filter Enable  0: Disable  1: Enable
1:0	R/W	0x1	FILTER_TYPE.  Average Filter Type  00: 2  01: 4  10: 8  11: 16

### 3.21.3.13. Thermal Sensor0\_1 Calibration Data Register

Offset: 0x74			Register Name: THS0_1_CDAT
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x800	THES1_CDAT.  Thermal Sensor 1 calibration data
15:12	/	/	/

11:0	R/W	0x800	THES0_CDATA.  Thermal Sensor 0 calibration data
------	-----	-------	---

### 3.21.3.14. Thermal Sensor2\_3 Calibration Data Register

Offset: 0x78			Register Name: THS2_3_CDAT
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x800	THES3_CDATA.  Thermal Sensor 1 calibration data
15:12	/	/	/
11:0	R/W	0x800	THES2_CDATA.  Thermal Sensor 0 calibration data

### 3.21.3.15. THS0 Data Register

Offset: 0x80			Register Name: THS0_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	THS0_DATA.  Temperature measurement data of sensing point 0

### 3.21.3.16. THS1 Data Register

Offset: 0x84			Register Name: THS1_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	THS1_DATA.  Temperature measurement data of sensing point 1

**3.21.3.17. THS2 Data Register**

Offset: 0x88			Register Name: THS2_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	THS2_DATA. Temperature measurement data of sensing point 2

**3.21.3.18. THS3 Data Register**

Offset: 0x8c			Register Name: THS3_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	THS3_DATA. Temperature measurement data of sensing point 3

### 3.21.4. Programming Guide

- 1) Configure the GPADC Clock Control Register, set Bit 17 of APB0 Module Clock Gating Register and Bit 17 of APB0 Module Software Reset Register to select the clock source.
- 2) Configure THS Interrupt Control Register to set the THERMAL\_PER and IRQ
- 3) Configure the Alarm threshold Control Register and Shutdown threshold Control Register to set the ALARM\_T\_HOT and SHUT\_T\_HOT.
- 4) Configure THS Control Register to set the SENSOR\_ACQ and enable the sensors
- 5) Before enable the thermal sensor, we need get the Thermal Sensor Calibration Data

(THES0\_CDATA) = (0x06003844~0x06003845)

(THES1\_CDATA) = (0x06003846~0x06003847)

(THES2\_CDATA)= (0x06003848~0x06003849)

(THES3\_CDATA) = (0x0600384A~0x0600384C)

- 6) The real temperature value of each sensor is Tem, then

$$\text{Tem} = (\text{THS\_DATA}-2794) / -14.882$$

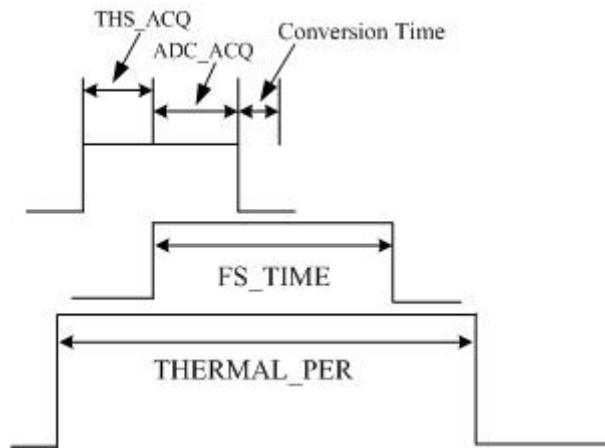
Temperature/Data relationships

TEMP	DIGITAL OUTPUT(Binary)	DIGITAL OUTPUT(Hex)
+125°C	001110100110	0x3A6
+113°C	010001011001	0x459
+85°C	010111111001	0x5F9
+60°C	011101101110	0x76E
+25°C	100101110110	0x976
-20°C	110000010100	0xC14

Reading back the temperature from the temperature value register requires a 2-byte read. Use 12-bit temperature data format.

- 7) Timing

Thermal sensor work timing must be like this:





### 3.22. KeyADC

#### 3.22.1. Overview

KeyADC is 6-bit resolution ADC for key application. The KEYADC can work up to 250Hz conversion rate.

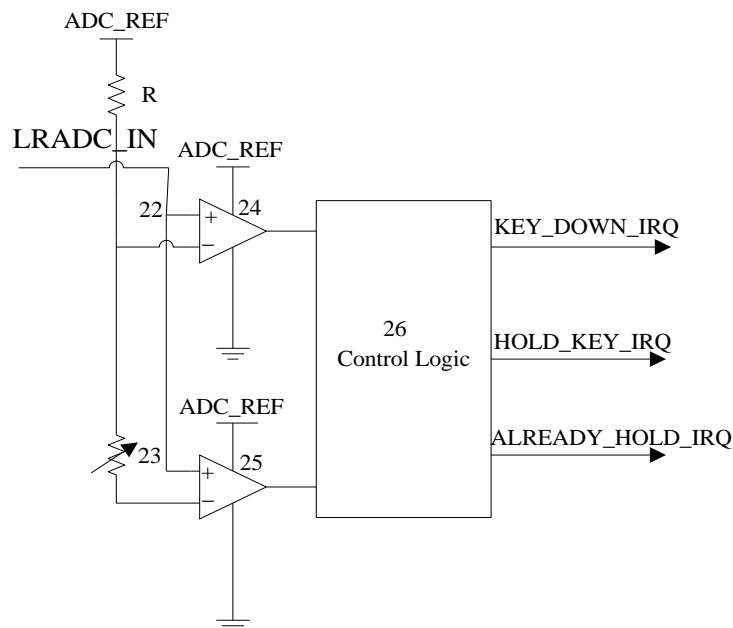
Features:

- Supports APB 32-bits bus width
- Support interrupt
- Support Hold Key and General Key
- Support Single Key and Continue Key mode
- Support 6-bits resolution
- Voltage input range between 0V to 1.35V
- Sample rate up to 250Hz

#### 3.22.2. Principle of operation

The KEYADC converted data can accessed by interrupt and polling method. If software can't access the last converted data instantly, the new converted data would update the old one at new sampling data.

Hold Key and General Key Function Introduction



When ADC\_IN Signal change from 1.8V to less than 1.35V (Level A), the comparator24 send first interrupt to control logic; When ADC\_IN Signal change from 1.35V to less than certain level (Program can set), the comparator25 give second interrupt. If the control Logic get the first interrupt, In a certain time range (program can set), doesn't get second interrupt, it will send hold key interrupt to the host; If the control Logic get the first interrupt, In a certain time range (program can set), get second interrupt, it will send key down interrupt to the host; If the control logic only get the second interrupt, doesn't get the first interrupt, it will send already hold interrupt to the host.

The KEYADC have three mode, Normal Mode、Single Mode and Continue Mode. Normal mode is that the KEYADC will report the result data of each convert all the time when the key is down. Single Mode is that the KEYADC will only report the first convert result data when the key is down. Continue Mode is that the KEYADC will report one of  $8*(N+1)$  (N is program can set) sample convert result data when key is down.

The KEYADC is support four sample rate such as 250Hz、125Hz、62.5Hz and 32.25Hz, you can configure the value of KEYADC\_SAMPLE\_RATE to select the fit sample rate.

### 3.22.3. KEYADC Register List

Module Name	Base Address
KEYADC	0x06001800

Register Name	Offset	Description
KEYADC_CTRL	0x00	KEYADC Control Register
KEYADC_INTC	0x04	KEYADC Interrupt Control Register
KEYADC_INTS	0x08	KEYADC Interrupt Status Register
KEYADC_DATA0	0x0c	KEYADC Data Register 0
KEYADC_DATA1	0x10	KEYADC Data Register 1

### 3.22.4. KEYADC Register Description

#### 3.22.4.1. KEYADC Control Register

Offset: 0x00			Register Name: KEYADC_CTRL
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x1	FIRST_CONVERT_DLY.  ADC First Convert Delay setting, ADC conversion is delayed by n samples
23:22	R/W	0x0	ADC_CHAN_SELECT.  ADC channel select  00: ADC0 channel  01: ADC1 channel  1x: ADC0&ADC1 channel
21:20	/	/	/
19:16	R/W	0x0	CONTINUE_TIME_SELECT.  Continue Mode time select, one of 8*(N+1) sample as a valuable sample data
15:14	/	/	/
13:12	R/W	0x0	KEY_MODE_SELECT.  Key Mode Select:  00: Normal Mode  01: Single Mode  10: Continue Mode
11:8	R/W	0x1	LEVELA_B_CNT.  Level A to Level B time threshold select, judge ADC convert value in level A to level B in n+1 samples
7	/	/	/
6	R/W	0x1	KEYADC_HOLD_EN.  KEYADC Sample hold Enable  0: Disable

			1: Enable
5: 4	R/W	0x2	LEVELB_VOL.  Level B Corresponding Data Value setting (the real voltage value)  00: 0x3C (~1.285v)  01: 0x39 (~1.221v)  10: 0x36 (~1.157v)  11: 0x33 (~1.092v)
3: 2	R/W	0x2	KEYADC_SAMPLE_RATE.  KEYADC Sample Rate  00: 250 Hz  01: 125 Hz  10: 62.5 Hz  11: 32.25 Hz
1	/	/	/
0	R/W	0x0	KEYADC_EN.  KEYADC enable  0: Disable  1: Enable

### 3.22.4.2. KEYADC Interrupt Control Register

Offset: 0x04			Register Name: KEYADC_INTC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
12	R/W	0x0	ADC1_KEYUP_IRQ_EN.  ADC 1 Key Up IRQ Enable

			0: Disable 1: Enable
11	R/W	0x0	ADC1_ALRDY_HOLD_IRQ_EN.  ADC 1 Already Hold Key IRQ Enable  0: Disable 1: Enable
10	R/W	0x0	ADC 1 Hold Key IRQ Enable  0: Disable 1: Enable
9	R/W	0x0	ADC1_KEYIRQ_EN.  ADC 1 Key IRQ Enable  0: Disable 1: Enable
8	R/W	0x0	ADC1_DATA_IRQ_EN.  ADC 1 DATA IRQ Enable  0: Disable 1: Enable
7:5	/	/	/
4	R/W	0x0	ADC0_KEYUP_IRQ_EN.  ADC 0 Key Up IRQ Enable  0: Disable 1: Enable
3	R/W	0x0	ADC0_ALRDY_HOLD_IRQ_EN.  ADC 0 Already Hold IRQ Enable  0: Disable 1: Enable
2	R/W	0x0	ADC0_HOLD_IRQ_EN.

			ADC 0 Hold Key IRQ Enable  0: Disable  1: Enable
1	R/W	0x0	ADC0_KEYDOWN_EN  ADC 0 Key Down Enable  0: Disable  1: Enable
0	R/W	0x0	ADC0_DATA_IRQ_EN.  ADC 0 Data IRQ Enable  0: Disable  1: Enable

### 3.22.4.3. KEYADC Interrupt Status Register

Offset: 0x08			Register Name: KEYADC_INTS
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	ADC1_KEYUP_PENDING.  ADC 1 Key up pending Bit  When general key pull up, it the corresponding interrupt is enabled.  0: No IRQ  1: IRQ Pending  Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable
11	R/W	0x0	ADC1_ALRDY_HOLD_PENDING.  ADC 1 Already Hold Pending Bit  When hold key pull down and pull the general key down, if the corresponding interrupt is enabled.

			<p>0: No IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable</p>
10	R/W	0x0	<p>ADC1_HOLDKEY_PENDING.</p> <p>ADC 1 Hold Key pending Bit</p> <p>When Hold key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled.</p> <p>0: NO IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.</p>
9	R/W	0x0	<p>ADC1_KEYDOWN_IRQ_PENDING.</p> <p>ADC 1 Key Down IRQ Pending Bit</p> <p>When General key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled.</p> <p>0: No IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.</p>
8	R/W	0x0	<p>ADC1_DATA_IRQ_PENDING.</p> <p>ADC 1 Data IRQ Pending Bit</p> <p>0: No IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.</p>
7:5	/	/	/
4	R/W	0x0	<p>ADC0_KEYUP_PENDING.</p>



			<p>ADC 0 Key up pending Bit</p> <p>When general key pull up, it the corresponding interrupt is enabled.</p> <p>0: No IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable</p>
3	R/W	0x0	<p>ADC0_ALRDY_HOLD_PENDING.</p> <p>ADC 0 Already Hold Pending Bit</p> <p>When hold key pull down and pull the general key down, if the corresponding interrupt is enabled.</p> <p>0: No IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable</p>
2	R/W	0x0	<p>ADC0_HOLDKEY_PENDING.</p> <p>ADC 0 Hold Key pending Bit</p> <p>When Hold key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled.</p> <p>0: NO IRQ</p> <p>1: IRQ Pending</p> <p>Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.</p>
1	R/W	0x0	<p>ADC0_KEYDOWN_PENDING.</p> <p>ADC 0 Key Down IRQ Pending Bit</p> <p>When General key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled.</p> <p>0: No IRQ</p>

			1: IRQ Pending  Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.
0	R/W	0x0	ADCO_DATA_PENDING.  ADC 0 Data IRQ Pending Bit  0: No IRQ  1: IRQ Pending  Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enable.

#### 3.22.4.4. KEYADC Data Register 0

<b>Offset: 0x0c</b>			<b>Register Name: KEYADC_DATA0</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default/Hex</b>	<b>Description</b>
31:6	/	/	/
5:0	R	0x0	KEYADCO_DATA.  KEYADC 0 Data

#### 3.22.4.5. KEYADC Data Register 1

<b>Offset: 0x10</b>			<b>Register Name: KEYADC_DATA1</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default/Hex</b>	<b>Description</b>
31:6	/	/	/
5:0	R	0x0	KEYADC1_DATA.  KEYADC 1 Data

### 3.23. R\_PRCM

#### 3.23.1. Overview

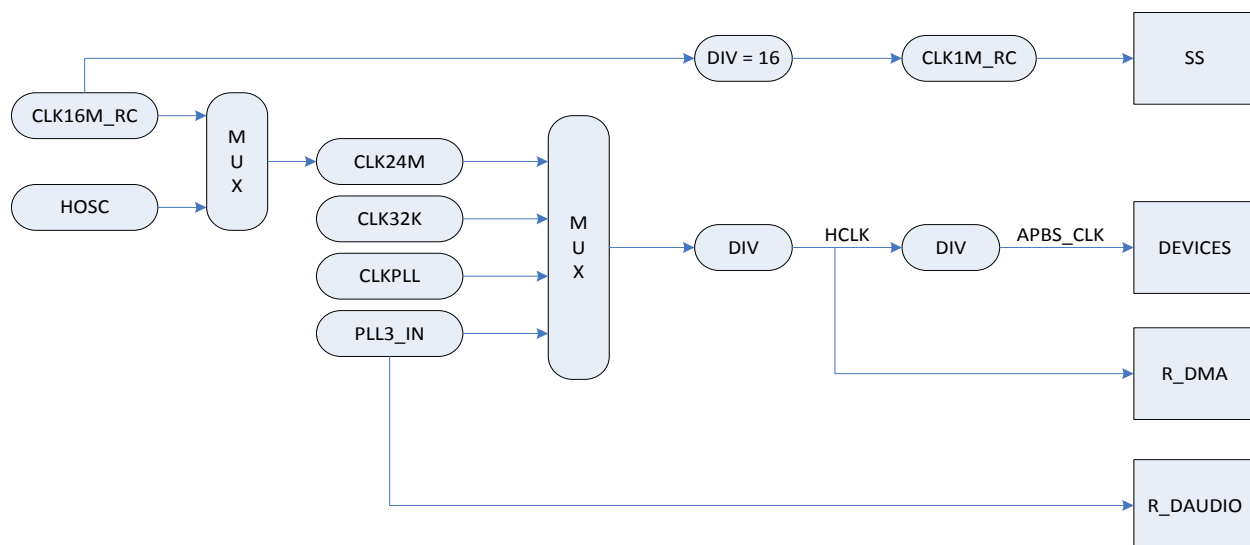
The R\_PRCM module is one of the most important design aspects in this system. It provides a versatile supporting multiple power-management techniques. And it also manages the gating and enabling of the clocks to the device modules.

The system-level reset management provides correct reset routing and sequencing when one or more devices are stacked together in the same package. The device-level reset management provides reset routing to relevant devices, such as r\_timer, r\_uart, r\_dma and so on.

The R\_PRCM includes the following features:

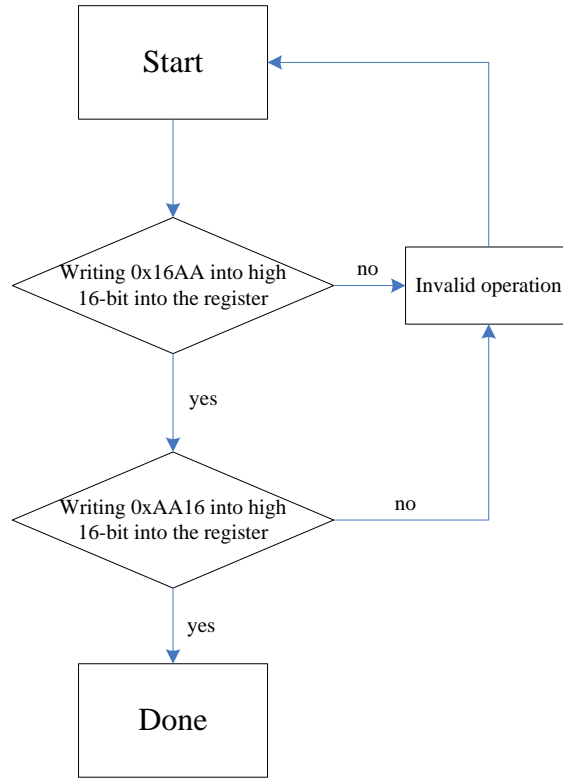
- Support CPUS, Cluster 0/1 CPUs and VDD\_SYS power domain modules reset control
- Support CPUS clock configuration
- Support devices clock gating, division and software reset in the CPUS domain
- Support Cluster 0/1 CPUs, system modules logic and GPU power off gating control
- Support Cluster 0/1 CPUs power switch control
- Support Super standby operations
- NMI interrupt control
- PLL\_AUDIO control
- Resistor calibration control

#### 3.23.2. Block Diagram



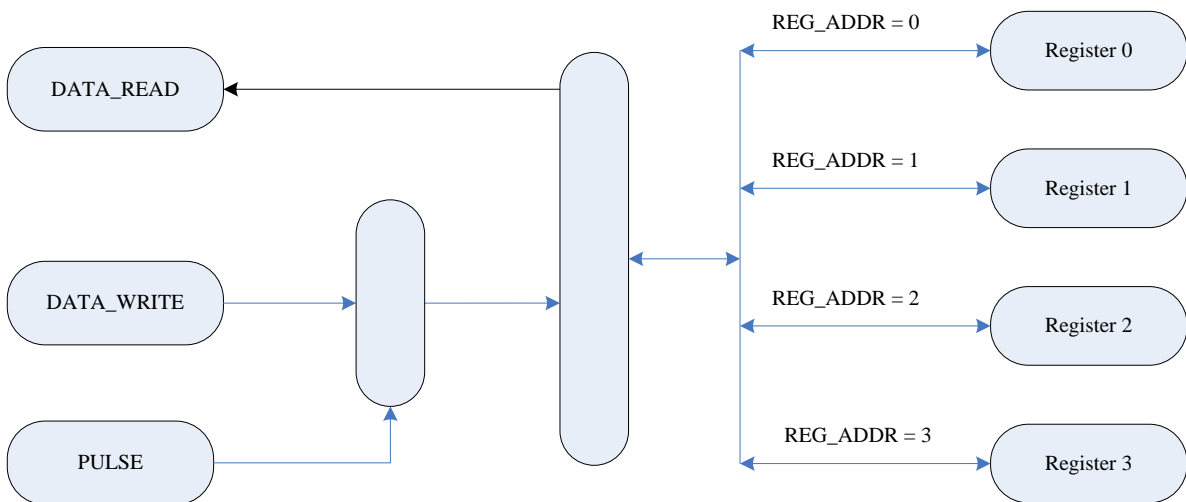
CPUS Clock Tree

The abbreviations of above figure refer to signal description. CLK1M\_RC is the oscillator source of TRNG in Security System.



Super Standby Flag Register Write Operation

The value of Super standby flag is defined by software, the active value is the low 16bit of Super Standby Flag Register; Before operating Super Standby Flag Register, the default value of low 16bit in the register is 0; It must be returned at the start operation process position after any invalid write-operation happened.



R\_PIO Pad Hold Register Write/Read Figure

There are four 8-bit registers in the RTC domain; The corresponding register can be selected through the bit[17:16] of **R\_PIO Hold Control Register**. DATA\_WRITE represents the value to be written in register; DATA\_READ represents the value to be read out from register.

A pulse is generated through writing bit[31] to 1 then to 0, the pulse is an enable condition of writing DATA\_WRITE.

It can hold GPIOL/M pad through writing 0xB in the Register0;the bit0 and bit1 of the register represent the ENH signal of GPIOL and GPIOM; bit3 represents BIAS\_EN signal that control the two group GPIO BIAS\_GEN circuit.Register1/2/3 have not been used.

### 3.23.3. Operation Modes

#### 3.23.3.1. HOSC

24MHz clock has two clock sources, one is from an external 24MHz crystal oscillator input, the other one is about 16MHz internal RC oscillator. In the super standby mode, CPUS uses 16MHz clock source to save the power consumption of 24MHz crystal, also make CPUS operate at a high frequency, fewer super standby wake-up time, improve the response speed. There is a simple operation process:

- (1) Enable 16MHz RC oscillator: The bit[0] of **OSC24M Control Register** is set to 1;
- (2) Switch HOSC clock source: The bit[1] of **OSC24M Control Register** is set to 1;
- (3) Close the external 24MHz oscillator: The bit[2] of **PLL Control Register 1** is cleared;

The next step can be operated before the previous step has been taken effect.

#### 3.23.3.2. PLL Gating

In the super standby mode,PLL\_AUDIO can selectively work with R\_DAUDIO,R\_DMA,other modules of CPUS area to achieve low-power super standby call function that talking standby. When super standby, PLL\_AUDIO control register is transferred from the CCU module to PRCM module,which can be switched by **PLL\_AUDIO Control Switch Register**.When there is only 24MHz,the level shift gating and common circuit gating of **VDD\_SYS Power Off Gating Register** need be cleared; When PLL\_AUDIO works, the level shift gating ,common circuit gating and PLL\_AUDIO gating need be cleared;if all PLL module is closed, the 5 bits of PLLVCC\_Gating can set.

### 3.23.3.3. Pad Hold

In the super standby mode,GPIO A/B/C/D/E/F/G/H of VDD\_SYS domain can hold a particular output state through GPIO pad hold function,but VCC power of GPIO can not he closed at this time;after exiting the super standby;if you need to use these GPIOs,need to be reconfigured.In power down,GPIO L and GPIO M can hold status through power off hold function;after each time system powered up,Using GPIO L/M need to reset **R\_PIO Hold Control Register**,so that avoid the previous power down residual configuration. Simple operation is as the following:

- (1) GPIO A/B/G/H pad hold: the bit[12] of **VDD\_SYS Power Off Gating Register** can set to 1.
- (2) GPIO L/M pad hold: Please refer **R\_PIO Pad Hold diagram** and **R\_PIO Hold Control Register**.

### 3.23.3.4. Power Switch

Each CPU power switch is composed by eight chains, there are chain0~chain7, which respectively correspond bit[0]~bit[7] of **Cluster CPU Power Switch Register**.Chain0~chain6 is belong to daisy chain, mainly for CPU charge to reduce the impact of the PMU power load.Chain7 is belong to fishbone chain, mainly for CPU core circuitry power supply.When CPU starts to power up,the corresponding value of **Cluster CPU Power Switch Register** should be 0xFF,namely power off;then successively write 0xFE、0xFC、0xF8、0xF0、0x00 to the register, there is a 10us delay after each write operation completed, then it starts to the next write operation. When the CPU power down, you can be directly written 0xFF to the register.

### 3.23.3.5. Signal Description

Signal Name	Direction	Description
CLK16M_RC	Input	16MHz clock input
CLK24M	Input	24MHz clock input
CLK32K	Input	External 32.768KHz clock input
PLL3_IN	Input	PLL_AUDIO clock
CLKPLL	Input	PLL_PERIPH clock
HCLK	Output	CPUS AHB bus clock
CLK1M_RC	Output	CLK1N_RC divide from CLK16M_RC

### 3.23.3.6. PLL\_AUDIO Control Switch

PRCM contains PLL\_AUDIO corresponding register,the register format and CCU is the same; When the cluster CPUX domain and the corresponding system module power down, PLL\_AUDIO still can output clock, DAUDIO of CPUS domain can work properly. PLL\_AUDIO Control Switch Register can switch the control area of PLL\_AUDIO.

### 3.23.3.7. CPUS Clock Source

CPUS contains four sources, including CLK32K from SOC external AC100 output clock ,frequency is generally 32.768KHz;the other three clock sources are OSC24M、PLL\_PERIPH/CPUS\_POST\_DIV、PLL\_AUDIO; it should firstly make sure clock source and the division factor valid before switching the clock sources.

### 3.23.3.8. OSC24 Control

RTC module of CPUS domain contains a RC oscillator about 16MHz.The frequency is divided to SS module for generating a true random number. At the same time,24MHz clock source can be switched to 16MHz,which need to effectively configure OSC24M control Register.

### 3.23.3.9. Super Standby

Whether VDD\_SYS powered down is the distinguishing mark of Super standby mode. In the super standby mode, Cluster CPUX, GPU, VE, ISP, DE and the vast majority of system modules is in power off state, but DRAM, PLL, USB, GPIO and analog modules have selective to maintain a certain status . CPUS system can also operate at a lower frequency, or low frequency clock input from outside, or 24MHz, or 16MHz; when CPUS running at 16MHz frequency, high frequency external 24MHz crystal oscillator can be closed.

To Write correctly super standby flag operations, please refer to <Super Standby Flag Register writing Process> Figure.

Enter super standby process, boot CPU (Cluster0 CPU0) writes an effective super standby flag; when boot CPU execution BROM code,it will firstly determine that if super standby flag is valid, if valid will enter super standby wake flow through CPU Software Entry Register 1 jumping to the corresponding address.

### 3.23.3.10. NMI IRQ

NMI is just an ordinary software interrupt,rather than hardware unmasked interrupt.Its interrupt trigger source can be changed depending on the actual application,the default is low level trigger. Either NMI interrupt sent to Cluster CPUX, or to CPUS;NMI interrupt can be used as a wake-up source from CPUS to super standby.

### 3.23.4. R\_PRCM Register List

Module Name	Base Address
R_PRCM	0x08001400

Register Name	Offset	Description
CPUS_RST_REG	0x0000	CPUS Reset Register
C0_CPUX_RST_CTRL	0x0004	Cluster0 CPUX Reset Control Register
C1_CPUX_RST_CTRL	0x0008	Cluster1 CPUX Reset Control Register
CPUS_CLK_CFG_REG	0x0010	CPUS Clock Configuration Register
APBS_CLK_DIV_REG	0x001C	APBS Clock Divide Register
APBS_CLK_GATING_REG	0x0028	APBS Clock Gating Register
PLL_CTRL_REG1	0x0044	PLL Control Register 1
R_ONE_WIRE_CLK_REG	0x0050	R_ONE_WIRE Clock Register
R_CIR_RX_CLK_REG	0x0054	R_CIR_RX Clock Register
R_DAUDIO0_CLK_REG	0x0058	R_DAUDIO0 Clock Register
R_DAUDIO1_CLK_REG	0x005C	R_DAUDIO1 Clock Register
APBS_SOFT_RST_REG	0x00B0	APBS Software Reset Register
COCPUX_PWROFF_GATING_REG	0x0100	Cluster0 CPUX Power Off Gating Register
C1CPUX_PWROFF_GATING_REG	0x0104	Cluster1 CPUX Power Off Gating Register
VDD_SYS_PWROFF_GATING_REG	0x0110	VDD_SYS Power Off Gating Register
GPU_PWROFF_GATING_REG	0x0118	GPU Power Off Gating Register
VDD_SYS_RST_REG	0x0120	VDD_SYS Reset Register
COCPU0_PWR_SWITCH_REG	0x0140	Cluster0 CPU0 Power Switch Register
COCPU1_PWR_SWITCH_REG	0x0144	Cluster0 CPU1 Power Switch Register
COCPU2_PWR_SWITCH_REG	0x0148	Cluster0 CPU2 Power Switch Register
COCPU3_PWR_SWITCH_REG	0x014C	Cluster0 CPU3 Power Switch Register
C1CPU0_PWR_SWITCH_REG	0x0150	Cluster1 CPU0 Power Switch Register
C1CPU1_PWR_SWITCH_REG	0x0154	Cluster1 CPU1 Power Switch Register
C1CPU2_PWR_SWITCH_REG	0x0158	Cluster1 CPU2 Power Switch Register
C1CPU3_PWR_SWITCH_REG	0x015C	Cluster1 CPU3 Power Switch Register
SUPER_STB_FLAG_REG	0x0160	Super Standby Flag Register
CPU_SOFT_ENTRY_REG	0x0164	CPU Software Entry Register
SUPER_STB_SOFT_ENTRY_REG	0x0168	Super Standby Software Entry Register
NMI_IRQ_CTRL_REG	0x01A0	NMI IRQ Control Register
NMI_IRQ_EN_REG	0x01A4	NMI IRQ Enable Register
NMI_IRQ_STATUS_REG	0x01A8	NMI IRQ Status Register
PLL_AUDIO_CTRL_REG	0x01C0	PLL_AUDIO Control Register
PLL_AUDIO_BIAS_REG	0x01C4	PLL_AUDIO Bias Register
PLL_AUDIO_PAT_CFG_REG	0x01C8	PLL_AUDIO Pattern Control Register
PLL_AUDIO_CTRL_SWITCH_REG	0x01CC	AUDIO_PLL Control Switch Register
R_PIO_HOLD_CTRL_REG	0x01F0	R_PIO Hold Control Register
OSC24M_CTRL_REG	0x01F4	OSC24M Control Register



### 3.23.5. R\_PRCM Register Description

#### 3.23.5.1. CPUS Reset Register(Default: 0x00000000)

Offset: 0x0000			Register Name: <b>CPUS_RST_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	CPUS_RESET. CPUS Reset Assert. 0: assert. 1: de-assert.

#### 3.23.5.2. Cluster0 CPUX Reset Register (Default: 0x0000000F)

Offset: 0x0004			Register Name: <b>CO_CPUX_RST_REG</b>
Bit	R/W	Default/Hex	Description
31:4	/	/	/
3:0	R/W	0xF	CO_CPUX_RESET. Cluster0 CPUX Reset Assert. These power-on reset signals initialize a processor all logic, including CPU Debug, breakpoint and watch point logic in the processor power domains. 0: assert. 1: de-assert.

#### 3.23.5.3. Cluster1 CPUX Reset Register (Default: 0x0000000F)

Offset: 0x0008			Register Name: <b>C1_CPUX_RST_REG</b>
Bit	R/W	Default/Hex	Description
31:4	/	/	/
3:0	R/W	0xF	C1_CPUX_RESET. Cluster1 CPUX Reset Assert. These power-on reset signals initialize a processor all logic, including CPU Debug, breakpoint and watch point logic in the processor power domains. 0: assert. 1: de-assert.

**3.23.5.4. CPUS Clock Configuration Register (Default: 0x00010000)**

Offset: 0x0010			Register Name: <b>CPUS_CLK_CFG_REG</b>
Bit	R/W	Default/Hex	Description
31:18	/	/	/
17:16	R/W	0x1	CPUS_CLK_SRC_SEL. CPUS Clock Source Select. 00: X32KI 01: OSC24M 10: PLL_PERIPH0/ CPUS_POST_DIV. 11: PLL_AUDIO Note: CPUS_CLK = CPUS Clock Source / CPUS_CLK_RATIO
15:13	/	/	/
12:8	R/W	0x0	CPUS_POST_DIV 00000: /1 00001: /2 00010: /3 ..... 11111: /32
7:6	/	/	/
5:4	R/W	0x0	CPUS_CLK_RATIO 00: /1 01: /2 10: /3 11: /4
3:0	/	/	/

**3.23.5.5. APBS Clock Divide Register (Default: 0x00000000)**

Offset: 0x001C			Register Name: <b>APBS_CLK_DIV_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	APBS_CLK_RATIO. APBS Clock divide ratio. APBS clock source is AHBS clock and the AHBS clock is equal to CPUS clock. 00: /1 01: /2 10: /3 11: /4

**3.23.5.6. APBS Clock Gating Register (Default: 0x00100000)**

Offset: 0x0028			Register Name: <b>APBS_CLK_GATING_REG</b>
Bit	R/W	Default/Hex	Description
31:21	/	/	/
20	R/W	0x1	R_TWD_GATING. Gating APBS clock for R_TWD. 0: Mask. 1: Pass.
19	/	/	/
18	R/W	0x0	R_DAUDIO1_GATING. Gating APBS clock for R_DAUDIO1. 0: Mask. 1: Pass.
17	R/W	0x0	R_DAUDIO0_GATING. Gating APBS clock for R_DAUDIO0. 0: Mask. 1: Pass.
16	R/W	0x0	R_DMA_GATING. Gating APBS clock for R_DMA. 0: Mask. 1: Pass.
15:14	/	/	/
13	R/W	0x0	R_PS2_1_GATING. Gating APBS clock for R_PS2_1. 0: Mask. 1: Pass.
12	R/W	0x0	R_PS2_0_GATING. Gating APBS clock for R_PS2_0. 0: Mask. 1: Pass.
11:8	/	/	/
7	R/W	0x0	R_TWI1_GATING. Gating APBS clock for R_TWI1. 0: Mask. 1: Pass.
6	R/W	0x0	R_TWI0_GATING. Gating APBS clock for R_TWI0. 0: Mask. 1: Pass.
5	R/W	0x0	R_ONE_WIRE_GATING. Gating APBS clock for R_ONE_WIRE. 0: Mask.

			1: Pass.
4	R/W	0x0	R_UART_GATING. Gating APBS clock for R_UART. 0: Mask. 1: Pass.
3	R/W	0x0	R_RSB_GATING. Gating APBS clock for R_RSB. 0: Mask. 1: Pass.
2	R/W	0x0	R_TIMER_GATING. Gating APBS clock for R_TIMER. 0: Mask. 1: Pass.
1	R/W	0x0	R_CIR_RX_GATING. Gating APBS clock for R_CIR-RX. 0: Mask. 1: Pass.
0	R/W	0x0	R_PIO_GATING. Gating APBS clock for R_PIO. 0: Mask. 1: Pass.

### 3.23.5.7. PLL Control Register 1 (Default: 0x00040015)

Offset: 0x0044			Register Name: <b>PLL_CTRL_REG1</b>
Bit	R/W	Default/Hex	Description
31:24	R/W	0x0	KEY_FIELD. Key Field for LDO Enable bit. If the key field value is 0xA7, the bit[23:0] can be modified.
23:3	/	/	/
2	R/W	1	OSC24M_EN. External crystal OSC24M enable.
1	/	/	/
0	R/W	1	LDO_EN. PLL power enable, the power source from VCC-PLL. 0: disable. 1: enable.

### 3.23.5.8. R\_ONE\_WIRE Clock Register (Default: 0x00000000)

Offset: 0x0050	Register Name: <b>R_ONE_WIRE_CLK_REG</b>
----------------	--

Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating SCLK (Max Clock = 100MHz). 0: Clock is OFF. 1: Clock is ON. The SCLK = Clock Source/Divider N/Divider M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL. Clock source select. 00: X32KI. 01: OSC24M. 10: / 11: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n). The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.
15:5	/	/	/
4:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 32.

### 3.23.5.9. R\_CIR\_RX Clock Register (Default: 0x00000000)

Offset: 0x0054			Register Name: R_CIR_RX_CLK_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating SCLK (Max Clock = 100MHz). 0: Clock is OFF. 1: Clock is ON. The SCLK = Clock Source/Divider N/Divider M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL. Clock source select. 00: X32KI. 01: OSC24M. 10: / 11: /
23:18	/	/	/
17:16	R/W	0x0	CLK_DIV_RATIO_N. Clock pre-divide ratio (n). The select clock source is pre-divided by 2 <sup>n</sup> . The divider is 1/2/4/8.

15:5	/	/	/
4:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 32.

### 3.23.5.10. R\_DAUDIO0 Clock Register (Default: 0x00000000)

Offset: 0x0058			Register Name: <b>R_DAUDIO0_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. The SCLK = Clock Source PLL_AUDIO/Divider M.
30:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

### 3.23.5.11. R\_DAUDIO1 Clock Register (Default: 0x00000000)

Offset: 0x005C			Register Name: <b>R_DAUDIO1_CLK_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SCLK_GATING. Gating SCLK (Max Clock = 200MHz). 0: Clock is OFF. 1: Clock is ON. The SCLK = Clock Source PLL_AUDIO/Divider M.
30:4	/	/	/
3:0	R/W	0x0	CLK_DIV_RATIO_M. Clock divide ratio (m). The pre-divided clock is divided by (m+1). The divider is from 1 to 16.

### 3.23.5.12. APBS Software Reset Register (Default: 0x00000000)

Offset: 0x00B0			Register Name: <b>APBS_SOFT_RST_REG</b>
Bit	R/W	Default/Hex	Description
31:19	/	/	/
18	R/W	0x0	R_DAUDIO1_RESET.

			Reset signal for R_DAUDIO1. 0: Assert. 1: De-assert.
17	R/W	0x0	R_DAUDIO0_RESET. Reset signal for R_DAUDIO0. 0: Assert. 1: De-assert.
16	R/W	0x0	R_DMA_RESET. Reset signal for R_DMA. 0: Assert. 1: De-assert.
15:14	/	/	/
13	R/W	0x0	R_PS2_1_RESET. Reset signal for R_PS2_1. 0: Assert. 1: De-assert.
12	R/W	0x0	R_PS2_0_RESET. Reset signal for R_PS2_0. 0: Assert. 1: De-assert.
11:8	/	/	/
7	R/W	0x0	R_TWI1_RESET. Reset signal for R_TWI1. 0: Assert. 1: De-assert.
6	R/W	0x0	R_TWI0_RESET. Reset signal for R_TWI0. 0: Assert. 1: De-assert.
5	R/W	0x0	R_ONE_WIRE_RESET. Reset signal for R_ONE_WIRE. 0: Assert. 1: De-assert.
4	R/W	0x0	R_UART_RESET. Reset signal for R_UART. 0: Assert. 1: De-assert.
3	R/W	0x0	R_RSB_RESET. Reset signal for R_RSB. 0: Assert. 1: De-assert.
2	R/W	0x0	R_TIMER_RESET. Reset signal for R_TIMER.

			0: Assert. 1: De-assert.
1	R/W	0x0	R_CIR_RX_RESET. Reset signal for R_CIR-RX. 0: Assert. 1: De-assert.
0	/	/	/

### 3.23.5.13. Cluster0 CPUX Power Off Gating Register (Default: 0x00000000)

Offset: 0x0100			Register Name: COCPUX_PWROFF_GATING_REG
Bit	R/W	Default/Hex	Description
31:5	/	/	/
4	R/W	0x0	CO_PWROFF_GATING. Gating the corresponding modules when Cluster0 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before Cluster0 power off, and it should be set to 0 after the Cluster0 power on.
3	R/W	0x0	COCPU3_PWROFF_GATING. Gating the corresponding modules when CPU3 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU3 power off while it should be set to 0 after the CPU3 power on.
2	R/W	0x0	COCPU2_PWROFF_GATING. Gating the corresponding modules when CPU2 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU2 power off while it should be set to 0 after the CPU2 power on.
1	R/W	0x0	COCPU1_PWROFF_GATING. Gating the corresponding modules when CPU1 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU1 power off while it should be set to 0 after the CPU1 power on.
0	R/W	0x0	COCPU0_PWROFF_GATING. Gating the corresponding modules when CPU0 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU0 power off while it should be set to 0



			after the CPU0 power on.
--	--	--	--------------------------

### 3.23.5.14. Cluster1 CPUX Power Off Gating Register (Default: 0x000001F)

Offset: 0x0104			Register Name: C1CPUX_PWROFF_GATING_REG
Bit	R/W	Default/Hex	Description
31:5	/	/	/
4	R/W	0x1	C1_PWROFF_GATING. Gating the corresponding modules when Cluster0 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before Cluster1 power off, and it should be set to 0 after the Cluster0 power on.
3	R/W	0x1	C1CPU3_PWROFF_GATING. Gating the corresponding modules when CPU3 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU3 power off while it should be set to 0 after the CPU3 power on.
2	R/W	0x1	C1CPU2_PWROFF_GATING. Gating the corresponding modules when CPU2 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU2 power off while it should be set to 0 after the CPU2 power on.
1	R/W	0x1	C1CPU1_PWROFF_GATING. Gating the corresponding modules when CPU1 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU1 power off while it should be set to 0 after the CPU1 power on.
0	R/W	0x1	C1CPU0_PWROFF_GATING. Gating the corresponding modules when CPU0 power off. 0: Invalid. 1: Valid. This bit should be set to 1 before CPU0 power off while it should be set to 0 after the CPU0 power on.

### 3.23.5.15. VDD\_SYS Power Off Gating Register (Default: 0x00000000)

Offset: 0x0110	Register Name: VDD_SYS_PWROFF_GATING_REG
----------------	--

Bit	R/W	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	VDDON_SYS_H3. 0: Not hold. 1: Hold. This bit is for display、HDMI、GPIO A/B/C/D/E/F/G/H_hold. This bit should be set to 1 before VDD_SYS power off while it should be set to 0 after the VDD_SYS power on, if it is necessary.
11	R/W	0x0	VDDON_SYS_H2. 0: Not hold. 1: Hold. This bit is for MIPI_CSI.
10	R/W	0x0	VDDON_SYS_H1. 0: Not hold. 1: Hold. This bit is for resistor calibration level shift.
9	R/W	0x0	VDDON_SYS_H0. 0: Not hold. 1: Hold. This bit is for U_boot pad ENH signal, LRADC, GPADC, MIPI_DSI.
8:4	R/W	0x0	PLLVCC_GATING. Gating the corresponding modules to the PLLVCC Power Domain when VDD_SYS power off. 0: Invalid. 1: Valid. When only use PLL_AUDIO(the control registers in PRCM module), bit[4], bit[5] and bit[8] need be set to 0. When use OSC24M, bit[4] and bit[8]need be set to 0. Otherwise, all these bit should be set to 1. Bit[8] : PLLVCC level shift gating. Bit[7] : Other PLLs gating. Bit[6] : PLL_PERIPH gating. Bit[5] : PLL_AUDIO gating. Bit[4] : Common circuit gating.
3	R/W	0x0	VDD_CPUS_GATING Gating the corresponding modules to the CPUS Power Domain( and USB power domain) when VDD_SYS power off. 0: Invalid. 1: Valid. This bit should be set to 1 before VDD_SYS power off while it should be set to 0 after the VDD_SYS power on, and CPUS clock source should switch to X32KI before CPU power off gating, if it is necessary.
2	R/W	0x0	VDD_VE_GATING. Gating the corresponding modules to the VE Power Domain.

			0: Invalid. 1: Valid.
1	R/W	0x0	DRAM_CH1_PAD_HOLD. Hold the pad of DRAM channel 1. 0:Not hold. 1:Hold DRAM Pad. Note: This bit should be set to 1 before VDD_SYS power off while it should be set to 0 after the VDD_SYS power on, if it is necessary.
0	R/W	0x0	DRAM_CH0_PAD_HOLD. Hold the pad of DRAM channel 0. 0:Not hold. 1:Hold DRAM Pad. This bit should be set to 1 before VDD_SYS power off while it should be set to 0 after the VDD_SYS power on, if it is necessary.

### 3.23.5.16. GPU Power Off Gating Register (Default: 0x00000000)

Offset: 0x0118			Register Name: <b>GPU_PWROFF_GATING_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	GPU_PWROFF_GATING. Gating the corresponding modules when GPU power off. 0: Invalid. 1: Valid. This bit should be set to 1 before GPU power off while it should be set to 0 after the GPU power on.

### 3.23.5.17. VDD\_SYS Reset Register (Default: 0x00000001)

Offset: 0x0120			Register Name: <b>VDD_SYS_RST_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	VDD_SYS_RESET. VDD_SYS power domain devices should be reset before VDD_SYS power on. 0: Assert. 1: De-assert.

**3.23.5.18. Cluster0 CPU0 Power Switch Register (Default: 0x00000000)**

Offset: 0x0140			Register Name: <b>COCPU0_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	COCPU0_PWR_SWITCH. Cluster0 CPU0 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

**3.23.5.19. Cluster0 CPU1 Power Switch Register (Default: 0x00000000)**

Offset: 0x0144			Register Name: <b>COCPU1_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	COCPU1_PWR_SWITCH. Cluster0 CPU1 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

**3.23.5.20. Cluster0 CPU2 Power Switch Register (Default: 0x00000000)**

Offset: 0x0148			Register Name: <b>COCPU2_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	COCPU2_PWR_SWITCH. Cluster0 CPU2 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

**3.23.5.21. Cluster0 CPU3 Power Switch Register (Default: 0x00000000)**

Offset: 0x014C			Register Name: <b>COCPU3_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	COCPU3_PWR_SWITCH.

			Cluster0 CPU3 Power Switch. 0x00: Power On. ... 0xFF: Power Off.
--	--	--	---

### 3.23.5.22. Cluster1 CPU0 Power Switch Register (Default: 0x000000FF)

Offset: 0x0150			Register Name: <b>C1CPU0_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xFF	C1CPU0_PWR_SWITCH. Cluster1 CPU0 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

### 3.23.5.23. Cluster1 CPU1 Power Switch Register (Default: 0x000000FF)

Offset: 0x0154			Register Name: <b>C1CPU1_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xFF	C1CPU1_PWR_SWITCH. Cluster1 CPU1 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

### 3.23.5.24. Cluster1 CPU2 Power Switch Register (Default: 0x000000FF)

Offset: 0x0158			Register Name: <b>C1CPU2_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xFF	C0CPU2_PWR_SWITCH. Cluster0 CPU2 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

**3.23.5.25. Cluster1 CPU3 Power Switch Register (Default: 0x000000FF)**

Offset: 0x015C			Register Name: <b>C1CPU3_PWR_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xFF	C1CPU3_PWR_SWITCH. Cluster1 CPU3 Power Switch. 0x00: Power On. ... 0xFF: Power Off.

**3.23.5.26. Super Standby Flag Register (Default: 0x00000000)**

Offset: 0x0160			Register Name: <b>SP_STB_FLAG_REG</b>
Bit	R/W	Default/Hex	Description
31:16	R/W	0x0	SP_STANDBY_FLAG. Key Field. Any value can be written and read back in the key field, but if the values are not appropriate, the lower 16 bits will not change in this register. Only follow the appropriate process, the super standby flag can be written in the lower 16 bits. Refer to <b>Super Standby</b> Diagram.
15:0	R/W	0x0	SP_STANBY_FLAG_DATA. When system is turned on, the value in the Super Standby Flag Register low 16 bits should be 0x0. If software programmer wants to write correct super standby flag ID in low 16 bits, the high 16 bits should be written with 0x16AA at first. Then, software programmer must write 0xAA16XXXX in the Super Standby Flag Register, the 'XXXX' means the correct super standby flag ID.

**3.23.5.27. CPU Software Entry Register (Default: 0x00000000)**

Offset: 0x0164			Register Name: <b>CPU_SOFT_ENTRY_REG</b>
Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	Boot CPU software entry register when acting from hot plug or Non-boot CPU software entry register.

**3.23.5.28. Super Standby Software Entry Register (Default: 0x00000000)**

Offset: 0x0168			Register Name: <b>SUPER_STB_SOFT_ENTRY_REG</b>
----------------	--	--	--

Bit	R/W	Default/Hex	Description
31:0	R/W	0x0	CPU software entry register when acting from supper standby.

### 3.23.5.29. NMI IRQ Control Register (Default: 0x00000000)

Offset: 0x01A0			Register Name: NMI_IRQ_CTRL_REG
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	NMI_SRC_TYPE. External NMI Interrupt Source Type. 00: Low level sensitive. 01: Negative edge triggered. 10: High level sensitive. 11: Positive edge sensitive.

### 3.23.5.30. NMI IRQ Enable Register (Default: 0x00000000)

Offset: 0x01A4			Register Name: NMI_IRQ_CTRL_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	NMI_IRQ_EN. 0: Disable. 1: Enable.

### 3.23.5.31. NMI IRQ Status Register (Default: 0x00000000)

Offset: 0x01A8			Register Name: NMI_IRQ_CTRL_REG
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	NMI_IRQ_PEND. NMI irq pending status. 0: No effect. 1: Pending. This bit will be pending when NMI interrupt occurred. Set 1 to this bit will clear it.

### 3.23.5.32. PLL\_AUDIO Control Register (Default: 0x00042B14)

Offset: 0x01C0			Register Name: PLL_AUDIO_CTRL_REG
----------------	--	--	-----------------------------------

Bit	R/W	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. This PLL is for Audio. The PLL Output = $24\text{MHz} * N / (\text{Input\_div} + 1) / (\text{Output\_div} + 1) / (P + 1)$ . Its default is 24.5714 MHz.
30:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23:19	/	/	/
18	R/W	0x1	PLL_OUTPUT_DIV. Div factor = 0 or 1.
17	/	/	/
16	R/W	0x0	PLL_INPUT_DIV. Div factor = 0 or 1.
15:8	R/W	0x2B	PLL_FACTOR_N. PLL Factor N. The range is from 12 to 255. In application, $N \geq 12$ .
7:6	/	/	/
5:0	R/W	0x14	PLL_POSTDIV_P. Post-div factor P, it is the post counter. The range is from 0 to 63.

### 3.23.5.33. PLL\_AUDIO Pattern Control Register (Default: 0x00000000)

Offset: 0x01C8			Register Name: PLL_AUDIO_PAT_CTRL_REG
Bit	R/W	Default/Hex	Description
31	R/W	0x0	SIGMA_DELTA_PAT_ENABLE. Sigma-delta pattern enable.
30:29	R/W	0x0	SPREAD_FREQ_MODE. Spread Frequency Mode. 00: DC=0. 01: DC=1. 10: Triangular (1 bit pattern). 11: Triangular (3 bit pattern).
28:20	R/W	0x0	WAVE_STEP. Wave step.
19	R/W	0x0	CLK_SRC_SELECT. 0: 24MHz.



			1: 12MHz.
18:17	R/W	0x0	FREQ_SELECT. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

### 3.23.5.34. PLL\_AUDIO Control Switch Register (Default: 0x00000000)

Offset: 0x01CC			Register Name: <b>PLL_AUDIO_CTRL_SWITCH_REG</b>
Bit	R/W	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	PLL_AUDIO_GATING. The gating of PLL_AUDIO registers. If this bit is set to 1, the PLL_AUDIO relevant registers are valid in PRCM module and the corresponding registers of CCMU are invalid. Before VDD_SYS powering off, this bit should be set to 1 to keep the PLL_AUDIO active.

### 3.23.5.35. R\_PIO Hold Control Register (Default: 0x00000000)

Offset: 0x01F0			Register Name: <b>R_PIO_HOLD_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31	R/W	0x0	WRITE_OPERATION_PULSE. Set 1 then set 0 to write the value of DATA_WRITE into the corresponding PIO Register that its address is allocated by PIO_REG_ADDR.
30:18	/	/	/
17:16	R/W	0x0	REG_ADDR. 00: R_PIO pad hold register. 01: / 10: / 11: / There are four 8-bit registers in RTC domain. Their address are successively 0x0、0x1、0x2、0x3.The different value of REG_ADDR shows the selected different register.Now only the 8-bit register in 0x0 address is used.
15:8	R/W	0x0	DATA_WRITE. The data write into the corresponding 8-bit register.
7:0	RO	0x0	DATA_READ. The value read from the corresponding 8-bit register.

### 3.23.5.36. OSC24M Control Register (Default: 0x00000000)

Offset: 0x01F4			Register Name: <b>OSC24M_CTRL_REG</b>
Bit	R/W	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	OSC24M_SRC_SELECT. 0: 24MHz, it is from external oscillator. 1: 16MHz, it is from RC oscillator in the SOC. Before selecting 16MHz OSC, the 16MHz OSC must be enabled.
0	R/W	0x0	OSC16M_ENABLE. 0: Disable. 1: Enable.

### 3.23.6. Programming Guidelines

- (1) Release the device's clock gating after software reset has been de-assert.
- (2) After the OSC24M switched from an external 24MHz clock source to the internal RC oscillator , the external crystal oscillator can be closed.
- (3) To switch device clock source need set reasonable and effective division factor.
- (4) Before power down or enter the super standby, the related GPIO pad hold need to set; then power up or exit super standby ,it is necessary to re-set the relevant registers..
- (5) If the super standby mode using PLL\_AUDIO, it must switch the control signal of PLL\_AUDIO to PRCM domain before VDD\_SYS powered down.
- (6) After wrote VDD\_SYS Power gating Register, gating can be valid after 1us delay; After wrote VDD\_SYS reset Register, reset can be valid after 1us delay.
- (7) After wrote GPU Power gating Register, gating can be valid after 1us delay.
- (8) After wrote Power Switch Register, power switch can be valid after 10us delay.
- (9) X32KI comes from the outside AC100 chip, usually its frequency is 32.768KHz; however AC100 also can output other frequencies to the pin.
- (10) Two CPU can not operate simultaneously **R\_PIO Hold Control Register**;

## 3.24. Port Controller(CPUx-Port)

### 3.24.1. Overview

The chip has 8 ports for multi-functional input/out pins. They are shown below:

- Port A(PA): 18 input/output port
- Port B(PB): 20 input/output port
- Port C(PC): 20 input/output port
- Port D(PD): 28 input/output port
- Port E(PE) : 21 input/output port
- Port F(PF) : 6 input/output port
- Port G(PG) : 16 input/output port
- Port H(PH) : 22 input/output port

For various system configurations, these ports can be easily configured by software. All these ports can be configured as GPIO if multiplexed functions are not used. The total 5 group external PIO interrupt sources are supported and interrupt mode can be configured by software.

### 3.24.2. Port Register List

Module Name	Base Address
PIO	0x06000800

Register Name	Offset	Description
Pn_CFG0	n*0x24+0x00	Port n Configuration Register 0 (n from 0 to 8)
Pn_CFG1	n*0x24+0x04	Port n Configuration Register 1 (n from 0 to 8)
Pn_CFG2	n*0x24+0x08	Port n Configuration Register 2 (n from 0 to 8)
Pn_CFG3	n*0x24+0x0C	Port n Configuration Register 3 (n from 0 to 8)
Pn_DAT	n*0x24+0x10	Port n Data Register (n from 0 to 8)
Pn_DRV0	n*0x24+0x14	Port n Multi-Driving Register 0 (n from 0 to 8)
Pn_DRV1	n*0x24+0x18	Port n Multi-Driving Register 1 (n from 0 to 8)
Pn_PUL0	n*0x24+0x1C	Port n Pull Register 0 (n from 0 to 8)
Pn_PUL1	n*0x24+0x20	Port n Pull Register 1 (n from 0 to 8)
Pn_INT_CFG0	0x200+n*0x20+0x00	PIO Interrupt Configuration Register 0
Pn_INT_CFG1	0x200+n*0x20+0x04	PIO Interrupt Configuration Register 1
Pn_INT_CFG2	0x200+n*0x20+0x08	PIO Interrupt Configuration Register 2
Pn_INT_CFG3	0x200+n*0x20+0x0C	PIO Interrupt Configuration Register 3
Pn_INT_CTL	0x200+n*0x20+0x10	PIO Interrupt Control Register
Pn_INT_STA	0x200+n*0x20+0x14	PIO Interrupt Status Register
Pn_INT_DEB	0x200+n*0x20+0x18	PIO Interrupt Debounce Register
Pn_GRP_CONFIG	0x300+n*0x04	PIO Group Configuration Register

### 3.24.3. Port Register Description

#### 3.24.3.1. PA Configuration Register 0

Offset: 0x00			Register Name: PA_CFG0
Default Value: 0x7777_7777			
Bit	Read/Write	Default	Description
31	/	/	/
			PA7_SELECT 000:Input                      001:Output 010:RMII-TXD3/MII-TXD3      011:Reserved 100:UART1_RING                101:Reserved
30:28	R/W	0x7	110:PA_EINT7                111:IO Disable
27	/	/	
			PA6_SELECT 000:Input                      001:Output 010:MII-RXERR                011:Reserved 100:UART1_DCD                101:Reserved
26:24	R/W	0x7	110:PA_EINT6                111:IO Disable
23	/	/	
			PA5_SELECT 000:Input                      001:Output 010:RGMII-RXCTL/MII-RXDV    011:Reserved 100:UART1_DSR                101:Reserved
22:20	R/W	0x7	110:PA_EINT5                111:IO Disable
19	/	/	
18:16	R/W	0x7	PA4_SELECT

			000:Input	001:Output
			010:RGMII-RXCK/MII-RXCK	011:Reserved
			100:UART1_DTR	101:Reserved
			110:PA_EINT4	111:IO Disable
15	/	/		
			PA3_SELECT	
			000:Input	001:Output
			010:RGMII-RXD0/MII-RXD0	011:Reserved
			100:UART1_CTS	101:Reserved
14:12	R/W	0x7	110:PA_EINT3	111:IO Disable
11	/	/		
			PA2_SELECT	
			000:Input	001:Output
			010:RGMII-RXD1/MII-RXD1	011:Reserved
			100:UART1_RTS	101:Reserved
10:8	R/W	0x7	110:PA_EINT2	111:IO Disable
7	/	/		
			PA1_SELECT	
			000:Input	001:Output
			010:RGMII-RXD2/MII-RXD2	011:Reserved
			100:UART1_RX	101:Reserved
6:4	R/W	0x7	110:PA_EINT1	111:IO Disable
3	/	/		
			PA0_SELECT	
			000:Input	001:Output
			010:RGMII-RXD3/MII-RXD3	011:Reserved
2:0	R/W	0x7		

			100:UART1_TX	101:Reserved
			110:PA_EINT0	111:IO Disable

### 3.24.3.2. PA Configuration Register 1

Offset: 0x04			Register Name: PA_CFG1	
Default Value: 0x7777_7777				
Bit	Read/Write	Default	Description	
31	/	/	/	
			PA15_SELECT	
			000:Input	001:Output
			010:RGMII-CLKIN/MII-COL	011:Reserved
			100:SPI1_CLK	101:Reserved
30:28	R/W	0x7	110:PA_EINT15	111:IO Disable
27	/	/	/	
			PA14_SELECT	
			000:Input	001:Output
			010:MII-TXERR	011:Reserved
			100:SPI1_CS0	101:Reserved
26:24	R/W	0x7	110:PA_EINT14	111:IO Disable
23	/	/	/	
			PA13_SELECT	
			000:Input	001:Output
			010:RGMII-TXCTL/MII-TXEN	011:Reserved
			100:PWM3_N	101:Reserved
22:20	R/W	0x7	110:PA_EINT13	111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PA12_SELECT	

			000:Input	001:Output
			010:RGMII-TXCK/MII-TXCK	011:Reserved
			100:PWM3_P	101:Reserved
			110:PA_EINT12	111:IO Disable
15	/	/		
			PA11_SELECT	
			000:Input	001:Output
			010:MII-CRS	011:Reserved
			100:CLKB_OUT	101:Reserved
14:12	R/W	0x7	110:PA_EINT11	111:IO Disable
11	/	/		
			PA10_SELECT	
			000:Input	001:Output
			010:RGMII-TXD0/MII-TXD0	011:Reserved
			100:CLKA_OUT	101:Reserved
10:8	R/W	0x7	110:PA_EINT10	111:IO Disable
7	/	/		
			PA9_SELECT	
			000:Input	001:Output
			010:RGMII-TXD1/MII-TXD1	011:Reserved
			100:ECLK_IN1	101:Reserved
6:4	R/W	0x7	110:PA_EINT9	111:IO Disable
3	/	/		
			PA8_SELECT	
			000:Input	001:Output
			010:RGMII-TXD2/MII-TXD2	011:Reserved
2:0	R/W	0x7		



			100:ECLK_IN0	101:Reserved
			110:PA_EINT8	111:IO Disable

### 3.24.3.3. PA Configuration Register 2

Offset: 0x08			Register Name: PA_CFG1	
Default Value: 0x0000_0077				
Bit	Read/Write	Default	Description	
31:8	/	/	/	
7	/	/		
			PA17_SELECT	
			000:Input	001:Output
			010:EMDIO	011:Reserved
			100:SPI1_MISO	101:Reserved
6:4	R/W	0x7	110:PA_EINT17	111:IO Disable
3	/	/		
			PA16_SELECT	
			000:Input	001:Output
			010:EMDC	011:Reserved
			100:SPI1_MOSI	101:Reserved
2:0	R/W	0x7	110:PA_EINT16	111:IO Disable

### 3.24.3.4. PA Configuration Register 3

Offset: 0x0C			Register Name: PA_CFG1	
Default Value: 0x0000_0000				
Bit	Read/Write	Default	Description	
31:0	/	/	/	

### 3.24.3.5. PA Data Register

<b>Offset: 0x10</b>			<b>Register Name: PA_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:18	/	/	/
17:0	R/W	0	PA_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.24.3.6. PA Multi-Driving Register 0

<b>Offset: 0x14</b>			<b>Register Name: PA_DRV0</b>
			<b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i] (i=0~15)	R/W	0x1	PA_DRV  PA[n] Multi-Driving Select (n = 0~15)  00: Level 0                      01: Level 1  10: Level 2                     11: Level 3

### 3.24.3.7. PA Multi-Driving Register 1

<b>Offset: 0x18</b>			<b>Register Name: PA_DRV1</b>
			<b>Default Value: 0x0000_0005</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
[2i+1:2i] (i=0~1)	R/W	0x1	PA_DRV  PA[n] Multi-Driving Select (n = 16~17)

			00: Level 0	01: Level 1
			10: Level 2	11: Level 3

### 3.24.3.8. PA Pull Register 0

<b>Offset: 0x1C</b>			<b>Register Name: PA_PULL0</b>	
			<b>Default Value: 0x0000_0000</b>	
Bit	Read/Write	Default	Description	
[2i+1:2i]			PA_PULL	
			PA[n] Pull-up/down Select (n = 0~15)	
			00: Pull-up/down disable	01: Pull-up
(i=0~15)	R/W	0x0	10: Pull-down	11: Reserved

### 3.24.3.9. PA Pull Register 1

<b>Offset: 0x20</b>			<b>Register Name: PA_PULL1</b>	
			<b>Default Value: 0x0000_0000</b>	
Bit	Read/Write	Default	Description	
31:4	/	/	/	
[2i+1:2i]			PA_PULL	
			PA[n] Pull-up/down Select (n = 16~17)	
			00: Pull-up/down disable	01: Pull-up enable
(i=0~1)	R/W	0x0	10: Pull-down	11: Reserved

### 3.24.3.10. PB Configuration Register 0

<b>Offset: 0x24</b>			<b>Register Name: PB_CFG0</b>	
			<b>Default Value: 0x7777_7777</b>	
Bit	Read/Write	Default	Description	
31:27	/	/	/	

			PB6_SELECT	
			000:Input	001:Output
			010:I2S0_DO3	011:UART3_RX
			100:Reserved	101:TRACE_DOUT6
26:24	R/W	0x7	110:PB_EINT6	111:IO Disable
23	/	/		
			PB5_SELECT	
			000:Input	001:Output
			010:I2S0_DO2	011:UART3_TX
			100:Reserved	101:TRACE_DOUT5
22:20	R/W	0x7	110:PB_EINT5	111:IO Disable
19:0	/	/	/	/

### 3.24.3.11. PB Configuration Register 1

<b>Offset: 0x28</b>			<b>Register Name: PB_CFG1</b>	
			<b>Default Value: 0x7777_7777</b>	
Bit	Read/Write	Default	Description	
31	/	/	/	
			PB15_SELECT	
			000:Input	001:Output
			010:/	011:MCSI_SCK
			100:TWI4_SCK	101:TRACE_DOUT15
30:28	R/W	0x7	110:PB_EINT15	111:IO Disable
27	/	/		
			PB14_SELECT	
			000:Input	001:Output
			010:/	011:MCSI_MCLK
26:24	R/W	0x7		

			100:Reserved	101:TRACE_DOUT14
			110:PB_EINT14	111:IO Disable
23:0	/	/	/	

### 3.24.3.12. PB Configuration Register 2

<b>Offset: 0x2C</b>			<b>Register Name: PB_CFG2</b>	
			<b>Default Value: 0x0000_7777</b>	
Bit	Read/Write	Default	Description	
31:3	/	/	/	
			PB16_SELECT	
			000:Input	001:Output
			010:/	011:MCSI_SDA
			100:TWI4_SDA	101:TRACE_CLK
2:0	R/W	0x7	110:PB_EINT16	111:IO Disable

### 3.24.3.13. PB Configuration Register 3

<b>Offset: 0x30</b>			<b>Register Name: PB_CFG3</b>	
			<b>Default Value: 0x0000_0000</b>	
Bit	Read/Write	Default	Description	
31:0	/	/	/	

### 3.24.3.14. PB Data Register

<b>Offset: 0x34</b>			<b>Register Name: PB_DAT</b>	
			<b>Default Value: 0x0000_0000</b>	
Bit	Read/Write	Default	Description	
31:20	/	/	/	
			PB_DAT	
19:0	R/W	0	If the port is configured as input, the corresponding bit is the pin state.	

			If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.
--	--	--	--

### 3.24.3.15. PB Multi-Driving Register 0

<b>Offset: 0x38</b>			<b>Register Name: PB_DRV0</b> <b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i]	R/W	0x1	PB_DRV PB[n] Multi-Driving Select (n = 0~15) 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
(i=0~15)			

### 3.24.3.16. PB Multi-Driving Register 1

<b>Offset: 0x3C</b>			<b>Register Name: PB_DRV1</b> <b>Default Value: 0x0000_0055</b>
Bit	Read/Write	Default	Description
31:8	/	/	Reserved
[2i+1:2i]	R/W	0x1	PB_DRV PB[n] Multi-Driving Select (n = 0~3) 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
(i=0~3)			

### 3.24.3.17. PB Pull Register 0

<b>Offset: 0x40</b>			<b>Register Name: PB_PULL0</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description

[2i+1:2i]			PB_PULL
(i=0~15)	R/W	0x0	PB[n] Pull-up/down Select (n = 0~15) 00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

### 3.24.3.18. PB Pull Register 1

<b>Offset: 0x44</b>			<b>Register Name: PB_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31;8	/	/	Reserved
[2i+1:2i]			PB_PULL
(i=0~3)	R/W	0x0	PB[n] Pull-up/down Select (n = 0~3) 00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

### 3.24.3.19. PC Configuration Register 0

<b>Offset: 0x48</b>			<b>Register Name: PC_CFG0</b>
			<b>Default Value: 0x7777_7777</b>
Bit	Read/Write	Default	Description
31	/	/	/
			PC7_SELECT
			000:Input                            001:Output 010:NAND0_RB1                    011:SDC2_CLK 100:Reserved                        101:Reserved
30:28	R/W	0x00	110:Reserved                        111:IO Disable
27	/	/	
26:24	R/W	0x00	PC6_SELECT

			000:Input	001:Output
			010:NAND0_RB0	011:SDC2_CMD
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
23	/	/		
			PC5_SELECT	
			000:Input	001:Output
			010:NAND0_RE	011:Reserved
			100:Reserved	101:Reserved
22:20	R/W	0x00	110:Reserved	111:IO Disable
19	/	/		
			PC4_SELECT	
			000:Input	001:Output
			010:NAND0_CEO	011:Reserved
			100:Reserved	101:Reserved
18:16	R/W	0x00	110:Reserved	111:IO Disable
15	/	/		
			PC3_SELECT	
			000:Input	001:Output
			010:NAND0_CE1	011:Reserved
			100:Reserved	101:Reserved
14:12	R/W	0x00	110:Reserved	111:IO Disable
11	/	/		
			PC2_SELECT	
			000:Input	001:Output
			010:NAND0_CLE	011:SPIO_CLK
10:8	R/W	0x00		



			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
7	/	/		
			PC1_SELECT	
			000:Input	001:Output
			010:NAND0_ALE	011:SPIO_MISO
			100:Reserved	101:Reserved
6:4	R/W	0x00	110:Reserved	111:IO Disable
3	/	/		
			PC0_SELECT	
			000:Input	001:Output
			010:NAND0_WE	011:SPIO_MOSI
			100:Reserved	101:Reserved
2:0	R/W	0x00	110:Reserved	111:IO Disable

### 3.24.3.20. PC Configuration Register 1

<b>Offset: 0x4C</b>			<b>Register Name: PC_CFG1</b>
			<b>Default Value: 0x7777_7777</b>
Bit	Read/Write	Default	Description
31	/	/	/
			PC15_SELECT
			000:Input
			001:Output
			010:NAND0_DQ7
			011:SDC2_D7
			100:Reserved
			101:Reserved
30:28	R/W	0x00	110:Reserved
			111:IO Disable
27	/	/	
			PC14_SELECT
26:24	R/W	0x00	

			000:Input	001:Output
			010:NAND0_DQ6	011:SDC2_D6
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
23	/	/		
			PC13_SELECT	
			000:Input	001:Output
			010:NAND0_DQ5	011:SDC2_D5
			100:Reserved	101:Reserved
22:20	R/W	0x00	110:Reserved	111:IO Disable
19	/	/		
			PC12_SELECT	
			000:Input	001:Output
			010:NAND0_DQ4	011:SDC2_D4
			100:Reserved	101:Reserved
18:16	R/W	0x00	110:Reserved	111:IO Disable
15	/	/		
			PC11_SELECT	
			000:Input	001:Output
			010:NAND0_DQ3	011:SDC2_D3
			100:Reserved	101:Reserved
14:12	R/W	0x00	110:Reserved	111:IO Disable
11	/	/		
			PC10_SELECT	
			000:Input	001:Output
			010:NAND0_DQ2	011:SDC2_D2
10:8	R/W	0x00		

			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
7	/	/		
			PC9_SELECT	
			000:Input	001:Output
			010:NAND0_DQ1	011:SDC2_D1
			100:Reserved	101:Reserved
6:4	R/W	0x00	110:Reserved	111:IO Disable
3	/	/		
			PC8_SELECT	
			000:Input	001:Output
			010:NAND0_DQ0	011:SDC2_D0
			100:Reserved	101:Reserved
2:0	R/W	0x00	110:Reserved	111:IO Disable

### 3.24.3.21. PC Configuration Register 2

<b>Offset: 0x50</b>			<b>Register Name: PC_CFG2</b>
			<b>Default Value: 0x0000_7777</b>
Bit	Read/Write	Default	Description
31:15	/	/	/
			PC19_SELECT
			000:Input
			001:Output
			010:Reserved
			011:SPIO_CS0
			100:Reserved
			101:Reserved
14:12	R/W	0x00	110:Reserved
			111:IO Disable
11	/	/	/
			PC18_SELECT
10:8	R/W	0x00	

			000:Input	001:Output
			010:NAND0_CE3	011:NAND0_DQS_B
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
7	/	/		
			PC17_SELECT	
			000:Input	001:Output
			010:NAND0_CE2	011:NAND0_RE_B
			100:Reserved	101:Reserved
6:4	R/W	0x00	110:Reserved	111:IO Disable
3	/	/		
			PC16_SELECT	
			000:Input	001:Output
			010:NAND0_DQS	011:SDC2_RST
			100:Reserved	101:Reserved
2:0	R/W	0x00	110:Reserved	111:IO Disable

### 3.24.3.22. PC Configuration Register 3

<b>Offset: 0x54</b>			<b>Register Name: PC_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.23. PC Data Register

<b>Offset: 0x58</b>			<b>Register Name: PC_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description

31:20	/	/	/
19:0	R/W	0	PC_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.24.3.24. PC Multi-Driving Register 0

<b>Offset: 0x5C</b>			<b>Register Name: PC_DRV0</b>  <b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i]	R/W	/	PC_DRV  PC[n] Multi-Driving_SELECT (n = 0~15)  00: Level 0                      01: Level 1  10: Level 2                      11: Level 3
(i=0~15)			

### 3.24.3.25. PC Multi-Driving Register 1

<b>Offset: 0x60</b>			<b>Register Name: PC_DRV1</b>  <b>Default Value: 0x0000_0055</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
[2i+1:2i]	R/W	/	PC_DRV  PC[n] Multi-Driving Select (n = 16~19)  00: Level 0                      01: Level 1  10: Level 2                      11: Level 3
(i=0~2)			

**3.24.3.26. PC Pull Register 0**

<b>Offset: 0x64</b>			<b>Register Name: PC_PULL0</b>
			<b>Default Value: 0x0000_5140</b>
Bit	Read/Write	Default	Description
[2i+1:2i]			PC_PULL PC[n] Pull-up/down Select (n = 0~15)
(i=0~15)	R/W	/	00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

**3.24.3.27. PC Pull Register 1**

<b>Offset: 0x68</b>			<b>Register Name: PC_PULL1</b>
			<b>Default Value: 0x0000_0054</b>
Bit	Read/Write	Default	Description
31:6	/	/	/
[2i+1:2i]			PC_PULL PC[n] Pull-up/down Select (n = 16~19)
(i=0~2)	R/W	/	00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

**3.24.3.28. PD Configuration Register 0**

<b>Offset: 0x6C</b>			<b>Register Name: PD_CFG0</b>
			<b>Default Value: 0x7777_7777</b>
Bit	Read/Write	Default	Description
31	/	/	/
30:28	R/W	0x7	PD7_SELECT 000:Input                            001:Output

			010:LCD_D7	011:LVDS0_VNC
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
27	/	/		
			PD6_SELECT	
			000:Input	001:Output
			010:LCD_D6	011:LVDS0_VPC
			100:Reserved	101:Reserved
26:24	R/W	0x7	110:Reserved	111:IO Disable
23	/	/		
			PD5_SELECT	
			000:Input	001:Output
			010:LCD_D5	011:LVDS0_VN2
			100:Reserved	101:Reserved
22:20	R/W	0x7	110:Reserved	111:IO Disable
19	/	/		
			PD4_SELECT	
			000:Input	001:Output
			010:LCD_D4	011:LVDS0_VP2
			100:Reserved	101:Reserved
18:16	R/W	0x7	110:Reserved	111:IO Disable
15	/	/		
			PD3_SELECT	
			000:Input	001:Output
			010:LCD_D3	011:LVDS0_VN1
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:Reserved	111:IO Disable

11	/	/	
			PD2_SELECT 000:Input                      001:Output 010:LCD_D2                      011:LVDS0_VP1 100:Reserved                      101:Reserved 110:Reserved                      111:IO Disable
10:8	R/W	0x7	
7	/	/	
			PD1_SELECT 000:Input                      001:Output 010:LCD_D1                      011:LVDS0_VN0 100:Reserved                      101:Reserved 110:Reserved                      111:IO Disable
6:4	R/W	0x7	
3	/	/	
			PD0_SELECT 000:Input                      001:Output 010:LCD_D0                      011:LVDS0_VP0 100:Reserved                      101:Reserved 110:Reserved                      111:IO Disable
2:0	R/W	0x7	

### 3.24.3.29. PD Configuration Register 1

<b>Offset: 0x70</b>			<b>Register Name: PD_CFG1</b>
			<b>Default Value: 0x7777_7777</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31	/	/	/
			PD15_SELECT 000:Input                      001:Output 010:LCD_D15                      011:LVDS1_VN2
30:28	R/W	0x7	



			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
27	/	/		
			PD14_SELECT	
			000:Input	001:Output
			010:LCD_D14	011:LVDS1_VP2
			100:Reserved	101:Reserved
26:24	R/W	0x7	110:Reserved	111:IO Disable
23	/	/		
			PD13_SELECT	
			000:Input	001:Output
			010:LCD_D13	011:LVDS1_VN1
			100:Reserved	101:Reserved
22:20	R/W	0x7	110:Reserved	111:IO Disable
19	/	/		
			PD12_SELECT	
			000:Input	001:Output
			010:LCD_D12	011:LVDS1_VP1
			100:Reserved	101:Reserved
18:16	R/W	0x7	110:Reserved	111:IO Disable
15	/	/		
			PD11_SELECT	
			000:Input	001:Output
			010:LCD_D11	011:LVDS1_VN0
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:Reserved	111:IO Disable
11	/	/		

			PD10_SELECT	
			000:Input	001:Output
			010:LCD_D10	011:LVDS1_VP0
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:Reserved	111:IO Disable
7	/	/		
			PD9_SELECT	
			000:Input	001:Output
			010:LCD_D9	011:LVDS0_VN3
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:Reserved	111:IO Disable
3	/	/		
			PD8_SELECT	
			000:Input	001:Output
			010:LCD_D8	011:LVDS0_VP3
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:Reserved	111:IO Disable

### 3.24.3.30. PD Configuration Register 2

<b>Offset: 0x74</b>			<b>Register Name: PD_CFG2</b>	
			<b>Default Value: 0x7777_7777</b>	
Bit	Read/Write	Default	Description	
31	/	/	/	
			PD23_SELECT	
			000:Input	001:Output
			010:LCD_D23	011:Reserved
30:28	R/W	0x7		

			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
27	/	/		
			PD22_SELECT	
			000:Input	001:Output
			010:LCD_D22	011:Reserved
			100:Reserved	101:Reserved
26:24	R/W	0x7	110:Reserved	111:IO Disable
23	/	/		
			PD21_SELECT	
			000:Input	001:Output
			010:LCD_D21	011:Reserved
			100:Reserved	101:Reserved
22:20	R/W	0x7	110:Reserved	111:IO Disable
19	/	/		
			PD20_SELECT	
			000:Input	001:Output
			010:LCD_D20	011:Reserved
			100:Reserved	101:Reserved
18:16	R/W	0x7	110:Reserved	111:IO Disable
15	/	/		
			PD19_SELECT	
			000:Input	001:Output
			010:LCD_D19	011:LVDS1_VN3
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:Reserved	111:IO Disable
11	/	/		

			PD18_SELECT
			000:Input                      001:Output
			010:LCD_D18                      011:LVDS1_VP3
			100:Reserved                      101:Reserved
10:8	R/W	0x7	110:Reserved                      111:IO Disable
7	/	/	
			PD17_SELECT
			000:Input                      001:Output
			010:LCD_D17                      011:LVDS1_VNC
			100:Reserved                      101:Reserved
6:4	R/W	0x7	110:Reserved                      111:IO Disable
3	/	/	
			PD16_SELECT
			000:Input                      001:Output
			010:LCD_D16                      011:LVDS1_VPC
			100:Reserved                      101:Reserved
2:0	R/W	0x7	110:Reserved                      111:IO Disable

### 3.24.3.31. PD Configuration Register 3

<b>Offset: 0x78</b>			<b>Register Name: PD_CFG3</b>
			<b>Default Value: 0x0000_7777</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
15	/	/	/
			PD27_SELECT
			000:Input                      001:Output
			010:LCD_VSYNC                      011:Reserved
14:12	R/W	0x7	

			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
11	/	/		
			PD26_SELECT	
			000:Input	001:Output
			010:LCD_HSYNC	011:Reserved
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:Reserved	111:IO Disable
7	/	/		
			PD25_SELECT	
			000:Input	001:Output
			010:LCD_DE	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:Reserved	111:IO Disable
3	/	/		
			PD24_SELECT	
			000:Input	001:Output
			010:LCD_CLK	011:Reserved
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:Reserved	111:IO Disable

### 3.24.3.32. PD Data Register

Offset: 0x7C			Register Name: PD_DAT
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:28	/	/	/
27:0	R/W	0	PD_DAT If the port is configured as input, the corresponding bit is the pin state.

			If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.
--	--	--	--

### 3.24.3.33. PD Multi-Driving Register 0

<b>Offset: 0x80</b>			<b>Register Name: PD_DRV0</b> <b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i]	R/W	0x1	PD_DRV PD[n] Multi-Driving Select (n = 0~15) 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
(i=0~15)			

### 3.24.3.34. PD Multi-Driving Register 1

<b>Offset: 0x84</b>			<b>Register Name: PD_DRV1</b> <b>Default Value: 0x0055_5555</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
[2i+1:2i]	R/W	0x1	PD_DRV PD[n] Multi-Driving Select (n = 16~27) 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
(i=0~11)			

### 3.24.3.35. PD Pull Register 0

<b>Offset: 0x88</b>			<b>Register Name: PD_PULL0</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description

			PD_PULL
			PD[n] Pull-up/down Select (n = 0~15)
[2i+1:2i]			00: Pull-up/down disable    01: Pull-up
(i=0~15)	R/W	0x0	10: Pull-down                    11: Reserved

### 3.24.3.36. PD Pull Register 1

<b>Offset: 0x8C</b>			<b>Register Name: PD_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
			PD_PULL
			PD[n] Pull-up/down Select (n = 16~27)
[2i+1:2i]			00: Pull-up/down disable    01: Pull-up enable
(i=0~11)	R/W	0x0	10: Pull-down                    11: Reserved

### 3.24.3.37. PE Configuration Register 0

<b>Offset: 0x90</b>			<b>Register Name: PE_CFG0</b>
			<b>Default Value: 0x7777_7777</b>
Bit	Read/Write	Default	Description
31	/	/	/
			PE7_SELECT
			000:Input                            001:Output
			010:CSI_D3                            011:SPI2_MISO
			100:UART5_CTS                    101:Reserved
30:28	R/W	0x7	110:PE_EINT7                    111:IO Disable
27	/	/	
			PE6_SELECT
26:24	R/W	0x7	

			000:Input 010:CSI_D2 100:UART5_RTS 110:PE_EINT6	001:Output 011:SPI2_MOSI 101:Reserved 111:IO Disable
23	/	/		
			PE5_SELECT 000:Input 010:CSI_D1 100:UART5_RX 110:PE_EINT5	001:Output 011:SPI2_CLK 101:Reserved 111:IO Disable
22:20	R/W	0x7		
19	/	/		
			PE4_SELECT 000:Input 010:CSI_D0 100:UART5_TX 110:PE_EINT4	001:Output 011:SPI2_CS0 101:Reserved 111:IO Disable
18:16	R/W	0x7		
15	/	/		
			PE3_SELECT 000:Input 010:CSI_VSYNC 100:Reserved 110:PE_EINT3	001:Output 011:TS_DVLD 101:Reserved 111:IO Disable
14:12	R/W	0x7		
11	/	/		
			PE2_SELECT 000:Input 010:CSI_HSYNC	001:Output 011:TS_SYNC
10:8	R/W	0x7		



			100:Reserved	101:Reserved
			110:PE_EINT2	111:IO Disable
7	/	/		
			PE1_SELECT	
			000:Input	001:Output
			010:CSI_MCLK	011:TS_ERR
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:PE_EINT1	111:IO Disable
3	/	/		
			PE0_SELECT	
			000:Input	001:Output
			010:CSI_PCLK	011:TS_CLK
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:PE_EINT0	111:IO Disable

### 3.24.3.38. PE Configuration Register 1

<b>Offset: 0x94</b>			<b>Register Name: PE_CFG1</b>	
			<b>Default Value: 0x7777_7777</b>	
Bit	Read/Write	Default	Description	
31	/	/	/	
			PE15_SELECT	
			000:Input	001:Output
			010:CSI_D11	011:TS_D7
			100:Reserved	101:Reserved
30:28	R/W	0x7	110:PE_EINT15	111:IO Disable
27	/	/		
			PE14_SELECT	
26:24	R/W	0x7		

			000:Input 010:CSI_D10 100:Reserved 110:PE_EINT14	001:Output 011:TS_D6 101:Reserved 111:IO Disable
23	/	/		
			PE13_SELECT 000:Input 010:CSI_D9 100:Reserved 110:PE_EINT13	001:Output 011:TS_D5 101:Reserved 111:IO Disable
22:20	R/W	0x7		
19	/	/		
			PE12_SELECT 000:Input 010:CSI_D8 100:Reserved 110:PE_EINT12	001:Output 011:TS_D4 101:Reserved 111:IO Disable
18:16	R/W	0x7		
15	/	/		
			PE11_SELECT 000:Input 010:CSI_D7 100:Reserved 110:PE_EINT11	001:Output 011:TS_D3 101:Reserved 111:IO Disable
14:12	R/W	0x7		
11	/	/		
			PE10_SELECT 000:Input 010:CSI_D6	001:Output 011:TS_D2
10:8	R/W	0x7		

			100:Reserved	101:Reserved
			110:PE_EINT10	111:IO Disable
7	/	/		
			PE9_SELECT	
			000:Input	001:Output
			010:CSI_D5	011:TS_D1
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:PE_EINT9	111:IO Disable
3	/	/		
			PE8_SELECT	
			000:Input	001:Output
			010:CSI_D4	011:TS_D0
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:PE_EINT8	111:IO Disable

### 3.24.3.39. PE Configuration Register 2

<b>Offset: 0x98</b>			<b>Register Name: PE_CFG2</b>
			<b>Default Value: 0x0007_7777</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
			PE17_SELECT
			000:Input
			001:Output
			010:CSI_SDA
			011:TWI4_SDA
			100:Reserved
			101:Reserved
6:4	R/W	0x7	110:PE_EINT17
			111:IO Disable
3	/	/	
			PE16_SELECT
2:0	R/W	0x7	

			000:Input	001:Output
			010:CSI_SCK	011:TWI4_SCK
			100:Reserved	101:Reserved
			110:PE_EINT16	111:IO Disable

### 3.24.3.40. PE Configuration Register 3

<b>Offset: 0x9C</b>			<b>Register Name: PE_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.41. PE Data Register

<b>Offset: 0xA0</b>			<b>Register Name: PE_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/
20:0	R/W	0	PE_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.24.3.42. PE Multi-Driving Register 0

<b>Offset: 0xA4</b>			<b>Register Name: PE_DRV0</b>
			<b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i] (i=0~15)	R/W	0x1	PE_DRV PE[n] Multi-Driving Select (n = 0~15)

			00: Level 0	01: Level 1
			10: Level 2	11: Level 3

### 3.24.3.43. PE Multi-Driving Register 1

<b>Offset: 0xA8</b>			<b>Register Name: PE_DRV1</b>
			<b>Default Value: 0x0000_0155</b>
Bit	Read/Write	Default	Description
31:10	/	/	/
[2i+1:2i]			PE_DRV PE[n] Multi-Driving Select (n = 16~20)
(i=0~4)	R/W	0x1	00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

### 3.24.3.44. PE Pull Register 0

<b>Offset: 0xAC</b>			<b>Register Name: PE_PULL0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[2i+1:2i]			PE_PULL PE[n] Pull-up/down Select (n = 0~15)
(i=0~15)	R/W	0x0	00: Pull-up/down disable    01: Pull-up 10: Pull-down                      11: Reserved

### 3.24.3.45. PE Pull Register 1

<b>Offset: 0xB0</b>			<b>Register Name: PE_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:10	/	/	/

			PE_PULL
			PE[n] Pull-up/down Select (n = 16~20)
[2i+1:2i]			00: Pull-up/down disable      01: Pull-up
(i=0~4)	R/W	0x0	10: Pull-down                      11: Reserved

### 3.24.3.46. PF Configuration Register 0

Offset: 0xB4			Register Name: PF_CFG0
Default Value: 0x0077_7777			
Bit	Read/Write	Default	Description
31:23	/	/	/
			PF5_SELECT
			000:Input                              001:Output
			010:SDC0_D2                          011:Reserved
			100:JTAG_CK                          101:GPU_TCK
22:20	R/W	0x7	110:Reserved                          111:IO Disable
19	/	/	
			PF4_SELECT
			000:Input                              001:Output
			010:SDC0_D3                          011:Reserved
			100:UART0_RX                          101:GPU_TRSTN
18:16	R/W	0x7	110:Reserved                          111:IO Disable
15	/	/	
			PF3_SELECT
			000:Input                              001:Output
			010:SDC0_CMD                          011:Reserved
14:12	R/W	0x7	

			100:JTAG_DO	101:GPU_TDO
			110:Reserved	111:IO Disable
11	/	/		
			PF2_SELECT	
			000:Input	001:Output
			010:SDC0_CLK	011:Reserved
			100:UART0_TX	101:Reserved
10:8	R/W	0x7	110:Reserved	111:IO Disable
7	/	/		
			PF1_SELECT	
			000:Input	001:Output
			010:SDC0_D0	011:Reserved
			100:JTAG_DI	101:GPU_TDI
6:4	R/W	0x7	110:Reserved	111:IO Disable
3	/	/		
			PF0_SELECT	
			000:Input	001:Output
			010:SDC0_D1	011:Reserved
			100:JTAG_MS	101:GPU_TMS
2:0	R/W	0x7	110:Reserved	111:IO Disable

### 3.24.3.47. PF Configuration Register 1

Offset: 0xB8			Register Name: PF_CFG1
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.48. PF Configuration Register 2**

<b>Offset: 0xBC</b>			<b>Register Name: PF_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.49. PF Configuration Register 3**

<b>Offset: 0xC0</b>			<b>Register Name: PF_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.50. PF Data Register**

<b>Offset: 0xC4</b>			<b>Register Name: PF_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:6	/	/	/
5:0	R/W	0	PF_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

**3.24.3.51. PF Multi-Driving Register 0**

<b>Offset: 0xC8</b>			<b>Register Name: PF_DRV0</b>
			<b>Default Value: 0x0000_0555</b>
Bit	Read/Write	Default	Description
31:12	/	/	/



			PF_DRV
			PF[n] Multi-Driving Select (n = 0~5)
[2i+1:2i]			00: Level 0                      01: Level 1
(i=0~5)	R/W	0x1	10: Level 2                      11: Level 3

### 3.24.3.52. PF Multi-Driving Register 1

<b>Offset: 0xCC</b>			<b>Register Name: PF_DRV1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.53. PF Pull Register 0

<b>Offset: 0xD0</b>			<b>Register Name: PF_PULL0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:12	/	/	/
			PF_PULL
			PF[n] Pull-up/down Select (n = 0~5)
[2i+1:2i]			00: Pull-up/down disable    01: Pull-up
(i=0~5)	R/W	0x0	10: Pull-down                      11: Reserved

### 3.24.3.54. PF Pull Register 1

<b>Offset: 0xD4</b>			<b>Register Name: PF_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.55. PG Configuration Register 0**

Offset: 0xD8			Register Name: PG_CFG0
Default Value: 0x7777_7777			
Bit	Read/Write	Default	Description
31	/	/	/
			PG7_SELECT
			000:Input                      001:Output
			010:UART2_RX                011:Reserved
			100:Reserved                101:Reserved
30:28	R/W	0x7	110:PG_EINT7                111:IO Disable
27	/	/	/
			PG6_SELECT
			000:Input                      001:Output
			010:UART2_TX                011:Reserved
			100:Reserved                101:Reserved
26:24	R/W	0x7	110:PG_EINT6                111:IO Disable
23	/	/	/
			PG5_SELECT
			000:Input                      001:Output
			010:SDC1_D3                011:Reserved
			100:Reserved                101:Reserved
22:20	R/W	0x7	110:PG_EINT5                111:IO Disable
19	/	/	/
			PG4_SELECT
			000:Input                      001:Output
			010:SDC1_D2                011:Reserved
			100:Reserved                101:Reserved
18:16	R/W	0x7	

			110:PG_EINT4	111:IO Disable
15	/	/		
			PG3_SELECT	
			000:Input	001:Output
			010:SDC1_D1	011:Reserved
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:PG_EINT3	111:IO Disable
11	/	/		
			PG2_SELECT	
			000:Input	001:Output
			010:SDC1_D0	011:Reserved
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:PG_EINT2	111:IO Disable
7	/	/		
			PG1_SELECT	
			000:Input	001:Output
			010:SDC1_CMD	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:PG_EINT1	111:IO Disable
3	/	/		
			PG0_SELECT	
			000:Input	001:Output
			010:SDC1_CLK	011:Reserved
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:PG_EINT0	111:IO Disable

### 3.24.3.56. PG Configuration Register 1

Offset: 0xDC			Register Name: PG_CFG1
Default Value: 0x7777_7777			Default Value: 0x7777_7777
Bit	Read/Write	Default	Description
31	/	/	/
			PG15_SELECT 000:Input                      001:Output 010:UART4_CTS                011:Reserved 100:Reserved                    101:Reserved
30:28	R/W	0x7	110:PG_EINT15                    111:IO Disable
27	/	/	/
			PG14_SELECT 000:Input                      001:Output 010:UART4_RTS                011:Reserved 100:Reserved                    101:Reserved
26:24	R/W	0x7	110:PG_EINT14                    111:IO Disable
23	/	/	/
			PG13_SELECT 000:Input                      001:Output 010:UART4_RX                 011:Reserved 100:Reserved                    101:Reserved
22:20	R/W	0x7	110:PG_EINT13                    111:IO Disable
19	/	/	/
			PG12_SELECT 000:Input                      001:Output 010:UART4_TX                 011:Reserved 100:Reserved                    101:Reserved
18:16	R/W	0x7	

			110:PG_EINT12	111:IO Disable
15	/	/		
			PG11_SELECT	
			000:Input	001:Output
			010:TWI3_SDA	011:Reserved
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:PG_EINT11	111:IO Disable
11	/	/		
			PG10_SELECT	
			000:Input	001:Output
			010:TWI3_SCK	011:Reserved
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:PG_EINT10	111:IO Disable
7	/	/		
			PG9_SELECT	
			000:Input	001:Output
			010:UART2_CTS	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:PG_EINT9	111:IO Disable
3	/	/		
			PG8_SELECT	
			000:Input	001:Output
			010:UART2_RTS	011:Reserved
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:PG_EINT8	111:IO Disable

**3.24.3.57. PG Configuration Register 2**

<b>Offset: 0xE0</b>			<b>Register Name: PG_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.58. PG Configuration Register 3**

<b>Offset: 0xE4</b>			<b>Register Name: PG_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.59. PG Data Register**

<b>Offset: 0xE8</b>			<b>Register Name: PG_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
15:0	R/W	0	PG_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

**3.24.3.60. PG Multi-Driving Register 0**

<b>Offset: 0xEC</b>			<b>Register Name: PG_DRV0</b>
			<b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i]	R/W	0x1	PG_DRV

(i=0~15)			PG[n] Multi-Driving Select (n = 0~15)
			00: Level 0                      01: Level 1
			10: Level 2                      11: Level 3

### 3.24.3.61. PG Multi-Driving Register 1

<b>Offset: 0xF0</b>			<b>Register Name: PG_DRV1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.62. PG Pull Register 0

<b>Offset: 0xF4</b>			<b>Register Name: PG_PULL0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[2i+1:2i]			PG_PULL
			PG[n] Pull-up/down Select (n = 0~15)
			00: Pull-up/down disable    01: Pull-up
(i=0~15)	R/W	0x0	10: Pull-down                      11: Reserved

### 3.24.3.63. PG Pull Register 1

<b>Offset: 0xF8</b>			<b>Register Name: PG_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.24.3.64. PH Configuration Register 0**

Offset: 0xFC			Register Name: PH_CFG0
Default Value: 0x7777_7777			
Bit	Read/Write	Default	Description
31:27	/	/	/
			PH6_SELECT
			000:Input                      001:Output
			010:PWM0                      011:Reserved
			100:Reserved                      101:Reserved
26:24	R/W	0x7	110:Reserved                      111:IO Disable
23	/	/	
			PH5_SELECT
			000:Input                      001:Output
			010:TWI2_SDA                      011:Reserved
			100:Reserved                      101:Reserved
22:20	R/W	0x7	110:Reserved                      111:IO Disable
19	/	/	
			PH4_SELECT
			000:Input                      001:Output
			010:TWI2_SCK                      011:Reserved
			100:Reserved                      101:Reserved
18:16	R/W	0x7	110:Reserved                      111:IO Disable
15	/	/	
			PH3_SELECT
			000:Input                      001:Output
			010:TWI1_SDA                      011:Reserved
			100:Reserved                      101:Reserved
14:12	R/W	0x7	



			110:Reserved	111:IO Disable
11	/	/		
			PH2_SELECT	
			000:Input	001:Output
			010:TWI1_SCK	011:Reserved
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:Reserved	111:IO Disable
7	/	/		
			PH1_SELECT	
			000:Input	001:Output
			010:TWI0_SDA	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:Reserved	111:IO Disable
3	/	/		
			PH0_SELECT	
			000:Input	001:Output
			010:TWI0_SCK	011:Reserved
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:Reserved	111:IO Disable

### 3.24.3.65. PH Configuration Register 1

<b>Offset: 0x100</b>			<b>Register Name: PH_CFG1</b>
			<b>Default Value: 0x7777_7777</b>
Bit	Read/Write	Default	Description
31	/	/	/
			PH15_SELECT
			000:Input
			001:Output
30:28	R/W	0x7	

			010:SPI3_MOSI	011:Reserved
			100:Reserved	101:Reserved
			110:PH_EINT15	111:IO Disable
27	/	/		
			PH14_SELECT	
			000:Input	001:Output
			010:SPI3_CLK	011:Reserved
			100:Reserved	101:Reserved
26:24	R/W	0x7	110:PH_EINT14	111:IO Disable
23	/	/		
			PH13_SELECT	
			000:Input	001:Output
			010:UART0_RX	011:SPI3_CS3
			100:Reserved	101:Reserved
22:20	R/W	0x7	110:PH_EINT13	111:IO Disable
19	/	/		
			PH12_SELECT	
			000:Input	001:Output
			010:UART0_TX	011:SPI3_CS2
			100:Reserved	101:Reserved
18:16	R/W	0x7	110:PH_EINT12	111:IO Disable
15	/	/		
			PH11_SELECT	
			000:Input	001:Output
			010:JTAG_DIO	011:PWM2_N
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:PH_EINT11	111:IO Disable

11	/	/	
			PH10_SELECT 000:Input                      001:Output 010:JTAG_DO0                      011:PWM2_P 100: Reserved                      101:Reserved 110:PH_EINT10                      111:IO Disable
10:8	R/W	0x7	
7	/	/	
			PH9_SELECT 000:Input                      001:Output 010:JTAG_CK0                      011:PWM1_N 100: Reserved                      101:Reserved 110:PH_EINT9                      111:IO Disable
6:4	R/W	0x7	
3	/	/	
			PH8_SELECT 000:Input                      001:Output 010:JTAG_MS0                      011:PWM1_P 100: Reserved                      101:Reserved 110:PH_EINT8                      111:IO Disable
2:0	R/W	0x7	

### 3.24.3.66. PH Configuration Register 2

<b>Offset: 0x104</b>			<b>Register Name: PH_CFG2</b>
			<b>Default Value: 0x0077_7777</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:24	/	/	/
23	/	/	/
			PH21_SELECT 000:Input                      001:Output
22:20	R/W	0x7	

			010:HCEC	011: Reserved
			100:Reserved	101:Reserved
			110: Reserved	111:IO Disable
19	/	/	/	
			PH20_SELECT	
			000:Input	001:Output
			010:HSDA	011: Reserved
			100:Reserved	101:Reserved
18:16	R/W	0x7	110: Reserved	111:IO Disable
15	/	/	/	
			PH19_SELECT	
			000:Input	001:Output
			010:HSCL	011: Reserved
			100:Reserved	101:Reserved
14:12	R/W	0x7	110: Reserved	111:IO Disable
11	/	/	/	
			PH18_SELECT	
			000:Input	001:Output
			010:SPI3_CS1	011:Reserved
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:PH_EINT18	111:IO Disable
7	/	/		
			PH17_SELECT	
			000:Input	001:Output
			010:SPI3_CS0	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:PH_EINT17	111:IO Disable

3	/	/	
			PH16_SELECT
			000:Input                      001:Output
			010:SPI3_MISO                  011:Reserved
			100:Reserved                    101:Reserved
2:0	R/W	0x7	110:PH_EINT16                  111:IO Disable

### 3.24.3.67. PH Configuration Register 3

<b>Offset: 0x108</b>			<b>Register Name: PH_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.68. PH Data Register

<b>Offset: 0x10C</b>			<b>Register Name: PH_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:22	/	/	/
21:0	R/W	0	PH_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.24.3.69. PH Multi-Driving Register 0

<b>Offset: 0x110</b>			<b>Register Name: PH_DRV0</b>
			<b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description

			PH_DRV
			PH[n] Multi-Driving Select (n = 0~15)
[2i+1:2i]			00: Level 0                      01: Level 1
(i=0~15)	R/W	/	10: Level 2                      11: Level 3

### 3.24.3.70. PH Multi-Driving Register 1

<b>Offset: 0x114</b>			<b>Register Name: PH_DRV1</b>
			<b>Default Value: 0x0000_0555</b>
Bit	Read/Write	Default	Description
31:12	/	/	/
			PH_DRV
			PH[n] Multi-Driving Select (n = 16~21)
[2i+1:2i]			00: Level 0                      01: Level 1
(i=0~2)	R/W	/	10: Level 2                      11: Level 3

### 3.24.3.71. PH Pull Register 0

<b>Offset: 0x118</b>			<b>Register Name: PH_PULL0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
			PH_PULL
			PH[n] Pull-up/down Select (n = 0~15)
[2i+1:2i]			00: Pull-up/down disable      01: Pull-up
(i=0~15)	R/W	0x00000000	10: Pull-down                      11: Reserved

### 3.24.3.72. PH Pull Register 1

<b>Offset: 0x11C</b>	<b>Register Name: PH_PULL1</b>
----------------------	--------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:12	/	/	/
[2i+1:2i]	R/W	0x00000000	PH_PULL PH[n] Pull-up/down Select (n = 16~21) 00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

### 3.24.3.73. PA External Interrupt Configuration Register 0

Offset: 0x200			Register Name: PA_EINT_CFG0
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
[4i+3:4i]	R/W	0	EINT_CFG External INTn Mode (n = 0~7) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative) Others: Reserved

**3.24.3.74. PA External Interrupt Configuration Register 1**

Offset: 0x204			Register Name: PA_EINT_CFG1
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
[4i+3:4i]			ENT_CFG External INTn Mode (n = 8~15) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative)
(i=0~7)	R/W	0	Others: Reserved

**3.24.3.75. PA External Interrupt Configuration Register 2**

Offset: 0x208			Register Name: PA_EINT_CFG2
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:8	/	/	/
[4i+3:4i]			EINT_CFG External INTn Mode (n = 16~17) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative)
(i=0~1)	R/W	0	



			Others: Reserved
--	--	--	------------------

### 3.24.3.76. PA External Interrupt Configuration Register 3

<b>Offset: 0x20C</b>			<b>Register Name: PA_EINT_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.77. PA External Interrupt Control Register

<b>Offset: 0x210</b>			<b>Register Name: PA_EINT_CTL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:18	/	/	/
[n] (n=0~17)	R/W	0	EINT_CTL External INTn Enable (n = 0~17) 0: Disable 1: Enable

### 3.24.3.78. PA External Interrupt Status Register

<b>Offset: 0x214</b>			<b>Register Name: PA_EINT_STATUS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:18	/	/	/
[n] (n=0~17)	R/W	0	EINT_STATUS External INTn Pending Bit (n = 0~17) 0: No IRQ pending 1: IRQ pending

			Write '1' to clear
--	--	--	--------------------

### 3.24.3.79. PA External Interrupt Debounce Register

<b>Offset: 0x218</b>			<b>Register Name: PA_EINT_DEB</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
			DEB_CLK_PRE_SCALE Debounce Clock Pre-scale n
6:4	R/W	0	The selected clock source is prescaled by 2 <sup>n</sup> .
3:1	/	/	/
			PIO_INT_CLK_SELECT PIO Interrupt Clock Select
0	R/W	0	0: LOSC 32Khz 1: HOSC 24Mhz

### 3.24.3.80. PB External Interrupt Configuration Register 0

<b>Offset: 0x220</b>			<b>Register Name: PB_EINT_CFG0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i] (i=0~7)	R/W	0	EINT_CFG External INTn Mode (n = 0~7)  0x0: Positive Edge  0x1: Negative Edge  0x2: High Level  0x3: Low Level

			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

### 3.24.3.81. PB External Interrupt Configuration Register 1

<b>Offset: 0x224</b>			<b>Register Name: PB_EINT_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i]	R/W	0	EINT_CFG External INTn Mode (n = 8~15) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative) Others: Reserved
(i=0~7)			

### 3.24.3.82. PB External Interrupt Configuration Register 2

<b>Offset: 0x228</b>			<b>Register Name: PB_EINT_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[4i+3:4i]	R/W	0	EINT_CFG External INTn Mode (n = 16~19) 0x0: Positive Edge 0x1: Negative Edge
(i=0~3)			

			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

### 3.24.3.83. PB External Interrupt Configuration Register 3

<b>Offset: 0x22C</b>			<b>Register Name: PB_EINT_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.84. PB External Interrupt Control Register

<b>Offset: 0x230</b>			<b>Register Name: PB_EINT_CTL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:20	/	/	/
[n] (n=0~19)	R/W	0	EINT_CTL  External INTn Enable (n = 0~19)  0: Disable  1: Enable

### 3.24.3.85. PB External Interrupt Status Register

<b>Offset: 0x234</b>			<b>Register Name: PB_EINT_STATUS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:20	/	/	/
[n]	R/W	0	EINT_STATUS

(n=0~19)			External INTn Pending Bit (n = 0~19)  0: No IRQ pending  1: IRQ pending  Write '1' to clear
----------	--	--	---

### 3.24.3.86. PB External Interrupt Debounce Register

<b>Offset: 0x238</b>			<b>Register Name: PB_EINT_DEB</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0	DEB_CLK_PRE_SCALE  Debounce Clock Pre-scale n  The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0	PIO_INT_CLK_SELECT  PIO Interrupt Clock Select  0: LOSC 32Khz  1: HOSC 24Mhz

### 3.24.3.87. PE External Interrupt Configuration Register 0

<b>Offset: 0x240</b>			<b>Register Name: PE_EINT_CFG0</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i] (i=0~7)	R/W	0	EINT_CFG  External INTn Mode (n = 0~7)  0x0: Positive Edge

			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

### 3.24.3.88. PE External Interrupt Configuration Register 1

<b>Offset: 0x244</b>			<b>Register Name: PE_EINT_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i]			ENT_CFG
			External INTn Mode (n = 8~15)
			0x0: Positive Edge
			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
(i=0~7)	R/W	0	Others: Reserved

### 3.24.3.89. PE External Interrupt Configuration Register 2

<b>Offset: 0x248</b>			<b>Register Name: PE_EINT_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:20	/	/	/
[4i+3:4i]			EINT_CFG
(i=0~4)	R/W	0	

		External INTn Mode (n = 16~20)
		0x0: Positive Edge
		0x1: Negative Edge
		0x2: High Level
		0x3: Low Level
		0x4: Double Edge (Positive/ Negative)
		Others: Reserved

### 3.24.3.90. PE External Interrupt Configuration Register 3

<b>Offset: 0x24C</b>			<b>Register Name: PE_EINT_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.91. PE External Interrupt Control Register

<b>Offset: 0x250</b>			<b>Register Name: PE_EINT_CTL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:21	/	/	/
[n] (n=0~20)	R/W	0	EINT_CTL  External INTn Enable (n = 0~20)  0: Disable  1: Enable

### 3.24.3.92. PE External Interrupt Status Register

<b>Offset: 0x254</b>	<b>Register Name: PE_EINT_STATUS</b>
----------------------	--------------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:21	/	/	/
[n] (n=0~20)	R/W	0	EINT_STATUS  External INTn Pending Bit (n = 0~20)  0: No IRQ pending  1: IRQ pending  Write '1' to clear

### 3.24.3.93. PE External Interrupt Debounce Register

			Register Name: PE_EINT_DEB
Offset: 0x258			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0	DEB_CLK_PRE_SCALE  Debounce Clock Pre-scale n  The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0	PIO_INT_CLK_SELECT  PIO Interrupt Clock Select  0: LOSC 32Khz  1: HOSC 24Mhz

### 3.24.3.94. PG External Interrupt Configuration Register 0

			Register Name: PG_EINT_CFG0
Offset: 0x260			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description



			EINT_CFG
			External INTn Mode (n = 0~7)
			0x0: Positive Edge
			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
[4i+3:4i]			0x4: Double Edge (Positive/ Negative)
(i=0~7)	R/W	0	Others: Reserved

### 3.24.3.95. PG External Interrupt Configuration Register 1

<b>Offset: 0x264</b>			<b>Register Name: PG_EINT_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
			EINT_CFG
			External INTn Mode (n = 8~15)
			0x0: Positive Edge
			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
[4i+3:4i]			0x4: Double Edge (Positive/ Negative)
(i=0~7)	R/W	0	Others: Reserved

### 3.24.3.96. PG External Interrupt Configuration Register 2

<b>Offset: 0x268</b>			<b>Register Name: PG_EINT_CFG2</b>
			<b>Default Value: 0x0000_0000</b>

Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.97. PG External Interrupt Configuration Register 3

<b>Offset: 0x26C</b>			<b>Register Name: PG_EINT_CFG3</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.98. PG External Interrupt Control Register

<b>Offset: 0x270</b>			<b>Register Name: PG_EINT_CTL</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n] (n=0~15)	R/W	0	EINT_CTL External INTn Enable (n = 0~15) 0: Disable 1: Enable

### 3.24.3.99. PG External Interrupt Status Register

<b>Offset: 0x274</b>			<b>Register Name: PG_EINT_STATUS</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n] (n=0~15)	R/W	0	EINT_STATUS External INTn Pending Bit (n = 0~15) 0: No IRQ pending

			1: IRQ pending  Write '1' to clear
--	--	--	--

### 3.24.3.100. PG External Interrupt Debounce Register

<b>Offset: 0x278</b>			<b>Register Name: PG_EINT_DEB</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
			DEB_CLK_PRE_SCALE  Debounce Clock Pre-scale n
6:4	R/W	0	The selected clock source is prescaled by 2 <sup>n</sup> .
3:1	/	/	/
			PIO_INT_CLK_SELECT  PIO Interrupt Clock Select
0	R/W	0	0: LOSC 32Khz  1: HOSC 24Mhz

### 3.24.3.101. PH External Interrupt Configuration Register 0

<b>Offset: 0x280</b>			<b>Register Name:PH_EINT_CFG0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i]  (i=7)	R/W	0	EINT_CFG  External INTn Mode (n = 7)  0x0: Positive Edge  0x1: Negative Edge

			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

### 3.24.3.102. PH External Interrupt Configuration Register 1

<b>Offset: 0x284</b>			<b>Register Name: PH_EINT_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i] (i=0~7)	R/W	0	ENT_CFG External INTn Mode (n = 8~15) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative) Others: Reserved

### 3.24.3.103. PH External Interrupt Configuration Register 2

<b>Offset: 0x288</b>			<b>Register Name: PH_EINT_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i] (i=0~2)	R/W	0	EINT_CFG External INTn Mode (n = 16~18) 0x0: Positive Edge

			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

### 3.24.3.104. PH External Interrupt Configuration Register 3

<b>Offset: 0x28C</b>			<b>Register Name: PH_EINT_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.24.3.105. PH External Interrupt Control Register

<b>Offset: 0x290</b>			<b>Register Name: PH_EINT_CTL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:12	/	/	/
[n] (n=7~18)	R/W	0	EINT_CTL External INTn Enable (n = 7~18) 0: Disable 1: Enable

### 3.24.3.106. PH External Interrupt Status Register

<b>Offset: 0x294</b>			<b>Register Name: PH_EINT_STATUS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:12	/	/	/

[n]			EINT_STATUS
(n=7~18)	R/W	0	External INTn Pending Bit (n = 7~18)
			0: No IRQ pending
			1: IRQ pending
			Write '1' to clear

### 3.24.3.107. PH External Interrupt Debounce Register

<b>Offset: 0x298</b>			<b>Register Name: PH_EINT_DEB</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0	DEB_CLK_PRE_SCALE Debounce Clock Pre-scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32Khz 1: HOSC 24Mhz

### 3.24.3.108. PA Group Configuration Register

<b>Offset: 0x300</b>			<b>Register Name: PA_GRP_CONFIG</b>
			<b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	IO_BIAS_CONFIG IO Pad Bias Configuration Value

		<p>0x0: for 1.8V Power Supply;</p> <p>0x6: for 2.5V Power Supply;</p> <p>0x9: for 2.8V Power Supply;</p> <p>0xA: for 3.0V Power Supply;</p> <p>0xD: for 3.3V Power Supply;</p> <p>Others: Reserved.</p> <p><b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b></p>
--	--	--

### 3.24.3.109. PB Group Configuration Register

<b>Offset: 0x304</b>			<b>Register Name: PB_GRP_CONFIG</b> <b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
			IO_BIAS_CONFIG IO Pad Bias Configuration Value 0x0: for 1.8V Power Supply; 0x6: for 2.5V Power Supply; 0x9: for 2.8V Power Supply; 0xA: for 3.0V Power Supply; 0xD: for 3.3V Power Supply; Others: Reserved. <p><b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b></p>
3:0	R/W	0xD	

**3.24.3.110. PC Group Configuration Register**

Offset: 0x308			Register Name: PC_GRP_CONFIG
Default Value: 0x0000_0006			
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0x6	IO_BIAS_CONFIG IO Pad Bias Configuration Value 0x0: for 1.8V Power Supply; 0x6: for 2.5V Power Supply; 0x9: for 2.8V Power Supply; 0xA: for 3.0V Power Supply; 0xD: for 3.3V Power Supply; Others: Reserved.  <b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>

**3.24.3.111. PD Group Configuration Register**

Offset: 0x30C			Register Name: PD_GRP_CONFIG
Default Value: 0x0000_000D			
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	IO_BIAS_CONFIG IO Pad Bias Configuration Value 0x0: for 1.8V Power Supply; 0x6: for 2.5V Power Supply; 0x9: for 2.8V Power Supply;



			0xA: for 3.0V Power Supply;  0xD: for 3.3V Power Supply;  Others: Reserved.  <b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>
--	--	--	--

### 3.24.3.112. PE Group Configuration Register

<b>Offset: 0x310</b>			<b>Register Name: PE_GRP_CONFIG</b>  <b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	IO_BIAS_CONFIG  IO Pad Bias Configuration Value  0x0: for 1.8V Power Supply;  0x6: for 2.5V Power Supply;  0x9: for 2.8V Power Supply;  0xA: for 3.0V Power Supply;  0xD: for 3.3V Power Supply;  Others: Reserved.  <b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>

### 3.24.3.113. PF Group Configuration Register

<b>Offset: 0x314</b>			<b>Register Name: PF_GRP_CONFIG</b>  <b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/

3:0	R/W	0xD	IO_BIAS_CONFIG  IO Pad Bias Configuration Value  0x0: for 1.8V Power Supply;  0x6: for 2.5V Power Supply;  0x9: for 2.8V Power Supply;  0xA: for 3.0V Power Supply;  0xD: for 3.3V Power Supply;  Others: Reserved.  <b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>
-----	-----	-----	--

#### 3.24.3.114. PG Group Configuration Register

Offset: 0x318			Register Name: PG_GRP_CONFIG  Default Value: 0x0000_000D
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	IO_BIAS_CONFIG  IO Pad Bias Configuration Value  0x0: for 1.8V Power Supply;  0x6: for 2.5V Power Supply;  0x9: for 2.8V Power Supply;  0xA: for 3.0V Power Supply;  0xD: for 3.3V Power Supply;  Others: Reserved.  <b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>

**3.24.3.115. PH Group Configuration Register**

Offset: 0x31C			Register Name: PH_GRP_CONFIG
Bit			Default Value: 0x0000_000D
Bit	Read/Write	Default	Description
31:4	/	/	/
			IO_BIAS_CONFIG  IO Pad Bias Configuration Value  0x0: for 1.8V Power Supply;  0x6: for 2.5V Power Supply;  0x9: for 2.8V Power Supply;  0xA: for 3.0V Power Supply;  0xD: for 3.3V Power Supply;  Others: Reserved.
3:0	R/W	0xD	<b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b>

## 3.25. Port Controller(CPUs-Port)

### 3.25.1. Overview

The chip has 3 ports for multi-functional input/out pins. They are shown below:

- Port L(PL) : 14 input/output port
- Port M(PM) : 16 input/output port
- Port N(PN) : 2 input/output port

For various system configurations, these ports can be easily configured by software. All these ports can be configured as GPIO if multiplexed functions not used. The external PIO interrupt sources are supported and interrupt mode can be configured by software.

### 3.25.2. Port Register List

Module Name	Base Address
R_PIO	0x08002C00

Register Name	Offset	Description
Pn_CFG0	n*0x24+0x00	Port n Configuration Register 0 (n from 0 to 2)
Pn_CFG1	n*0x24+0x04	Port n Configuration Register 1 (n from 0 to 2)
Pn_CFG2	n*0x24+0x08	Port n Configuration Register 2 (n from 0 to 2)
Pn_CFG3	n*0x24+0x0C	Port n Configuration Register 3 (n from 0 to 2)
Pn_DAT	n*0x24+0x10	Port n Data Register (n from 0 to 2)
Pn_DRV0	n*0x24+0x14	Port n Multi-Driving Register 0 (n from 0 to 2)
Pn_DRV1	n*0x24+0x18	Port n Multi-Driving Register 1 (n from 0 to 2)
Pn_PUL0	n*0x24+0x1C	Port n Pull Register 0 (n from 0 to 2)
Pn_PUL1	n*0x24+0x20	Port n Pull Register 1 (n from 0 to 2)
Pn_INT_CFG0	0x200+n*0x20+0x00	PIO Interrupt Configure Register 0
Pn_INT_CFG1	0x200+n*0x20+0x04	PIO Interrupt Configure Register 1
Pn_INT_CFG2	0x200+n*0x20+0x08	PIO Interrupt Configure Register 2
Pn_INT_CFG3	0x200+n*0x20+0x0C	PIO Interrupt Configure Register 3
Pn_INT_CTL	0x200+n*0x20+0x10	PIO Interrupt Control Register
Pn_INT_STA	0x200+n*0x20+0x14	PIO Interrupt Status Register
Pn_INT_DEB	0x200+n*0x20+0x18	PIO Interrupt Debounce Register
Pn_GRP_CONFIG	0x300+n*0x04	PIO Group Configuration Register

### 3.25.3. Port Register Description

#### 3.25.3.1. PL Configuration Register 0

Offset: 0x00			Register Name: PL_CFG0	
Default			Default Value: 0x7777_7777	
Bit	Read/Write	Default	Description	
31	/	/	/	
			PL7_SELECT	
			000:Input	001:Output
			010: Reserved	011:1WIRE
			100:Reserved	101:Reserved
30:28	R/W	0x7	110: S_PL_EINT7	111:IO Disable
27	/	/	/	
			PL6_SELECT	
			000:Input	001:Output
			010: Reserved	011:S_CIR_RX
			100:Reserved	101:Reserved
26:24	R/W	0x7	110: S_PL_EINT6	111:IO Disable
23	/	/	/	
			PL5_SELECT	
			000:Input	001:Output
			010:Reserved	011:S_JTAG_TDI
			100:Reserved	101:Reserved
22:20	R/W	0x7	110: S_PL_EINT5	111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PL4_SELECT	

			000:Input 010: Reserved 100:Reserved 110: S_PL_EINT4	001:Output 011:S_JTAG_TDO 101:Reserved 111:IO Disable
15	/	/		
			PL3_SELECT 000:Input 010: Reserved 100:Reserved 110: S_PL_EINT3	001:Output 011:S_JTAG_TCK 101:Reserved 111:IO Disable
14:12	R/W	0x7		
11	/	/		
			PL2_SELECT 000:Input 010:Reserved 100:Reserved 110: S_PL_EINT2	001:Output 011:S_JTAG_TMS 101:Reserved 111:IO Disable
10:8	R/W	0x7		
7	/	/		
			PL1_SELECT 000:Input 010:Reserved 100:Reserved 110:S_PL_EINT1	001:Output 011:S_UART_RX 101:Reserved 111:IO Disable
6:4	R/W	0x7		
3	/	/		
			PLO_SELECT 000:Input 010:Reserved	001:Output 011:S_UART_TX
2:0	R/W	0x7		

			100:Reserved	101:Reserved
			110:S_PL_EINT0	111:IO Disable

### 3.25.3.2. PL Configuration Register 1

Offset: 0x04			Register Name: PL_CFG1	
Default Value: 0x0077_7777				
Bit	Read/Write	Default	Description	
31:24	/	/	/	
23	/	/	/	
22:20	R/W	0x7	/	
19	/	/	/	
18:16	R/W	0x7	/	
15	/	/	/	
14:12	R/W	0x7	/	
11	/	/	/	
10:8	R/W	0x7	/	
7	/	/	/	
			PL9_SELECT	
			000:Input	001:Output
			010:Reserved	011:S_PS2_SDA1
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:S_PL_EINT9	111:IO Disable
3	/	/	/	
			PL8_SELECT	
			000:Input	001:Output
			010:Reserved	011:S_PS2_SCK1
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:S_PL_EINT8	111:IO Disable



### 3.25.3.3. PL Configuration Register 2

<b>Offset: 0x08</b>			<b>Register Name: PL_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.4. PL Configuration Register 3

<b>Offset: 0x0C</b>			<b>Register Name: PL_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.5. PL Data Register

<b>Offset: 0x10</b>			<b>Register Name: PL_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:14	/	/	/
13:0	R/W	0	PL_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.25.3.6. PL Multi-Driving Register 0

<b>Offset: 0x14</b>			<b>Register Name: PL_DRV0</b>
			<b>Default Value: 0x0555_5555</b>
Bit	Read/Write	Default	Description
31:28	/	/	/

			PL_DRV
			PL[n] Multi-Driving Select (n = 0~13)
[2i+1:2i]			00: Level 0                      01: Level 1
(i=0~13)	R/W	0x1	10: Level 2                      11: Level 3

### 3.25.3.7. PL Multi-Driving Register 1

<b>Offset: 0x18</b>			<b>Register Name: PL_DRV1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.8. PL Pull Register 0

<b>Offset: 0x1C</b>			<b>Register Name: PL_PULL0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:28	/	/	/
			PL_PULL
			PL[n] Pull-up/down Select (n = 0~13)
[2i+1:2i]			00: Pull-up/down disable    01: Pull-up
(i=0~13)	R/W	0x0	10: Pull-down                      11: Reserved

### 3.25.3.9. PL Pull Register 1

<b>Offset: 0x20</b>			<b>Register Name: PL_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.10. PM Configuration Register 0**

Offset: 0x24			Register Name: PM_CFG0
Default Value: 0x7777_7777			
Bit	Read/Write	Default	Description
31	/	/	/
30:28	R/W	0x7	/
27	/	/	
26:24	R/W	0x7	/
23	/	/	
22:20	R/W	0x7	/
19	/	/	
			PM4_SELECT
			000:Input                      001:Output
			010: Reserved                011:S_I2S1_LRCKR
			100:Reserved                101:Reserved
18:16	R/W	0x7	110: S_PM_EINT4            111:IO Disable
15	/	/	
			PM3_SELECT
			000:Input                      001:Output
			010:Reserved                011:Reserved
			100:Reserved                101:Reserved
14:12	R/W	0x7	110:S_PM_EINT3            111:IO Disable
11	/	/	
			PM2_SELECT
			000:Input                      001:Output
			010Reserved                011:Reserved
			100:Reserved                101:Reserved
10:8	R/W	0x7	110:S_PM_EINT2            111:IO Disable
7	/	/	

			PM1_SELECT	
			000:Input	001:Output
			010:Reserved	011:Reserved
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:S_PM_EINT1	111:IO Disable
3	/	/		
			PM0_SELECT	
			000:Input	001:Output
			010:Reserved	011:Reserved
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:S_PM_EINT0	111:IO Disable

### 3.25.3.11. PM Configuration Register 1

<b>Offset: 0x28</b>			<b>Register Name: PM_CFG1</b>	
			<b>Default Value: 0x7777_7777</b>	
Bit	Read/Write	Default	Description	
31	/	/	/	
			PM15_SELECT	
			000:Input	001:Output
			010:Reserved	011:Reserved
			100:Reserved	101:Reserved
30:28	R/W	0x7	110:S_PM_EINT15	111:IO Disable
27	/	/		
			PM14_SELECT	
			000:Input	001:Output
			010:S_I2S_DOUT	011: S_I2S_DOUT0
26:24	R/W	0x7		

			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
23	/	/		
			PM13_SELECT	
			000:Input	001:Output
			010:S_I2S_DIN	011: S_I2S1_DIN
			100:Reserved	101:Reserved
22:20	R/W	0x7	110:Reserved	111:IO Disable
19	/	/		
			PM12_SELECT	
			000:Input	001:Output
			010:S_I2S_LRCK	011: S_I2S1_LRCK
			100:Reserved	101:Reserved
18:16	R/W	0x7	110:Reserved	111:IO Disable
15	/	/		
			PM11_SELECT	
			000:Input	001:Output
			010:S_I2S_BCLK	011: S_I2S1_BCLK
			100:Reserved	101:Reserved
14:12	R/W	0x7	110:Reserved	111:IO Disable
11	/	/		
			PM10_SELECT	
			000:Input	001:Output
			010:S_I2S_MCLK	011:S_I2S1_MCLK
			100:Reserved	101:Reserved
10:8	R/W	0x7	110:Reserved	111:IO Disable
7	/	/		

			PM9_SELECT	
			000:Input	001:Output
			010:Reserved	011:S_TWI1_SDA
			100:Reserved	101:Reserved
6:4	R/W	0x7	110:S_PM_EINT9	111:IO Disable
3	/	/		
			PM8_SELECT	
			000:Input	001:Output
			010:Reserved	011:S_TWI1_SCK
			100:Reserved	101:Reserved
2:0	R/W	0x7	110:S_PM_EINT8	111:IO Disable

### 3.25.3.12. PM Configuration Register 2

<b>Offset: 0x2C</b>			<b>Register Name: PM_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.13. PM Configuration Register 3

<b>Offset: 0x30</b>			<b>Register Name: PM_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.14. PM Data Register

<b>Offset: 0x34</b>			<b>Register Name: PM_DAT</b>
			<b>Default Value: 0x0000_0000</b>

Bit	Read/Write	Default	Description
31:16	/	/	/
15:0	R/W	0	PM_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

### 3.25.3.15. PM Multi-Driving Register 0

<b>Offset: 0x38</b>			<b>Register Name: PM_DRV0</b>  <b>Default Value: 0x5555_5555</b>
Bit	Read/Write	Default	Description
[2i+1:2i]	R/W	0x1	PM_DRV  PM[n] Multi-Driving Select (n = 0~15)  00: Level 0                      01: Level 1  10: Level 2                      11: Level 3
(i=0~15)			

### 3.25.3.16. PM Multi-Driving Register 1

<b>Offset: 0x3C</b>			<b>Register Name: PM_DRV1</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.17. PM Pull Register 0

<b>Offset: 0x40</b>			<b>Register Name: PM_PULL0</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
	R/W	0x0	PM_PULL

[2i+1:2i]			PM[n] Pull-up/down Select (n = 0~15)
(i=0~15)			00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

### 3.25.3.18. PM Pull Register 1

<b>Offset: 0x44</b>			<b>Register Name: PM_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.19. PN Configuration Register 0

<b>Offset: 0x48</b>			<b>Register Name: PN_CFG0</b>
			<b>Default Value: 0x0000_0077</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
			PN1_SELECT 000:Input                            001:Output 010:S_TWI0_SDA                    011:S_RSB_SDA 100:Reserved                        101:Reserved
6:4	R/W	0x7	110:Reserved                        111:IO Disable
3	/	/	
			PN0_SELECT 000:Input                            001:Output 010:S_TWI0_SCK                    011:S_RSB_SCK 100:Reserved                        101:Reserved
2:0	R/W	0x7	110:Reserved                        111:IO Disable



**3.25.3.20. PN Configuration Register 1**

<b>Offset: 0x4C</b>			<b>Register Name: PN_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.21. PN Configuration Register 2**

<b>Offset: 0x50</b>			<b>Register Name: PN_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.22. PN Configuration Register 3**

<b>Offset: 0x54</b>			<b>Register Name: PN_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.23. PN Data Register**

<b>Offset: 0x58</b>			<b>Register Name: PN_DAT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:2	/	/	/
1:0	R/W	0	PM_DAT  If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

**3.25.3.24. PN Multi-Driving Register 0**

<b>Offset: 0x5C</b>			<b>Register Name: PN_DRV0</b>
			<b>Default Value: 0x0000_0005</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
[2i+1:2i]			PM_DRV PM[n] Multi-Driving Select (n = 0~1)
(i=0~1)	R/W	0x1	00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

**3.25.3.25. PN Multi-Driving Register 1**

<b>Offset: 0x60</b>			<b>Register Name: PN_DRV1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.26. PN Pull Register 0**

<b>Offset: 0x64</b>			<b>Register Name: PN_PULL0</b>
			<b>Default Value: 0x0000_0005</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
[2i+1:2i]			PM_PULL PM[n] Pull-up/down Select (n = 0~1)
(i=0~1)	R/W	0x1	00: Pull-up/down disable    01: Pull-up 10: Pull-down                    11: Reserved

**3.25.3.27. PN Pull Register 1**

<b>Offset: 0x68</b>			<b>Register Name: PN_PULL1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.28. PL External Interrupt Configure Register 0**

<b>Offset: 0x200</b>			<b>Register Name: PL_EINT_CFG0</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
[4i+3:4i] (i=0~7)	R/W	0	EINT_CFG External INTn Mode (n = 0~7) 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative) Others: Reserved

**3.25.3.29. PL External Interrupt Configure Register 1**

<b>Offset: 0x204</b>			<b>Register Name: PL_EINT_CFG1</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
[4i+3:4i] (i=0~5)	R/W	0	EINT_CFG External INTn Mode (n = 8~13)

			0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/ Negative) Others: Reserved
--	--	--	--

### 3.25.3.30. PL External Interrupt Configure Register 2

<b>Offset: 0x208</b>			<b>Register Name: PL_EINT_CFG2</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.31. PL External Interrupt Configure Register 3

<b>Offset: 0x20C</b>			<b>Register Name: PL_EINT_CFG3</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

### 3.25.3.32. PL External Interrupt Control Register

<b>Offset: 0x210</b>			<b>Register Name: PL_EINT_CTL</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n] (n=0~15)	R/W	0	EINT_CTL External INTn Enable (n = 0~15)

			0: Disable
			1: Enable

### 3.25.3.33. PL External Interrupt Status Register

<b>Offset: 0x214</b>			<b>Register Name: PL_EINT_STATUS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n]			EINT_STATUS External INTn Pending Bit (n = 0~15) 0: No IRQ pending 1: IRQ pending
(n=0~15)	R/W	0	Write '1' to clear

### 3.25.3.34. PL External Interrupt Debounce Register

<b>Offset: 0x218</b>			<b>Register Name: PL_EINT_DEB</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0	DEB_CLK_PRE_SCALE Debounce Clock Pre-scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32Khz 1: HOSC 24Mhz

**3.25.3.35. PM External Interrupt Configure Register 0**

Offset: 0x220			Register Name: PM_EINT_CFG0
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
[4i+3:4i]	R/W	0	EINT_CFG
(i=0~7)			External INTn Mode (n = 0~7)
			0x0: Positive Edge
			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

**3.25.3.36. PM External Interrupt Configure Register 1**

Offset: 0x224			Register Name: PM_EINT_CFG1
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
[4i+3:4i]	R/W	0	EINT_CFG
(i=0~1,7)			External INTn Mode (n = 8~9,15)
			0x0: Positive Edge
			0x1: Negative Edge
			0x2: High Level
			0x3: Low Level
			0x4: Double Edge (Positive/ Negative)
			Others: Reserved

**3.25.3.37. PM External Interrupt Configure Register 2**

<b>Offset: 0x228</b>			<b>Register Name: PM_EINT_CFG2</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.38. PM External Interrupt Configure Register 3**

<b>Offset: 0x22C</b>			<b>Register Name: PM_EINT_CFG3</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

**3.25.3.39. PM External Interrupt Control Register**

<b>Offset: 0x230</b>			<b>Register Name: PM_EINT_CTL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n] (n=0~9,15)	R/W	0	EINT_CTL External INTn Enable (n = 0~9,15) 0: Disable 1: Enable

**3.25.3.40. PM External Interrupt Status Register**

<b>Offset: 0x234</b>			<b>Register Name: PM_EINT_STATUS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
[n]	R/W	0	EINT_STATUS

(n=0~9,15)			External INTn Pending Bit (n = 0~9,15)
			0: No IRQ pending
			1: IRQ pending
			Write '1' to clear

### 3.25.3.41. PM External Interrupt Debounce Register

<b>Offset: 0x238</b>			<b>Register Name: PM_EINT_DEB</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0	DEB_CLK_PRE_SCALE Debounce Clock Pre-scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32Khz 1: HOSC 24Mhz

### 3.25.3.42. PL Group Configuration Register

<b>Offset: 0x300</b>			<b>Register Name: PL_GRP_CONFIG</b>
			<b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	IO_BIAS_CONFIG IO Pad Bias Configuration Value 0x0: for 1.8V Power Supply;



		<p>0x6: for 2.5V Power Supply;</p> <p>0x9: for 2.8V Power Supply;</p> <p>0xA: for 3.0V Power Supply;</p> <p>0xD: for 3.3V Power Supply;</p> <p>Others: Reserved.</p> <p><b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b></p>
--	--	---

### 3.25.3.43. PM Group Configuration Register

<b>Offset: 0x304</b>			<b>Register Name: PM_GRP_CONFIG</b>  <b>Default Value: 0x0000_000D</b>
Bit	Read/Write	Default	Description
31:4	/	/	/
3:0	R/W	0xD	<p>IO_BIAS_CONFIG</p> <p>IO Pad Bias Configuration Value</p> <p>0x0: for 1.8V Power Supply;</p> <p>0x6: for 2.5V Power Supply;</p> <p>0x9: for 2.8V Power Supply;</p> <p>0xA: for 3.0V Power Supply;</p> <p>0xD: for 3.3V Power Supply;</p> <p>Others: Reserved.</p> <p><b>Note: The configuration value must greater than or equal to the corresponding value of the voltage supplied to the IO Group.</b></p>

## Chapter 4 Memory

This section describes the A80 memory from two aspects:

- SDRAM
- NAND Flash

## 4.1. SDRAM

### 4.1.1. Overview

The SDRAM Controller (DRAMC) provides a simple, flexible, burst-optimized interface to all in-dusty-standard Low Power DDR2/3, DDR3/DDR3L SDRAM. It supports up to a 64G bits memory address space.

The DRAMC automatically handles memory management, initialization, and refresh operations. It gives the host CPU a simple command interface, hiding details of the required address, page, and burst handling procedures. All memory parameters are runtime-configurable, including timing, memory setting, SDRAM type, and Extended-Mode-Register settings. To simplify chip system integration, DDR controller works in half rate mode.

The DRAMC includes the following features:

- Dual Channels SDRAM Controller with its own physical layer
- System can be configured 16/32-bits one channel or 16/32-bits two channel
- Support LPDDR2/3, DDR3/DDR3L SDRAM
- Support different memory device's power voltage of 1.2V, 1.35V, 1.5V
- Support memory capacity up to 64G bits (8G Bytes)
- Support 2 chip select signals per channel
- 16 address lines and three bank address lines per channel
- Data bus size can up to 32-bit per channel
- Automatically generates initialization and refresh sequences
- Runtime-configurable parameters setting for application flexibility
- Clock frequency can be chosen for different application
- Priority of transferring through multiple ports is programmable
- Random read or write operation is supported

## 4.2. NAND Flash

### 4.2.1. Overview

The NDFC is the NAND Flash Controller which supports all NAND flash memory available in the market. New type flash can be supported by software re-configuration.

The On-the-fly error correction code (ECC) is built-in NDFC for enhancing reliability. BCH is implemented and it can detect and correct up to 72 bits error per 512 or 1024 bytes data. The on chip ECC and parity checking circuitry of NDFC frees CPU for other tasks. The ECC function can be disabled by software.

The data can be transferred by DMA or by CPU memory-mapped IO method. The NDFC provides automatic timing control for reading or writing external Flash. The NDFC maintains the proper relativity for CLE, CE# and ALE control signal lines. Three kind of modes are supported for serial read access. The conventional serial access is mode 0 and mode 1 is for EDO type and mode 2 for extension EDO type. NDFC can monitor the status of R/B# signal line.

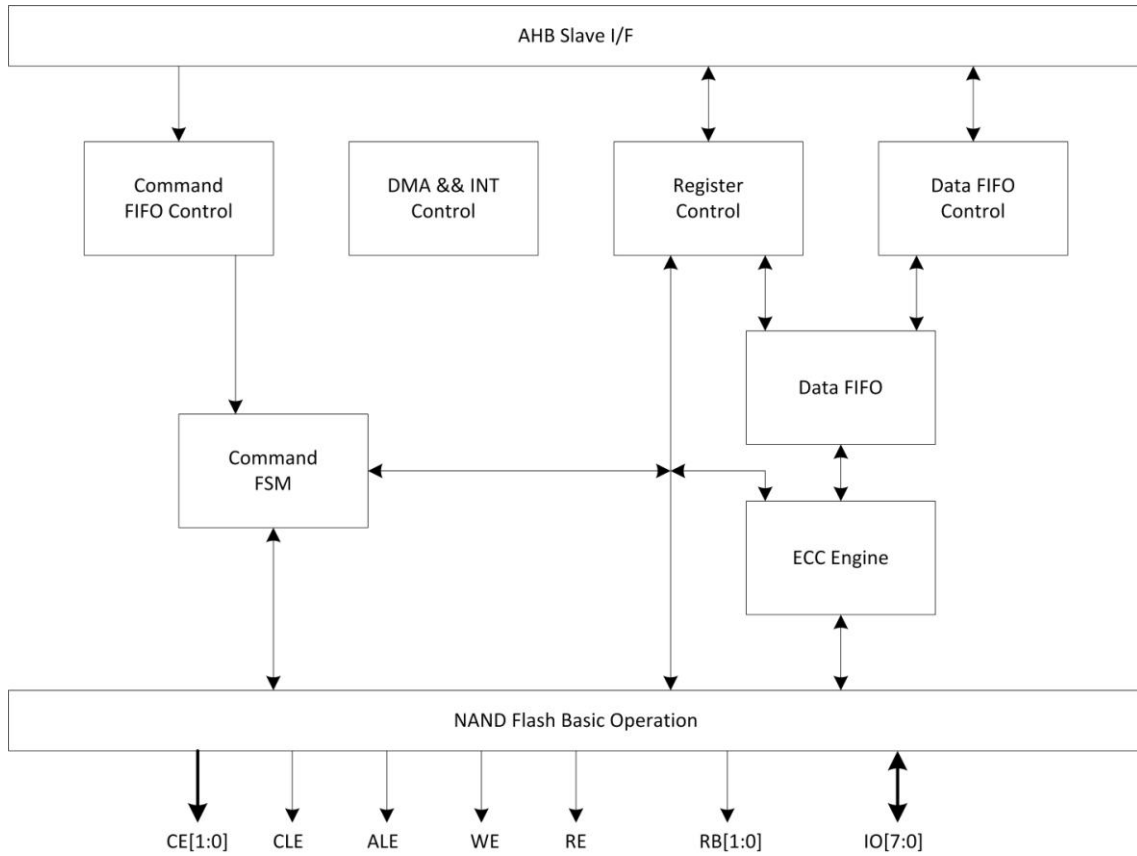
Block management and wear leveling management are implemented in software.

The NAND Flash Controller (NDFC) includes the following features:

- Supports all SLC/MLC/TLC flash and EF-NAND memory available in the market
- Software configure seed for randomize engine
- Software configure method for adaptability to a variety of system and memory types
- Supports 8-bit data bus width
- Supports 1024, 2048, 4096, 8192, 16384, 32768 bytes size per page
- Supports 1.8/3.3 V voltage supply Flash
- Supports Conventional and EDO serial access method for serial reading Flash
- On-the-fly BCH error correction code which correcting up to 72 bits per 512 or 1024 bytes
- Corrected Error bits number information report
- ECC automatic disable function for all 0xff data
- NDFC status information is reported by its' registers and interrupt is supported
- One Command FIFO
- External DMA is supported for transferring data
- Support internal DMA controller based on chain-structured descriptor list
- Two 256x32-bit RAM for pipeline procession
- Support SDR, ONFI 2.0/Toggle1.0 DDR and ONFI 3.0/Toggle 2.0 DDR2 NAND Flash
- Support self –debug for NDFC debug

### 4.2.2. Block Diagram

The NAND Flash Controller (NDFC) system block diagram is shown below:

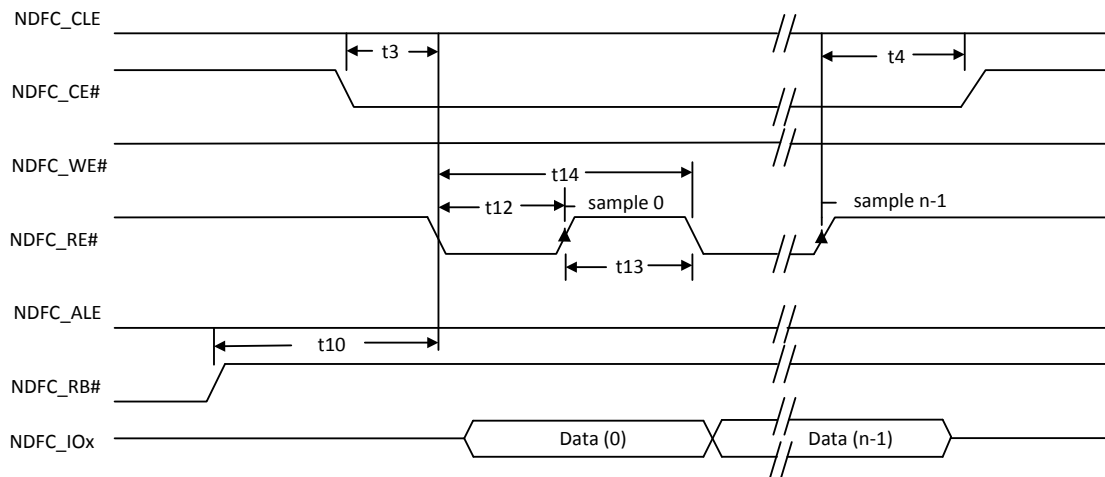


NDFC Block Diagram

### 4.2.3. NDFC Timing Diagram

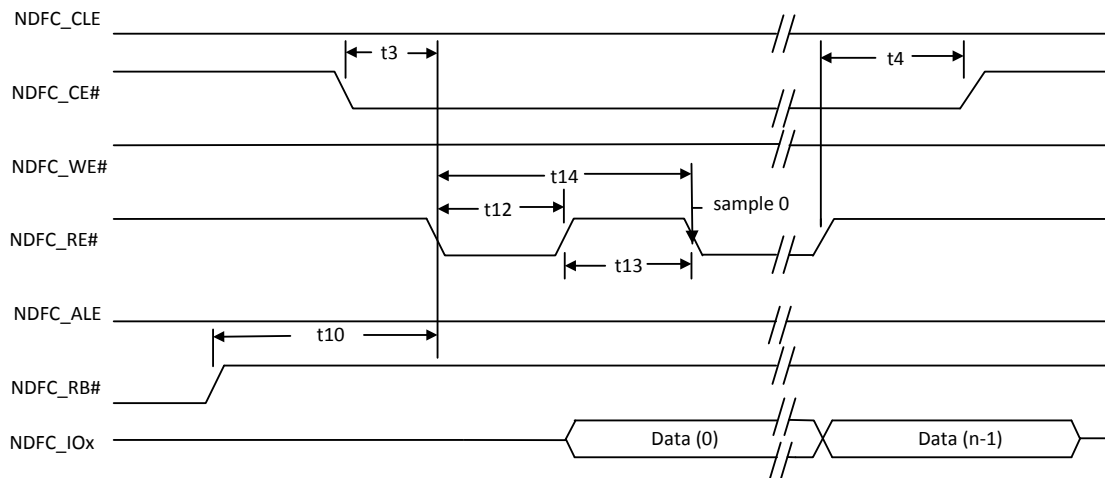
Typically, there are two kinds of serial access method. One method is conventional method which fetching data at the rise edge of NDFC\_RE# signal line. Another one is EDO type which fetching data at the next fall edge of NDFC\_RE# signal line.

#### Conventional Serial Access after Read Cycle (SAM0)

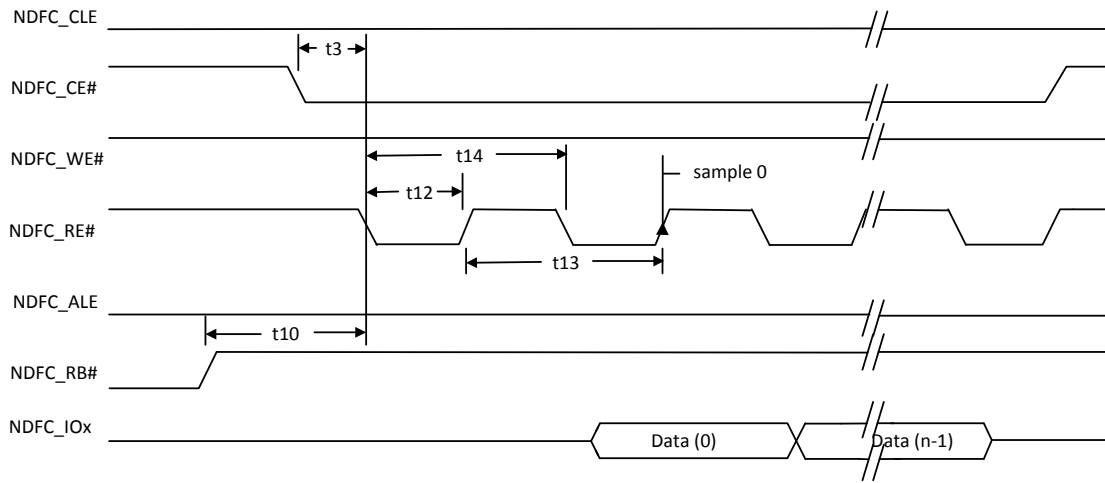


Conventional Serial Access Cycle Diagram (SAM0)

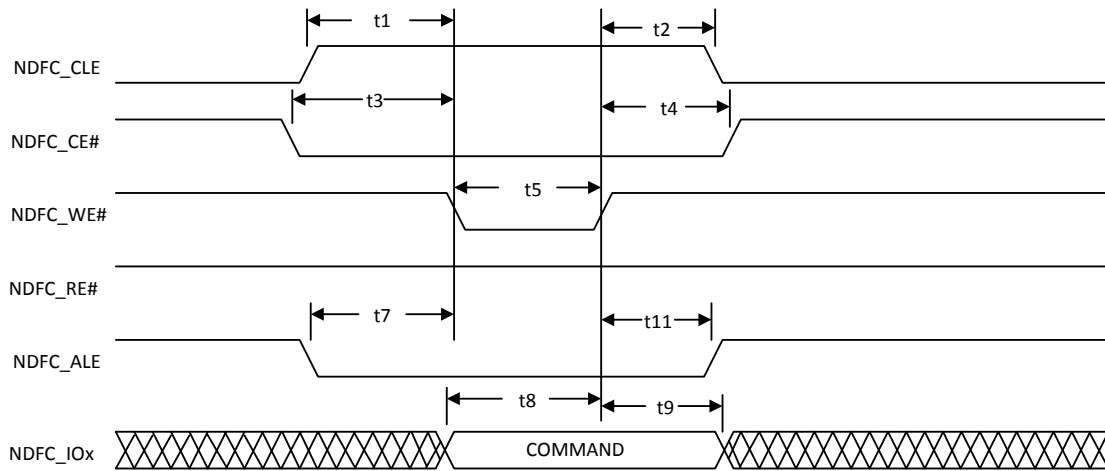
#### EDO type Serial Access after Read Cycle (SAM1)



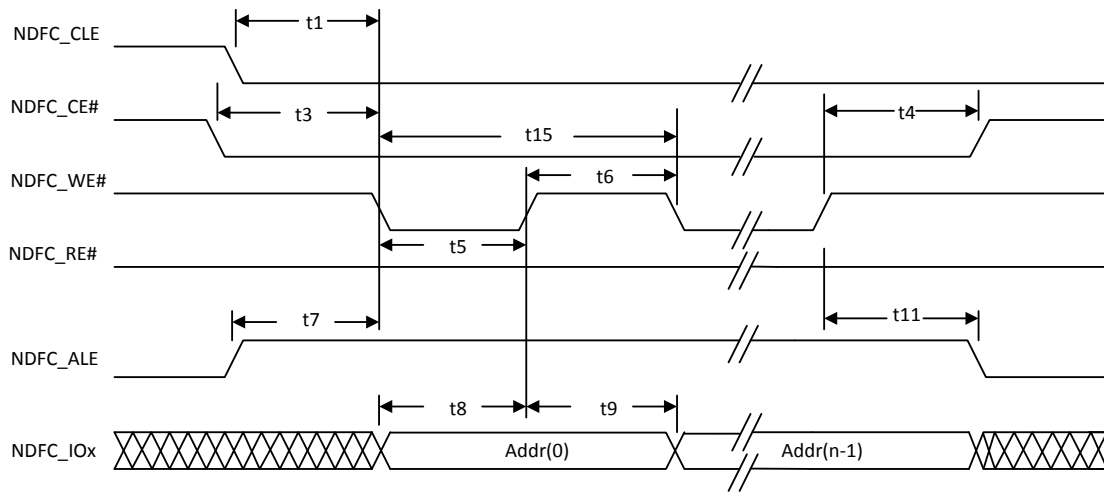
EDO type Serial Access after Read Cycle (SAM1)



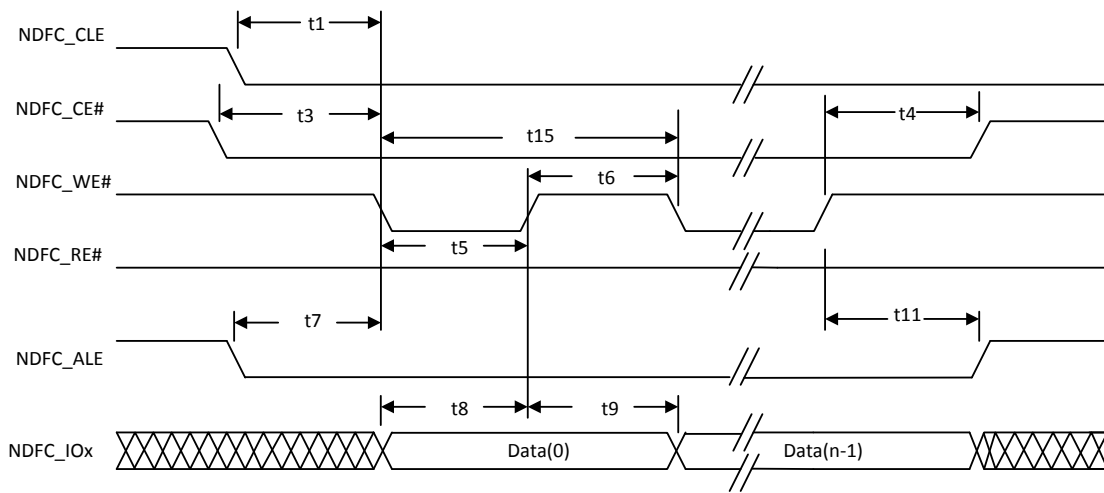
Extending EDO type Serial Access Mode (SAM2)



Command Latch Cycle

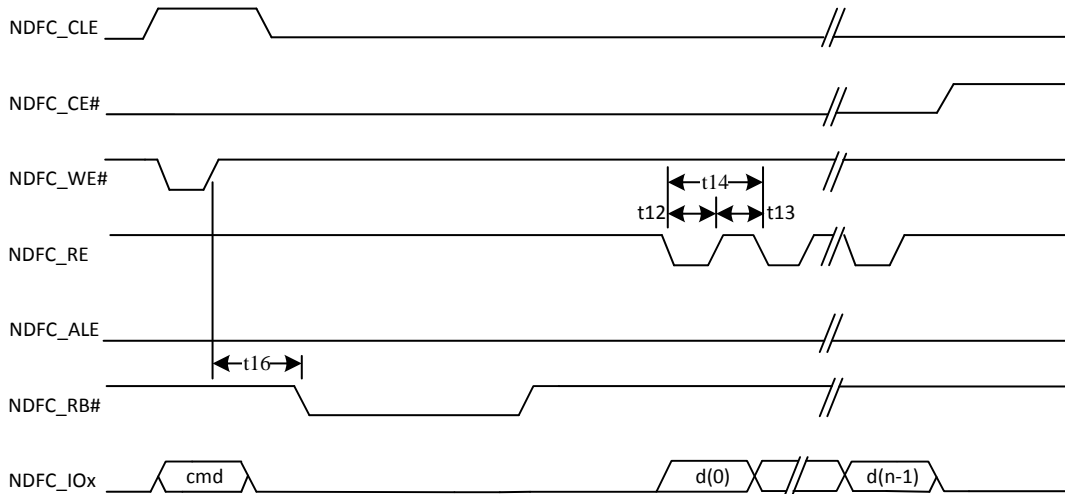


Address Latch Cycle

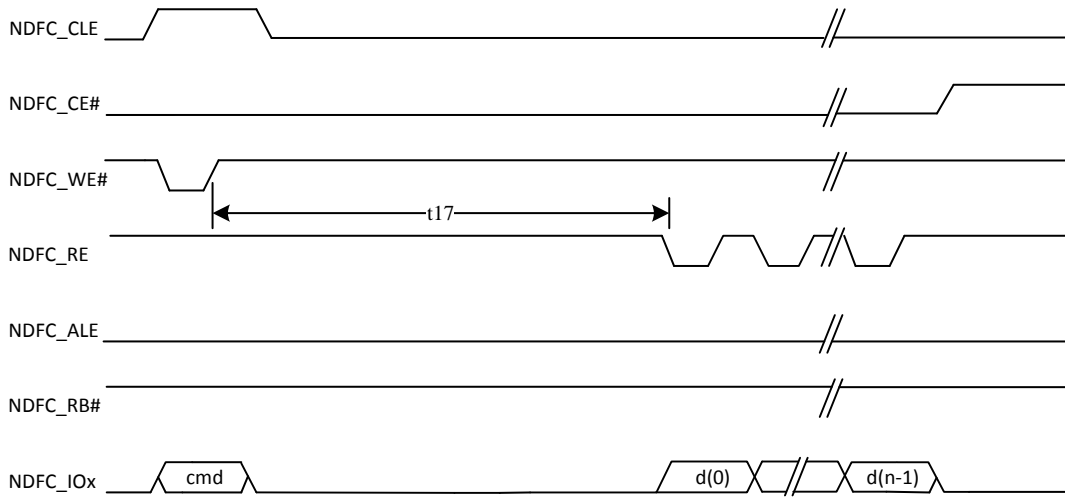


Write Data to Flash Cycle

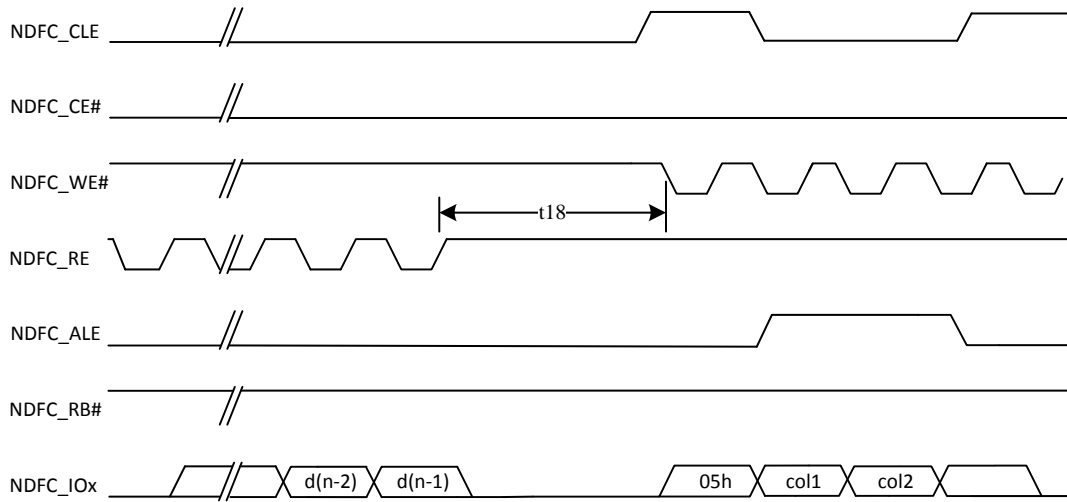




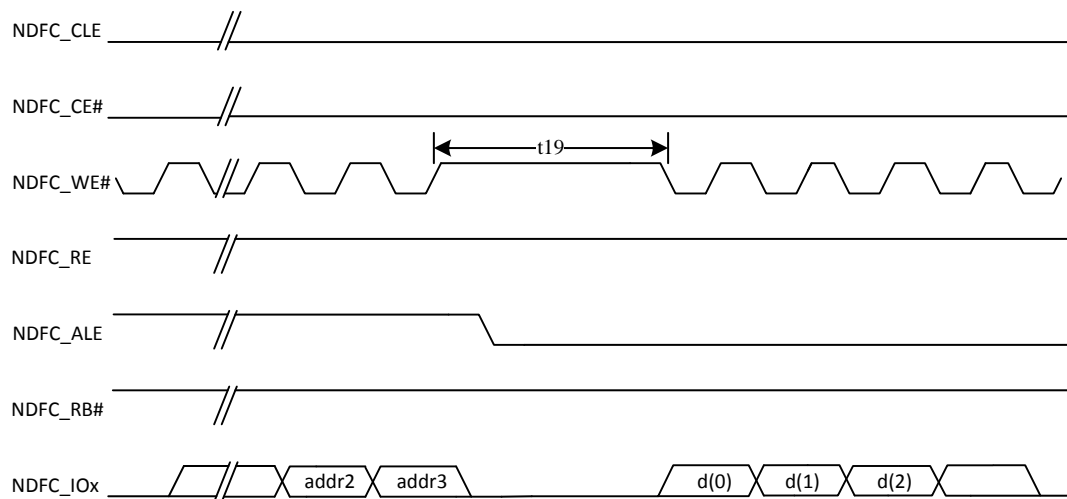
Waiting R/B# ready Diagram



WE# high to RE# low Timing Diagram



RE# high to WE# low Timing Diagram



Address to Data Loading Timing Diagram

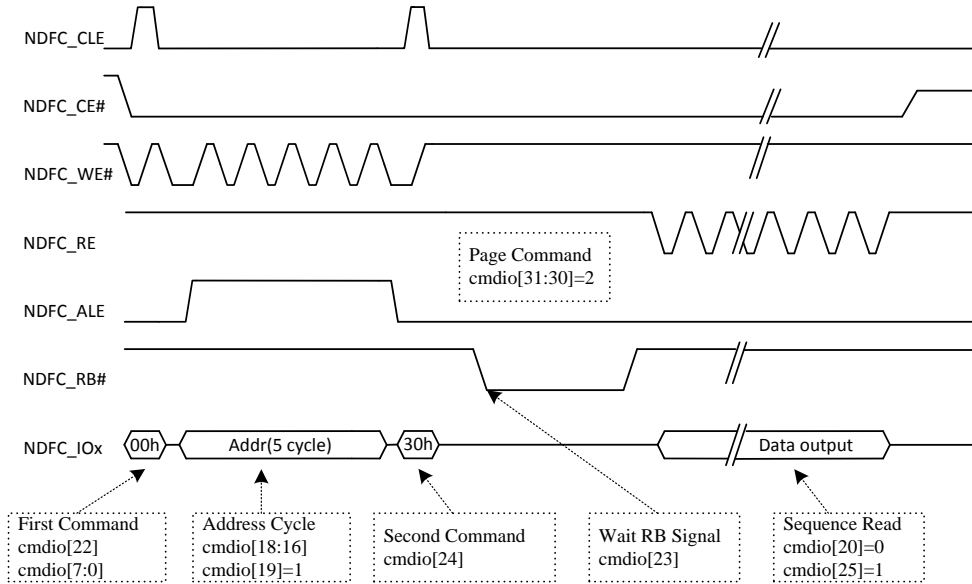
Timing cycle list:

ID	Parameter	Timing	Notes
t1	NDFC_CLE setup time	T	
t2	NDFC_CLE hold time	T	
t3	NDFC_CE setup time	T	
t4	NDFC_CE hold time	T	
t5	NDFC_WE# pulse width	T	
t6	NDFC_WE# hold time	T	

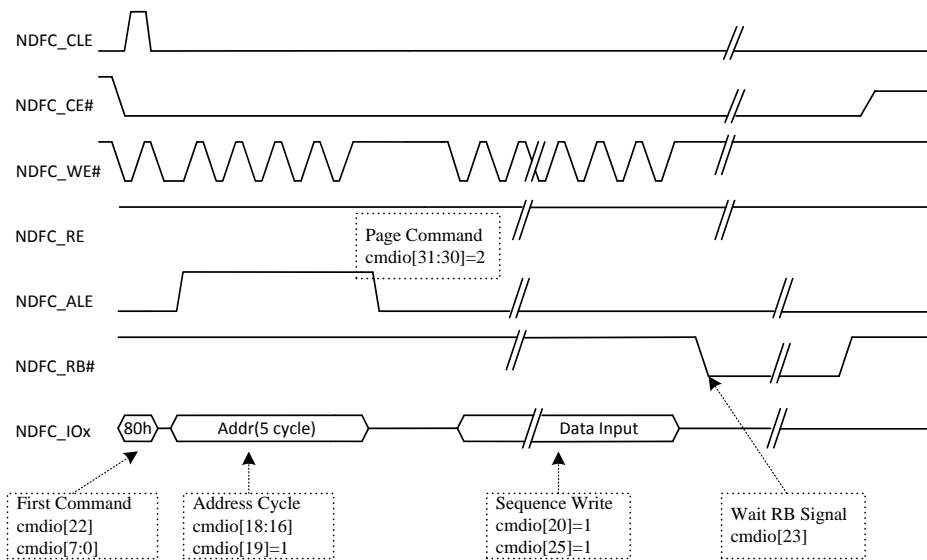
t7	NDFC_ALE setup time	T	
t8	Data setup time	T	
t9	Data hold time	T	
t10	Ready to NDFC_RE# low	3T	
t11	NDFC_ALE hold time	T	
t12	NDFC_RE# pulse width	T	
t13	NDFC_RE# hold time	T	
t14	Read cycle time	2T	
t15	Write cycle time	2T	
t16	NDFC_WE# high to R/B# busy	tWB	Specified by timing configure register (NDFC_TIMING_CFG)
t17	NDFC_WE# high to NDFC_RE# low	tWHR	Specified by timing configure register (NDFC_TIMING_CFG)
t18	NDFC_RE# high to NDFC_WE# low	tRHW	Specified by timing configure register (NDFC_TIMING_CFG)
t19	Address to Data Loading time	tADL	Specified by timing configure register (NDFC_TIMING_CFG)

Notes: T is the clock period duration of NDFC\_CLK (x2).

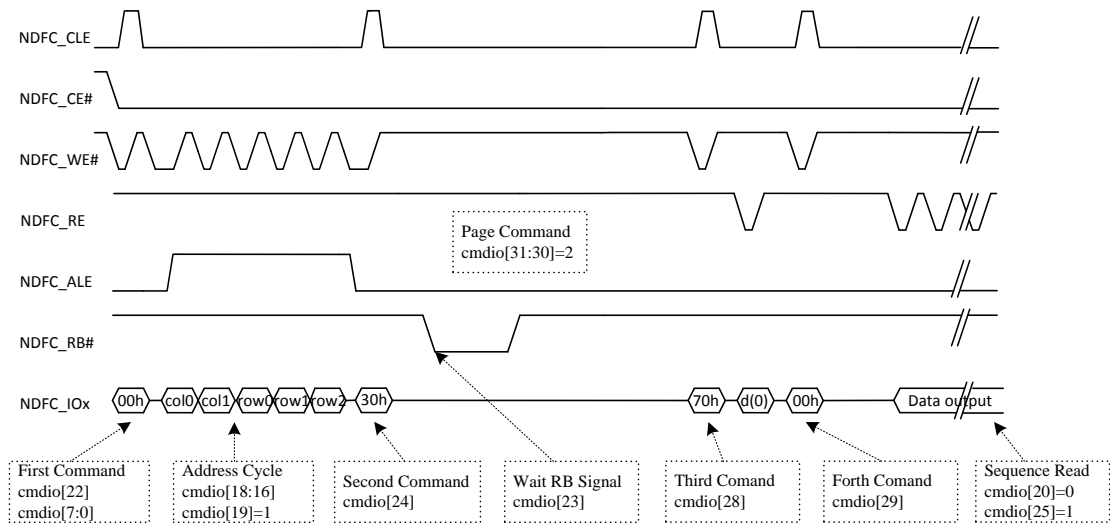
### 4.2.4. NDFC Operation Guide



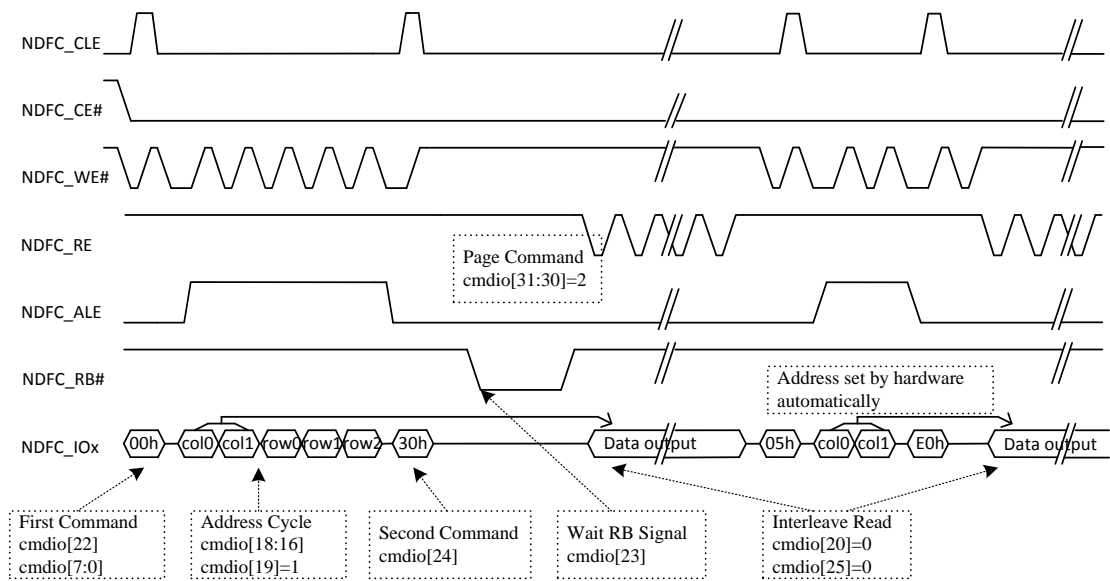
Page Read Command Diagram



Page Program Diagram



EF-NAND Page Read Diagram



Interleave Page Read Diagram

## Chapter 5 Graphic

This section describes the graphic of A80 from two aspects:

- GPU control
- Mixer processor

## 5.1. GPU Control

GPU control module is mainly used to control GPU internal power response and BIST test.

### 5.1.1. GPU Control Module Register List

Module Name	Base Address
GCM	0x01C08000

Register Name	Offset	Description
GCM_IDLE_STATUS_REG	0x0008	GPU Idle Status Register
GCM_QOS_REG	0x000C	GPU QOS Register
GCM_INT_PWROFF_GATING_REG	0x0010	GPU Internal Power Off Gating Register
GCM_INT_PWR_MOD_REG	0x0014	GPU Internal Power Event Process Mode Register
GCM_INT_PWR_DLY_REG	0x0018	GPU Internal Power Event Delay Register
GCM_INT_PWR_EVENT_REQ_REG	0x001C	GPU Internal Power Event Request Register
GCM_INT_PWR_RESPONSE_REQ	0x0020	GPU Internal Power Event Response Register

## 5.1.2. GPU Control Module Register Description

### 5.1.2.1. GCM\_IDLE\_STATUS\_REG

Offset: 0x8			Register Name: GCM_IDLE_STATUS_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R	0x1	IDLE_STATUS  Idle_status  GPU idle status  0:gpu is busy  1:gpu is idle

### 5.1.2.2. GCM\_QOS\_REG

Offset: 0xC			Register Name: GCM_QOS_REG
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x40	MAX_WRITE_CMD_LEN
23:16	R/W	0x40	MAX_READ_CMD_LEN
15:8	/	/	/
7:4	R/W	0x0	AXI_WR_QOS
3:0	R/W	0x0	AXI_RD_QOS

### 5.1.2.3. GCM\_INT\_PWROFF\_GATING\_REG

Offset: 0x10			Register Name: GCM_INT_PWROFF_GATING_REG
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x1	INT_PWROFF_GATING  Gating the corresponding modules when GPU Rascal/Dust power on/off  0:Invalid



			1:Valid  Note:this bit should be set to 1 before GPU Rascal/Dust power off while it should be set to 0 after GPU Rascal/Dust power on
7:0	R/W	0xFF	INT_PWROFF_EN  GPU Rascal/Dust power on/off  0:ON  1:OFF  These bits can be written only in software mode.

#### 5.1.2.4. GCM\_INT\_PWR\_MOD\_REG

Offset: 0x14			Register Name: GCM_INT_PWR_MOD_REG
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	GPU_INT_PWR_INTERRUPT_EN  GPU internal power event interrupt enable  0: disable  1: enable
7:2	/	/	/
1	R/W	0x1	GPU_INT_PWR_HW_RES_MOD  GPU internal power event hardware response mode  0: abort  1: complete  Note: This bit will be valid only when GPU power event software process disable.
0	R/W	0x0	GPU_INT_PWR_SW_ENABLE  GPU internal power event software process enable  0:disable  1:enable

			Note: when software process enable, CPU received power_event_req IRQ, power on/off according event_type, then set abort or complete after power status is stable. When software process disable, hardware do these works if power_event_req pull high.
--	--	--	--

#### 5.1.2.5. GCM\_INT\_PWR\_DLY\_REG

Offset: 0x18			Register Name: GCM_INT_PWR_DLY_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	/	/	/
15:8	R/W	0x8	GPU_INT_PWR_DLY1  GPU internal power event process delay, the clock of counter is 32k. This delay means the interval between all power bits on/off and power stable
7:0	R/W	0x8	GPU_INT_PWR_DLY  GPU internal power event process delay, the clock of counter is 32k. This delay is interval between opening/closing each power on/off bit.  Note: in hardware mode, this value means the interval between hardware setting each internal power gate on/off bit.

#### 5.1.2.6. GCM\_INT\_PWR\_EVENT\_REQ\_REG

Offset: 0x1C			Register Name: GCM_INT_PWR_EVENT_REQ_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R	0x0	PWR_EVENT_TYPE  GPU internal power event type  0: Power off  1: Power on
0	R	0x0	PWR_EVENT_REQ  GPU internal power event request pending  0: no power event request;  1: power event request pending

### 5.1.2.7. GCM\_INT\_PWR\_RESPONSE\_REG

Offset: 0x20			Register Name: GCM_INT_PWR_RESPONSE_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	<p>PWR_EVENT_ABORT</p> <p>0: no effect</p> <p>1: abort power event response</p> <p>Note: this bit will be cleared automatically when internal power event request pull down</p>
0	R/W	0x0	<p>PWR_EVENT_COMPLETE</p> <p>0: no effect</p> <p>1: power event process completed</p> <p>Note: this bit will be cleared automatically when internal power event request pull down</p>

## 5.2. Mixer Processor

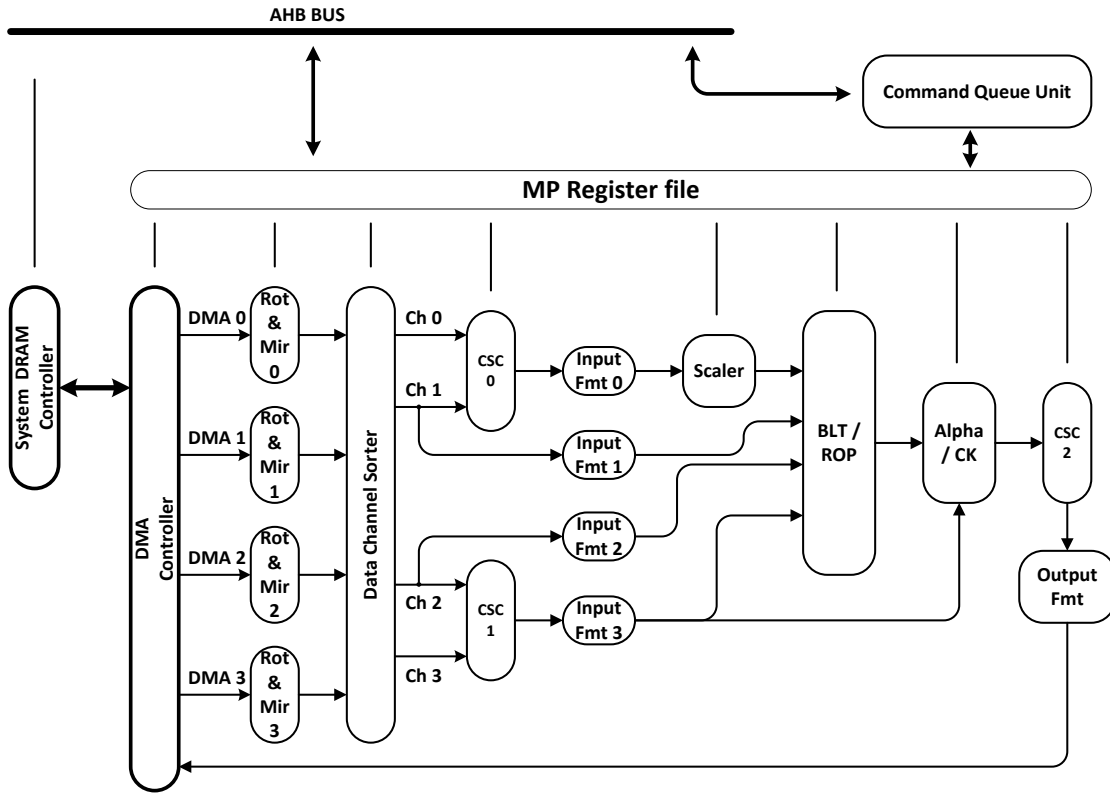
### 5.2.1. Overview

Mixer Processor is 2D graphic engine, it can read data from SDRAM and do some operation as covert data format, rotate, scalar or doing ROP operation to the data etc.

It features:

- Support buffer size up to 8192x8192 pixels
- Support window clipping
- Support alpha blending
- Support color key
- Support pre-multiply alpha image data
- Support 0/90/180/270 degree rotate
- Support horizontal/vertical flip
- Support memory scanning order
- Support rectangle fill color
- Support BitBlit, StretchBlit and MaskBlit
- Support 4x4taps 32 phase scaling
- Support color space convert
- Support input formats:1/2/4/8bpp mono and palette format RGB565/ARGB1555/ARGB4444/ARGB8888/iYUV422/iYUV444/YUV422 UVC/YUV420 UVC/YUV411 UVC
- Support output formats: 1/2/4/8bpp mono format RGB565/ARGB1555/ARGB4444/ARGB8888/iYUV422/iYUV444/YUV422 UVC/YUV420 UVC/YUV411 UVC/YUV422/YUV420/YUV411

5.2.2. Block Diagram



Mixer Processor General Diagram

### 5.2.3. MP Register List

Module name	Base address
MP	0x03f00000

Register name	Offset	Description
MP_CTL_REG	0x0	Mixer control register
MP_STS_REG	0x4	Mixer Status register
MP_IDMAGLBCTL_REG	0x8	Input DMA globe control register
MP_IDMA_H4ADD_REG	0xC	Input DMA start address high 4bits register
MP_IDMA_L32ADD_REG	0x10 – 0x1C	Input DMA start address low 32bits register
MP_IDMALINEWIDTH_REG	0x20 – 0x2C	Input DMA line width register
MP_IDMASIZE_REG	0x30 – 0x3C	Input DMA memory block size register
MP_IDMACOOR_REG	0x40 – 0x4C	Input DMA memory block coordinate control register
MP_IDMASET_REG	0x50 – 0x5C	Input DMA setting register
MP_IDMAFILLCOLOR_REG	0x60 – 0x6C	Input DMA fill-color register
MP_CSC0CTL_REG	0x74	Color space converter 0 control register
MP_CSC1CTL_REG	0x78	Color space converter 1 control register
MP_SCACTL_REG	0x80	Scaler control register
MP_SCAOUTSIZE_REG	0x84	Scaling output size register
MP_SCAHORFCT_REG	0x88	Scaler horizontal scaling factor register
MP_SCAVERFCT_REG	0x8C	Scaler vertical scaling factor register
MP_SCAHORPHASE_REG	0x90	Scaler horizontal start phase setting register
MP_SCAVERPHASE_REG	0x94	Scaler vertical start phase setting register
MP_ROPCTL_REG	0xB0	ROP control register
MP_ROPIDX0CTL_REG	0xB8	ROP channel 3 index 0 control table setting register
MP_ROPIDX1CTL_REG	0xBC	ROP channel 3 index 1 control table setting register

MP_ALPHACKCTL_REG	0xC0	Alpha / Color key control register
MP_CKMIN_REG	0xC4	Color key min color register
MP_CKMAX_REG	0xC8	Color key max color register
MP_ROPOUTFILLCOLOR_REG	0xCC	Fill color of ROP output setting register
MP_CSC2CTL_REG	0xD0	Color space converter 2 control register
MP_OUTCTL_REG	0xE0	Output control register
MP_OUTSIZE_REG	0xE8	Output size register
MP_OUTH4ADD_REG	0xEC	Output address high 4bits register
MP_OUTL32ADD_REG	0xF0 – 0xF8	Output address low 32bits register
MP_OUTLINEWIDTH_REG	0x100 – 0x108	Output line width register
MP_OUTALPHACTL_REG	0x120	Output alpha control register
MP_MBCTL_REG	0x130 – 0x13C	MB control register
MP_ICSCYGCOEF_REG	0x180 – 0x188	CSC0/1 Y/G coefficient register
MP_ICSCYGCONS_REG	0x18C	CSC0/1 Y/G constant register
MP_ICSCURCOEF_REG	0x190 – 0x198	CSC0/1 U/R coefficient register
MP_ICSCURCONS_REG	0x19C	CSC0/1 U/R constant register
MP_ICSCVBCOEF_REG	0x1A0 – 0x1A8	CSC0/1 V/B coefficient register
MP_ICSCVBCONS_REG	0x1AC	CSC0/1 V/B constant register
MP_OCSCYGCOEF_REG	0x1C0 – 0x1C8	CSC2 Y/G coefficient register
MP_OCSCYGCONS_REG	0x1CC	CSC2 Y/G constant register
MP_OCSCURCOEF_REG	0x1D0 – 0x1D8	CSC2 U/R coefficient register
MP_OCSCURCONS_REG	0x1DC	CSC2 U/R constant register
MP_OCSCVBCOEF_REG	0x1E0 – 0x1E8	CSC2 V/B coefficient register
MP_OCSCVBCONS_REG	0x1EC	CSC2 V/B constant register
CMDQUECTL_REG	0x140	Command queue control register
CMDQUESTS_REG	0x144	Command queue status register
CMDQUEADD_REG	0x148	Command queue storage start address register

Memories	0x200 – 0x27C	Scaling horizontal filtering coefficient RAM block
Memories	0x280 – 0x2FC	Scaling vertical filtering coefficient RAM block
Memories	0x400 – 0x7FF	Palette table



## 5.2.4. MP Register Description

### 5.2.4.1. Mixer Control Register

Offset: 0x0			Register Name: MP_CTL_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9	R/W	0	HWERRIRQ_EN  Hardware error IRQ enable control  0:disable  1:enable
8	R/W	0	FINISHIRQ_EN  Mission finish IRQ enable control  0:disable  1:enable
7:2	/	/	/
1	R/W	0	START_CTL  Start control  If the bit is set, the module will start 1 frame operation and stop auto.
0	R/W	0	MP_EN  Enable control

			0:disable  1:enable
--	--	--	---------------------------

#### 5.2.4.2. Mixer Status Register

Offset: 0x4			Register Name: MP_STS_REG
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R	0	HWERR_FLAG  Hardware error status
12	R	0	BUSY_FLAG  Module working status  0:idle  1:running
11:10	/	/	/
9	R/W	0	HWERRIRQ_FLAG  Hardware error IRQ  It will be set when hardware error occur, and cleared by writing 1.
8	R/W	0	FINISHIRQ_FLAG  Mission finish IRQ  It will be set when 1 frame operation accomplished, and cleared by writing 1.
7:0	/	/	/

**5.2.4.3. Input DMA Globe Control Register**

Offset: 0x8			Register Name: MP_IDMAGLBCTL_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9:8	R/W	0	MEMSCANORDER  Memory scan order selection  0:  Top to down  Left to right  1:  Top to down  Right to left  2:  Down to top  Left to right  3:  Down to top  Right to left

			<p>Note:</p> <p>----Four input DMA channel use the same scan rule.</p> <p>----The each output DMA channel should match the same memory scan order rule with the input DMA channel.</p>
7:0	/	/	/

#### 5.2.4.4. Input DMA Start Address High 4bits Register

Offset: 0xC			Register Name: MP_IDMA_H4ADD_REG
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0	IDMA3_H4ADD  iDMA3  High 4bits address in bits
23:20	/	/	/
19:16	R/W	0	IDMA2_H4ADD  iDMA2  High 4bits address in bits
15:12	/	/	/
11:8	R/W	0	IDMA1_H4ADD  iDMA1  High 4bits address in bits

7:4	/	/	/
3:0	R/W	0	IDMA0_H4ADD  iDMA0  High 4bits address in bits

**5.2.4.5. Input DMA Start Address Low 32bits Register**

Offset:			Register Name: MP_IDMA_L32ADD_REG
iDMA0:0x10			
iDMA1:0x14			
iDMA2:0x18			
iDMA3:0x1C			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0	IDMA_L32ADD  iDMA  Low 32bits address in bits

**5.2.4.6. Input DMA Line Width Register**

Offset:		Register Name: MP_IDMALINEWIDTH_REG
iDMA0:0x20		
iDMA1:0x24		

iDMA2:0x28			
iDMA3:0x2C			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0	IDMA_LINEWIDTH  iDMA  Line width in bits

#### 5.2.4.7. Input DMA Memory Block Size Register

Offset:			Register Name: MP_IDMASIZE_REG
iDMA0:0x30			
iDMA1:0x34			
iDMA2:0x38			
iDMA3:0x3C			
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0	IDMA_HEIGHT  Memory block height in pixels  The height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	0	IDMA_WIDTH

			Memory block width in pixels  The width = The value of these bits add 1
--	--	--	---

#### 5.2.4.8. Input DMA Memory Block Coordinate Control Register

Offset:  iDMA0:0x40  iDMA1:0x44  iDMA2:0x48  iDMA3:0x4C			Register Name: MP_IDMACOOR_REG
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0	IDMA_YCOOR  Y coordinate  Y is the left-top y coordinate of layer on output window in pixels  The Y represent the two's complement
15:0	R/W	0	IDMA_XCOOR  X coordinate  X is left-top x coordinate of the layer on output window in pixels  The X represent the two's complement

#### 5.2.4.9. Input DMA setting register

Offset:			Register Name: MP_IDMASET_REG
iDMA0:0x50			
iDMA1:0x54			
iDMA2:0x58			
iDMA3:0x5C			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	IDMA_GLBALPHA  Globe alpha value
23	/	/	/
22	R/W	0	MBFMT  Micro block format enable  0: disable  1: enable
21:20	R/W	0	MBSIZE_Y  Micro block size in bytes in y direction  0: 16  1: 32  2: 64  3: 128



19:17	R/W	0	<p>MBSIZE_X</p> <p>Micro block size in bytes in x direction</p> <p>0: 16</p> <p>1: 32</p> <p>2: 64</p> <p>3: 128</p> <p>Other: reserved</p>
16	R/W	0	<p>IDMA_FCMODEN</p> <p>Fill color mode enable control</p> <p>0: disable</p> <p>1: enable</p>
15:12	R/W	0	<p>IDMA_PS</p> <p>Input data pixel sequence</p> <p>Reference input pixel sequence table</p>
11:8	R/W	0	<p>IDMA_FMT</p> <p>Input data format</p> <p>0x0:32bpp – A8R8G8B8 or interleaved AYUV8888</p> <p>0x1:16bpp – A4R4G4B4</p> <p>0x2:16bpp – A1R5G5B5</p>

			<p>0x3:16bpp – R5G6B5</p> <p>0x4:16bpp – interleaved YUV422</p> <p>0x5:16bpp – U8V8</p> <p>0x6:8bpp – Y8</p> <p>0x7:8bpp – MONO or palette</p> <p>0x8:4bpp – MONO or palette</p> <p>0x9:2bpp – MONO or palette</p> <p>0xa:1bpp – MONO or palette</p> <p>Other: reserved</p> <p>Note: if the input data format is 16 or 32bpp, and the work mode is palette mode, only the low 8 bits input data is valid.</p>
7:4	R/W	0	<p>IDMA_ROTMIIRCTL</p> <p>Rotation and mirroring control</p> <p>0:normal</p> <p>1:X</p> <p>2:Y</p> <p>3:XY</p> <p>4:A</p>

			5:AX  6:AY  7:AXY  Other: reserved
3:2	R/W	0	IDMA_ALPHACTL  Alpha control  0:Ignore  Output alpha value = pixels alpha, if no pixel alpha, the alpha value equal 0xff  1:Globe alpha enable  Ignore pixel alpha value  Output alpha value = globe alpha value  2: Globe alpha mix pixel alpha  Output alpha value = globe alpha value * pixels alpha value  3:Reserved  Note: the output alpha value here means the input alpha value of the ALU following the DMA controller.

1	R/W	0	IDMA_WORKMOD  Work mode selection  0: normal mode ( non-palette mode )  1: palette mode
0	R/W	0	IDMA_EN  Input DMA enable control  0:disable input DMA channel, the respective fill-color value will stand of the input data.  1:enable

#### 5.2.4.10. Input DMA Fill-Color Register

Offset:		Register Name: MP_IDMAFILLCOLOR_REG	
iDMA0:0x60			
iDMA1:0x64			
iDMA2:0x68			
iDMA3:0x6C			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	IDMA_FCALPHA  Alpha
23:16	R/W	0	IDMA_FCRED  Red

15:8	R/W	0	IDMA_FCGREEN  Green
7:0	R/W	0	IDMA_FCBLUE  Blue

#### 5.2.4.11. Color Space Converter 0 Control Register

Offset: 0x74			Register Name: MP_CSC0CTL_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:4	R/W	0	<p>CSC0_DATAMOD</p> <p>Data mode control</p> <p>0:  Interleaved AYUV8888 mode</p> <p>1:  Interleaved YUV422 mode</p> <p>In mode 0 and mode 1, only the channel 0 data path is valid for this module, the channel 1 data flow will by-pass the csc0 module, and direct to input formatter 1.</p> <p>2:Planar YUV422 mode (UV combined only)</p>

			<p>3:Planar YUV420 mode (UV combined only)</p> <p>4:Planar YUV411 mode (UV combined only)</p> <p>In mode 2/3/4, following rule:</p> <p>---In this mode, the output data of the input formatter 1 will be stead of the respective fill-color value.</p>
3:1	/	/	/
0	R/W	0	<p>CSC0_EN</p> <p>Enable control</p> <p>0:</p> <p>Disable color space function, ignore the control setting, and the data flow will by-pass the module.</p> <p>1:</p> <p>Enable color space converting function.</p>

#### 5.2.4.12. Color Space Converter 1 Control Register

Offset: 0x78			Register Name: MP_CSC1CTL_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:4	R/W	0	<p>CSC1_DATAMOD</p> <p>Data mode control</p>

			<p>0:</p> <p>Interleaved AYUV8888 mode</p> <p>1:</p> <p>Interleaved YUV422 mode</p> <p>In mode 0 and mode 1, only the channel 3 data path is valid for this module, the channel 2 data flow will by-pass the csc1 module, and direct to input formatter 2.</p> <p>2:Planar YUV422 mode (UV combined only)</p> <p>3:Planar YUV420 mode (UV combined only)</p> <p>4:Planar YUV411 mode (UV combined only)</p> <p>In mode 2/3/4, following rule:</p> <p>----In this mode, the output data of the input formatter 2 will be stead of the respective fill-color value.</p>
3:1	/	/	/
0	R/W	0	<p>CSC1_EN</p> <p>Enable control</p> <p>0:</p>

			<p>Disable color space function, ignore the control setting, and the data flow will by-pass the module.</p> <p>1:</p> <p>Enable color space converting function.</p>
--	--	--	--

#### 5.2.4.13. Scaler Control Register

Offset: 0x80			Register Name: MP_SCACTL_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5:4	R/W	0	<p>SCA_ALGSEL</p> <p>Scaling algorithm selection</p> <p>0: bi-cubic(4 taps in vertical and horizontal)</p> <p>1: linear in vertical and bi-linear in horizontal(2 taps in vertical and 4 taps in horizontal)</p> <p>2: extractive in vertical and bi-linear in horizontal(1 tap in vertical and 4 taps in horizontal)</p> <p>3: reserved</p>
3:1	/	/	/
0	R/W	0	<p>SCA_EN</p> <p>Enable control</p> <p>0:</p> <p>Disable scalar, ignore the whole scaling setting, and the data flow will by-pass the</p>



			module.  1:  Enable scaling function.
--	--	--	---

#### 5.2.4.14. Scaling Output Size Register

Offset: 0x84			Register Name: MP_SCAOUTSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0	SCA_OUTHEIGHT  Output height  The output height = The value of these bits add 1   The minimum output height is 8 pixels.
15:13	/	/	/
12:0	R/W	0	SCA_OUTWIDTH  Output width  The output width = The value of these bits add 1   The minimum output width is 16 pixels.

#### 5.2.4.15. Scaler Horizontal Scaling Factor Register

Offset: 0x88			Register Name: MP_SCAHORFCT_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0	SCA_HORINTFCT  The integer part of the horizontal scaling ratio  the horizontal scaling ratio = input width/output width
15:00	R/W	0	SCA_HORFRAFCT  The fractional part of the horizontal scaling ratio  the horizontal scaling ratio = input width/output width  The input width is the memory block width of respective iDMA channel.

#### 5.2.4.16. Scaler Vertical Scaling Factor Register

Offset: 0x8C			Register Name: MP_SCAVERFCT_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0	SCA_VERINTFCT  The integer part of the vertical scaling ratio  the vertical scaling ratio = input height/output height

15:00	R/W	0	<p>SCA_VERFRACT</p> <p>The fractional part of the vertical scaling ratio</p> <p>the vertical scaling ratio = input height /output height</p> <p>The input height is the memory block height of respective iDMA channel.</p>
-------	-----	---	---

#### 5.2.4.17. Scaler Horizontal Start Phase Setting Register

Offset: 0x90			Register Name: MP_SCAHORPHASE_REG
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:00	R/W	0	<p>SCA_HORPHASE</p> <p>Start phase in horizontal (complement)</p> <p>This value equals to start phase * 2<sup>16</sup></p>

#### 5.2.4.18. Scaler Vertical Start Phase Setting Register

Offset: 0x94			Register Name: MP_SCAVERPHASE_REG
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:00	R/W	0	<p>SCA_VERPHASE</p> <p>Start phase in vertical (complement)</p> <p>This value equals to start phase * 2<sup>16</sup></p>

**5.2.4.19. ROP Control Register**

Offset: 0xB0			Register Name: MP_ROPCTL_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:14	R/W	0	ROP_ALPHABYPASSESEL  ROP output Alpha channel selection  0: channel 0  1: channel 1  2: channel 2  3:reserved  Note: the bit is only valid in by-pass mode of Alpha channel
13:12	R/W	0	ROP_REDBYPASSESEL  ROP output Red channel selection  0: channel 0  1: channel 1  2: channel 2  3:reserved

			Note: the bit is only valid in by-pass mode of Red channel
11:10	R/W	0	<p>ROP_GREENBYPASSEL</p> <p>ROP output Green channel selection</p> <p>0: channel 0</p> <p>1: channel 1</p> <p>2: channel 2</p> <p>3:reserved</p> <p>Note: the bit is only valid in by-pass mode of Green channel</p>
9:8	R/W	0	<p>ROP_BLUEBYPASSEL</p> <p>ROP output Blue channel selection</p> <p>0: channel 0</p> <p>1: channel 1</p> <p>2: channel 2</p> <p>3:reserved</p> <p>Note: the bit is only valid in by-pass mode of Blue channel</p>
7	R/W	0	<p>ROP_ALPHABYPASSEN</p> <p>ROP Alpha channel by-pass enable control</p>

			0:pass through  1:by-pass
6	R/W	0	ROP_REDBYPASSEN  ROP Red channel by-pass enable control  0:pass through  1:by-pass
5	R/W	0	ROP_GREENBYPASSEN  ROP Green channel by-pass enable control  0:pass through  1:by-pass
4	R/W	0	ROP_BLUEBYPASSEN  ROP Blue channel by-pass enable control  0:pass through  1:by-pass
3:1	/	/	/
0	R/W	0	ROP_MOD  ROP type selection  0:ROP3  1:ROP4

			<p>----In ROP3 mode, only the value of 'channel 3 index 0 control table setting register' will be selected.</p> <p>----In ROP3 mode, the channel 3 data will by-pass the ROP module.</p> <p>----In ROP3 mode, the channel 3 data will direct to Alpha/CK module.</p> <p>----In ROP4 mode, the respective input DMA channel fill color of channel 3 will transfer to Alpha/CK module.</p>
--	--	--	--

#### 5.2.4.20. ROP Channel 3 Index 0 Control Table Setting Register

Offset: 0xB8			Register Name: MP_ROPIDX0CTL_REG
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0	CH2IGN_EN  Channel 2 ignore mode enable control  0:disable  1:enable  When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.
17	R/W	0	CH1IGN_EN  Channel 1 ignore mode enable control  0:disable

			<p>1:enable</p> <p>When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.</p>
16	R/W	0	<p>CH0IGN_EN</p> <p>Channel 0 ignore mode enable control</p> <p>0:disable</p> <p>1:enable</p> <p>When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.</p>
15	R/W	0	<p>NOD7_CTL</p> <p>Index 0 node7 setting ( channel 0' and channel 1' and channel 2' mix not logic )</p> <p>0:by-pass</p> <p>1:not</p>
14:11	R/W	0	<p>NOD6_CTL</p> <p>Index 0 node6 setting ( channel 0' and channel 1' and channel 2' mix logic )</p> <p>0:and</p> <p>1:or</p> <p>2:xor</p>



			3:add in byte  4:add in word (32bit)  5:multiply in byte  6:multiply in byte  7:channel 0' mix channel 1' then sub channel 2' in byte  8:channel 0' mix channel 1' then sub channel 2' in word (32bit)  Other: Reserved
10	R/W	0	NOD5_CTL  Index 0 node5 setting ( channel 0' and channel 1' mix not logic )  0:by-pass  1:not
9:6	R/W	0	NOD4_CTL  Index 0 node4 setting ( channel 0' and channel 1' mix logic )  0:and  1:or  2:xor  3:add in byte  4:add in word (32bit)  5:multiply in byte

			6:multiply in byte  7:channel 0' sub channel 1' in byte  8:channel 0' sub channel 1' in word (32bit)  Other: Reserved
5	R/W	0	NOD3_CTL  Index 0 node3 setting ( channel 2' not logic )  0:by-pass  1:not
4	R/W	0	NOD2_CTL  Index 0 node2 setting ( channel 1' not logic )  0:by-pass  1:not
3	R/W	0	NOD1_CTL  Index 0 node1 setting ( channel 0' not logic)  0:by-pass  1:not
2:0	R/W	0	NOD0_CTL  Index 0 node0 setting ( sorting control )  0:012

			1:021
			2:102
			3:120
			4:201
			5:210
			Other: Reserved

Note: the result of the add or multiply operation will select the high 8 (byte operation) or 32bits (word operation).

#### 5.2.4.21. ROP Channel 3 Index 1 Control Table Setting Register

Offset: 0xBC			Register Name: MP_ROPIDX1CTL_REG
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0	CH2IGN_EN  Channel 2 ignore mode enable control  0:disable  1:enable  When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.
17	R/W	0	CH1IGN_EN  Channel 1 ignore mode enable control

			<p>0:disable</p> <p>1:enable</p> <p>When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.</p>
16	R/W	0	<p>CHOIGN_EN</p> <p>Channel 0 ignore mode enable control</p> <p>0:disable</p> <p>1:enable</p> <p>When ignore mode is enabled, the data of channel will be ignored, and the data ZERO will be instead of the channel into the ROP module.</p>
15	R/W	0	<p>NOD7_CTL</p> <p>Index 1 node7 setting ( channel 0' and channel 1' and channel 2' mix not logic )</p> <p>0:by-pass</p> <p>1:not</p>
14:11	R/W	0	<p>NOD6_CTL</p> <p>Index 1 node6 setting ( channel 0' and channel 1' and channel 2' mix logic )</p> <p>0:and</p> <p>1:or</p>

			2:xor  3:add in byte  4:add in word (32bit)  5:multiply in byte  6:multiply in byte  7:channel 0' mix channel 1' then sub channel 2' in byte  8:channel 0' mix channel 1' then sub channel 2' in word (32bit)  Other: Reserved
10	R/W	0	NOD5_CTL  Index 1 node5 setting ( channel 0' and channel 1' mix not logic )  0:by-pass  1:not
9:6	R/W	0	NOD4_CTL  Index 1 node4 setting ( channel 0' and channel 1' mix logic )  0:and  1:or  2:xor  3:add in byte  4:add in word (32bit)

			5:multiply in byte  6:multiply in byte  7:channel 0' sub channel 1' in byte  8:channel 0' sub channel 1' in word (32bit)  Other: Reserved
5	R/W	0	NOD3_CTL  Index 1 node3 setting ( channel 2' not logic )  0:by-pass  1:not
4	R/W	0	NOD2_CTL  Index 1 node2 setting ( channel 1' not logic )  0:by-pass  1:not
3	R/W	0	NOD1_CTL  Index 1 node1 setting ( channel 0' not logic)  0:by-pass  1:not
2:0	R/W	0	NOD0_CTL  Index 1 node0 setting ( sorting control )

			0:012
			1:021
			2:102
			3:120
			4:201
			5:210
			Other: Reserved

Note: the result of the add or multiply operation will select the high 8 (byte operation) or 32bits (word operation).

#### 5.2.4.22. Alpha / Color Key Control Register

Offset: 0xC0			Register Name: MP_ALPHACKCTL_REG
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	CH3GALPHA  Ch3 globe alpha value of alpha / color key module
23:16	R/W	0	ROPGALPHA  ROP globe alpha value of alpha / color key module
15:14	R/W	0	CH3ALPHACTL  0:Ignore  Output alpha value = pixels alpha, the pixel alpha here means the mixed alpha value of Ch3 mixed alpha

			<p>1:Ch3 globe alpha enable</p> <p>Ignore pixel alpha value</p> <p>Output alpha value = Ch3 globe alpha value</p> <p>2: Globe alpha mix pixel alpha</p> <p>Output alpha value = Ch3 globe alpha value * pixels alpha value</p> <p>3:Reserved</p> <p>Note: the output alpha value here means the input alpha value of Alpha CK module.</p>
13:12	R/W	0	<p>ROPALPHACTL</p> <p>0:Ignore</p> <p>Output alpha value = pixels alpha, the pixel alpha here means the mixed alpha value of ROP module</p> <p>1:ROP globe alpha enable</p> <p>Ignore pixel alpha value</p> <p>Output alpha value = ROP globe alpha value</p>



			<p>2: Globe alpha mix pixel alpha</p> <p>Output alpha value = ROP globe alpha value * pixels alpha value</p> <p>3:Reserved</p> <p>Note: the output alpha value here means the input alpha value of Alpha CK module.</p>
11	/	/	/
10	R/W	0	<p>CK_REDCON</p> <p>Red control condition</p> <p>0: if (R value of ck min color) &lt;= (R value of layer0) &lt;= (R value of ck max color), The red control condition is true, else the condition is false.</p> <p>1: if (R value of ck min color) &gt; (R value of layer0) or (R value of layer0) &gt; (R value of ck max color), The red control condition is true, else the condition is false.</p>
9	R/W	0	<p>CK_GREENCON</p> <p>Green control condition</p> <p>0: if (G value of ck min color) &lt;= (G value of layer0) &lt;= (G value of ck max color), The green control condition is true, else the condition is false.</p>

			<p>1: if (G value of ck min color) &gt; (G value of layer0) or (G value of layer0) &gt; (G value of ck max color), The green control condition is true, else the condition is false.</p>
8	R/W	0	<p>CK_BLUECON</p> <p>Blue control condition</p> <p>0: if (B value of ck min color) &lt;= (B value of layer0) &lt;= (B value of ck max color), The blue control condition is true, else the condition is false.</p> <p>1: if (B value of ck min color) &gt; (B value of layer0) or (B value of layer0) &gt; (B value of ck max color), The blue control condition is true, else the condition is false.</p>
7	R/W	0	<p>ICH3_PREMUL</p> <p>0: normal data 1: pre-multiply input data</p>
6	R/W	0	<p>IROP_PREMUL</p> <p>0: normal data 1: pre-multiply input data</p>
5	R/W	0	<p>O_PREMUL</p>

			<p>0: output normal data</p> <p>1: output pre-multiply data</p>
4	R/W	0	<p>PRI</p> <p>Priority selection</p> <p>0: ROP output channel is higher than channel 3</p> <p>1: Channel 3 is higher than ROP output channel</p>
3	/	/	/
2:1	R/W	0	<p>ALPHACK_MOD</p> <p>Alpha / Color key mode selection</p> <p>0: alpha mode</p> <p>1: color key mode, using the high priority layer as matching condition, if it is true, the low priority layer pass.</p> <p>2: color key mode, using the low priority layer as matching condition, if it is true, the high priority layer pass.</p> <p>3: Reserved</p>
0	R/W	0	<p>ALPHACK_EN</p> <p>Enable control</p> <p>0: the ROP data will by-pass the alpha/ck module</p> <p>1: enable</p> <p>Note: if the module is disabled, the data of channel 3 will be ignored, and only</p>

			the ROP data will pass through to CSC2 module.
--	--	--	--

#### 5.2.4.23. Color Key Min Color Register

Offset: 0xC4			Register Name: MP_CKMIN_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0	CKMIN_R Red
15:8	R/W	0	CKMIN_G Green
7:0	R/W	0	CKMIN_B Blue

#### 5.2.4.24. Color Key Max Color Register

Offset: 0xC8			Register Name: MP_CKMAX_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0	CKMAX_R Red
15:8	R/W	0	CKMAX_G Green
7:0	R/W	0	CKMAX_B Blue

**5.2.4.25. Fill Color Of ROP Output Setting Register**

Offset: 0xCC			Register Name: MP_ROPOUTFILLCOLOR_REG
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	Alpha
23:16	R/W	0	Red
15:8	R/W	0	Green
7:0	R/W	0	Blue

**5.2.4.26. Color Space Converter 2 Control Register**

Offset: 0xD0			Register Name: MP_CSC2CTL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0	CSC2_EN  Enable control  0:  Disable color space function, ignore the control setting, and the data flow will by-pass the module.  1:  Enable color space converting function.

**5.2.4.27. Output Control Register**

Offset: 0xE0			Register Name: MP_OUTCTL_REG
Bit	Read/Write	Default/Hex	Description

	e	ex	
31:12	/	/	/
11:8	R/W	0	OUT_PS  Output data pixel sequence  Reference output pixel sequence table
7	R/W	0	RND_EN  Round enable  0:disabled  1:enabled
6:4	/	/	/
3:0	R/W	0	OUT_FMT  Output data format  0x0: 32bpp – A8R8G8B8 or interleaved AYUV8888  0x1: 16bpp – A4R4G4B4  0x2: 16bpp – A1R5G5B5  0x3: 16bpp – R5G6B5  0x4: 16bpp – interleaved YUV422  0x5: planar YUV422 (UV combined)  0x6: planar YUV422  0x7: 8bpp – MONO

		<p>0x8: 4bpp – MONO</p> <p>0x9: 2bpp – MONO</p> <p>0xa: 1bpp – MONO</p> <p>0xb: planar YUV420 (UV combined)</p> <p>0xc: planar YUV420</p> <p>0xd: planar YUV411 (UV combined)</p> <p>0xe: planar YUV411</p> <p>Other: reserved</p> <p>Note: In all YUV output data format, the CSC2 must be enabled, otherwise the output data mode will be 32bpp A8R8G8B8 mode.</p>
--	--	--

Output data mode and output data ports mapping:

Output data mode	Output data channel selection		
	Channel 0	Channel 1	Channel 2
A8R8G8B8 or interleaved AYUV8888	ARGB or AYUV	Ignore	Ignore
A4R4G4B4	ARGB	Ignore	Ignore
A1R5G5B5	ARGB	Ignore	Ignore
R5G6B5	RGB	Ignore	Ignore
interleaved YUV422	YUV	Ignore	Ignore
planar YUV422 (UV combined)	Y	UV	Ignore
planar YUV422	Y	U	V
8bpp – MONO	MONO	Ignore	Ignore

4bpp – MONO	MONO	Ignore	Ignore
2bpp – MONO	MONO	Ignore	Ignore
1bpp – MONO	MONO	Ignore	Ignore
planar YUV420 (UV combined)	Y	UV	Ignore
planar YUV420	Y	U	V
planar YUV411 (UV combined)	Y	UV	Ignore
planar YUV411	Y	U	V

#### 5.2.4.28. Output Size Register

Offset: 0xE8			Register Name: MP_OUTSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0	OUT_HEIGHT  Height  The value add 1 equal the actual output image height
15:13	/	/	/
12:0	R/W	0	OUT_WIDTH  Width  The value add 1 equal the actual output image width

#### 5.2.4.29. Output Address High 4bits Register

Offset: 0xEC			Register Name: MP_OUTH4ADD_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	



31:20	/	/	/
19:16	R/W	0	OUTCH2_H4ADD  Output channel 2  High 4bits address in bits
15:12	/	/	/
11:8	R/W	0	OUTCH1_H4ADD  Output channel 1  High 4bits address in bits
7:4	/	/	/
3:0	R/W	0	OUTCH0_H4ADD  Output channel 0  High 4bits address in bits

**5.2.4.30. Output Address Low 32bits Register**

Offset:		Register Name: MP_OUTL32ADD_REG	
Out channel 0:0xF0			
Out channel 1:0xF4			
Out channel 2:0xF8			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0	OUT_L32ADD  Output channel

			Low 32bits address in bits
--	--	--	----------------------------

#### 5.2.4.31. Output Line Width Register

Offset:			Register Name: MP_OUTLINEWIDTH_REG
Out channel 0:0x100			
Out channel 1:0x104			
Out channel 2:0x108			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0	OUT_LINEWIDTH  Output channel  Line width in bits

#### 5.2.4.32. Output Alpha Control Register

Offset: 0x120			Register Name: MP_OUTALPHACTL_REG
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	IMG_ALPHA  Output image area alpha value, the image area include A0,A1 and overlapping area A2.
23:16	R/W	0	NONIMG_ALPHA  Output non-image area alpha value, the non-image area means the pure fill color area.
15:8	/	/	/

7:6	R/W	0	<p>A2ALPHACTL</p> <p>A2 area alpha value control</p> <p>0: using A0 self pixel alpha (A0pA)</p> <p>1: using A1 self pixel alpha (A1pA)</p> <p>2: Mixed alpha A (<math>A0pA + A1pA * (1 - A0pA)</math>)</p> <p>3: using the Output image area alpha value (bit31:24)</p>
5:4	R/W	0	<p>A3ALPHACTL</p> <p>A3 area alpha value control</p> <p>0: 0xff</p> <p>1: using the Output non-image area alpha value (bit23:16)</p> <p>2: Mixed alpha A</p> <p>Other: reserved</p>
3:2	R/W	0	<p>A1ALPHACTL</p> <p>A1 area alpha value control</p> <p>0: using A1 self pixel alpha</p> <p>1: using the Output image area alpha value (bit31:24)</p> <p>2: Mixed alpha A</p> <p>Other: reserved</p>
1:0	R/W	0	<p>A0ALPHACTL</p>

			<p>A0 area alpha value control</p> <p>0: using A0 self pixel alpha</p> <p>1: using the Output image area alpha value (bit31:24)</p> <p>2: Mixed alpha A</p> <p>Other: reserved</p>
--	--	--	--

Description:

There is some area in output memory block:

**The alpha / color key module is enabled:**

Only the high priority image area is called A0

Only the low priority image area is called A1

The high priority and low priority mixed image area is called A2

The other area is called A3

And the A0,A1,A2 is called image area, the A3 is called non-image area.

**The alpha / color key module is disabled:**

Only the ROP output image area is called A0, A0 is called image area.

The other area is called A3, A3 is called non-image area.

**Mixed alpha A:** reference Alpha / Color key description

Note: the register setting is only valid in ARGB or AYUV mode.

**5.2.4.33. MB Control Register**

Offset:			Register Name: MP_MBCTL_REG
iDMA0:0X130			
iDMA1:0X134			
iDMA2:0X138			
iDMA3:0X13C			
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0	Y_OFFSET  The y offset of the top-left point in the whole image
15:0	R/W	0	X_OFFSET  The x offset of the top-left point in the whole image

**5.2.4.34. CSC0/1 Y/G coefficient register**

Offset:			Register Name: MP_ICSCYGCOEF_REG
G/Y component: 0x180			
R/U component: 0x184			
B/V component: 0x188			
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/

28:16	R/W	0x4a7	CSC1_YGCOEF
		0x1e6f	the Y/G coefficient for CSC1
		0x1cbf	the value equals to coefficient*2 <sup>10</sup>
15:13	/	/	/
12:00	R/W	0x4a7	CSC0_YGCOEF
		0x1e6f	the Y/G coefficient for CSC0
		0x1cbf	the value equals to coefficient*2 <sup>10</sup>

#### 5.2.4.35. CSC0/1 Y/G constant register

Offset: 0x18C			Register Name: MP_ICSCYGCONS_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	0x877	CSC1_YGCONS
			the Y/G constant for CSC1
			the value equals to coefficient*2 <sup>4</sup>
15:14	/	/	/
13:00	R/W	0x877	CSC0_YGCONS
			the Y/G constant for CSC0
			the value equals to coefficient*2 <sup>4</sup>

#### 5.2.4.36. CSC0/1 U/R coefficient register

Offset:	Register Name: MP_ICSCURCOEF_REG
---------	----------------------------------

G/Y component: 0x190			
R/U component: 0x194			
B/V component: 0x198			
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x4a7	CSC1_URCOEF
		0x00	the U/R coefficient for CSC1
		0x662	the value equals to coefficient*2 <sup>10</sup>
15:13	/	/	/
12:00	R/W	0x4a7	CSC0_URCOEF
		0x00	the U/R coefficient for CSC0
		0x662	the value equals to coefficient*2 <sup>10</sup>

#### 5.2.4.37. CSC0/1 U/R constant register

Offset: 0x19C			Register Name: MP_ICSCURCONS_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	0x3211	CSC1_URCONS
			the U/R constant for CSC1
			the value equals to coefficient*2 <sup>4</sup>
15:14	/	/	/

13:00	R/W	0x3211	CSC0_URCONS  the U/R constant for CSC0  the value equals to coefficient*2 <sup>4</sup>
-------	-----	--------	--

#### 5.2.4.38. CSC0/1 V/B coefficient register

Offset:			Register Name: MP_ICSCVBCOEF_REG
G/Y component: 0x1A0			
R/U component: 0x1A4			
B/V component: 0x1A8			
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x4a7	CSC1_VBCOEF
		0x812	the V/B coefficient for CSC1
		0x00	the value equals to coefficient*2 <sup>10</sup>
15:13	/	/	/
12:00	R/W	0x4a7	CSC0_VBCOEF
		0x812	the V/B coefficient for CSC0
		0x00	the value equals to coefficient*2 <sup>10</sup>

#### 5.2.4.39. CSC0/1 V/B constant register

Offset: 0x1AC			Register Name: MP_ICSCVBCONS_REG
Bit	Read/Write	Default/Hex	Description



	e	ex	
31:30	/	/	/
29:16	R/W	0x2eb1	CSC1_VBCONS  the V/B constant for CSC1  the value equals to coefficient*2 <sup>4</sup>
15:14	/	/	/
13:00	R/W	0x2eb1	CSC0_VBCONS  the V/B constant for CSC0  the value equals to coefficient*2 <sup>4</sup>

#### 5.2.4.40. CSC2 Y/G coefficient register

Offset:		Register Name: MP_OCSCYGCOEF_REG	
G/Y component: 0x1C0			
R/U component: 0x1C4			
B/V component: 0x1C8			
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:00	R/W		CSC2_YGCOEF  the Y/G coefficient  the value equals to coefficient*2 <sup>10</sup>

**5.2.4.41. CSC2 Y/G constant register**

Offset: 0x1CC			Register Name: MP_OCSCYGCONS_REG
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:00	R/W		CSC2_YGCONS the Y/G constant the value equals to coefficient*2 <sup>4</sup>

**5.2.4.42. CSC2 U/R coefficient register**

Offset: G/Y component: 0x1D0 R/U component: 0x1D4 B/V component: 0x1D8			Register Name: MP_OCSCURCOEF_REG
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:00	R/W		CSC2_URCOEF the U/R coefficient the value equals to coefficient*2 <sup>10</sup>

**5.2.4.43. CSC2 U/R constant register**

Offset: 0x1DC			Register Name: MP_OCSCURCONS_REG
Bit	Read/Write	Default/Hex	Description

	e	ex	
31:14	/	/	/
13:00	R/W		CSC2_URCONS  the U/R constant  the value equals to coefficient*2 <sup>4</sup>

#### 5.2.4.44. CSC2 V/B coefficient register

Offset:  G/Y component: 0x1E0  R/U component: 0x1E4  B/V component: 0x1E8			Register Name: MP_OCSCVBCOEF_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:13	/	/	/
12:00	R/W		CSC2_VBCOEF  the V/B coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 5.2.4.45. CSC2 V/B constant register

Offset: 0x1EC			Register Name: MP_OCSCVBCONS_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:14	/	/	/
13:00	R/W		CSC2_VBCONS

			the V/B constant  the value equals to coefficient*2 <sup>4</sup>
--	--	--	--

#### 5.2.4.46. Scaling horizontal filtering coefficient RAM block

Offset:  0x200 – 0x27C			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	Horizontal tap3 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0	Horizontal tap2 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:08	R/W	0	Horizontal tap1 coefficient  The value equals to coefficient*2 <sup>6</sup>
07:00	R/W	0	Horizontal tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>

**5.2.4.47. Scaling vertical filtering coefficient RAM block**

Offset:			
0x280 – 0x2FC			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	Vertical tap3 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0	Vertical tap2 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:08	R/W	0	Vertical tap1 coefficient  The value equals to coefficient*2 <sup>6</sup>
07:00	R/W	0	Vertical tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>

**5.2.4.48. Palette table**

Offset:			
0x400-0x7FF			
Bit	Read/Write	Default/Hex	Description

	e	ex	
31:24	R/W	UDF	Alpha value
23:16	R/W	UDF	Red value
15:08	R/W	UDF	Green value
07:00	R/W	UDF	Blue value

#### 5.2.4.49. Command queue control register

Offset: 0x140		Register Name: CMDQUECTL_REG	
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:9	/	/	/
8	R/W	0	FINISHIRQ_EN  Mission finish IRQ enable control  0:disable  1:enable
7:2	/	/	/
1	R/W	0	START_CTL  Start control  If the bit is set, the module will start a operation sets and stop auto.  The operation sets is stored in external memory.
0	R/W	0	EN  Command queue function enable control

			0:disable 1:enable
--	--	--	-----------------------

#### 5.2.4.50. Command queue status register

Offset: 0x144			Register Name: CMDQUESTS_REG
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R	0	BUSY_FLAG  Module working status  0:idle 1:running
11:9	/	/	/
8	R/W	0	FINISHIRQ_FLAG  Mission finish IRQ  It will be set when 1 frame operation accomplished, and cleared by writing 1.
7:0	/	/	/

#### 5.2.4.51. Command queue storage start address register

Offset:  0x148			Register Name: CMDQUEADD_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0	STARTADD

			Command queue start address in bytes
--	--	--	--------------------------------------

### 5.2.4.52. Input data pixel sequence table

Note: x means no care

1-bpp mode

PS=xx00

Bit

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

P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16
P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx01

Bit

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

P24	P25	P26	P27	P28	P29	P30	P31	P16	P17	P18	P19	P20	P21	P22	P23
P08	P09	P10	P11	P12	P13	P14	P15	P00	P01	P02	P03	P04	P05	P06	P07

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx10

Bit

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

P07	P06	P05	P04	P03	P02	P01	P00	P15	P14	P13	P12	P11	P10	P09	P08
P23	P22	P21	P20	P19	P18	P17	P16	P31	P30	P29	P28	P27	P26	P25	P24



15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx11

Bit

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

P00	P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	P15
P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

2-bpp mode

PS=xx00

Bit

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

P15	P14	P13	P12	P11	P10	P09	P08
P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx01

Bit

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

P12	P13	P14	P15	P08	P09	P10	P11
P04	P05	P06	P07	P00	P01	P02	P03

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx10

Bit

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

P03	P02	P01	P00	P07	P06	P05	P04
P11	P10	P09	P08	P15	P14	P13	P12

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx11

Bit

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

P00	P01	P02	P03	P04	P05	P06	P07
P08	P09	P10	P11	P12	P13	P14	P15

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

4-bpp mode

PS=xx00

Bit

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

P07	P06	P05	P04
P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx01

Bit

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

P06	P07	P04	P05
P02	P03	P00	P01

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx10

Bit

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

P01					P00					P03					P02				
P05					P04					P07					P06				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx11

Bit

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

P00					P01					P02					P03				
P04					P05					P06					P07				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

8-bpp mode

PS=xx00 / xx11

Bit

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

P3										P2									
P1										P0									

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx01 / xx10

Bit

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

P0															P1														
----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	----	--	--	--	--	--	--	--	--	--	--	--	--	--	--

P2										P3					
----	--	--	--	--	--	--	--	--	--	----	--	--	--	--	--

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

16-bpp @ A4R4G4B4 mode

PS=0x00

Bit

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

A1					R1					G1					B1				
A0					R0					G0					B0				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x01

Bit

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

A0					R0					G0					B0				
A1					R1					G1					B1				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x10

Bit

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

B1					G1					R1					A1				
B0					G0					R0					A0				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x11

Bit

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

B0					G0					R0					A0				
B1					G1					R1					A1				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=1xxx, the R component is swapped with B component

16-bpp @ A1R5G5B5 mode

PS=0x00

Bit

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

A1		R1					G1					B1				
A0		R0					G0					B0				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x01

Bit

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

A0		R0					G0					B0				
A1		R1					G1					B1				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x10

Bit

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

B1	G1	R1	A1
B0	G0	R0	A0

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x11

Bit

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

B0	G0	R0	A0
B1	G1	R1	A1

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=1xxx, the R component is swapped with B component

16-bpp @ R5G6B5 mode

PS=0x00

Bit

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

R1	G1	B1
R0	G0	B0

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=0x01

Bit

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

R0	G0	B0
----	----	----

R1									G1					B1			
----	--	--	--	--	--	--	--	--	----	--	--	--	--	----	--	--	--

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=1xxx, the R component is swapped with B component

16-bpp @ interleaved YUV422 mode

PS=xx00 / xx11

Bit

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

V0										Y1					
U0										Y0					

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx01 / xx10

Bit

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

Y1										V0					
Y0										U0					

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

16-bpp @ U8V8 mode

PS=xxxx

Bit

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

V1	U1
V0	U0

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

32-bpp ARGB or AYUV mode

PS=xx00 / xx01

Bit

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

A	R (Y)
G (U)	B (V)

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=xx10 / xx11

Bit

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

B (V)	G (U)
R (Y)	A

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=1xxx, the R component is swapped with B component

#### 5.2.4.53. Output data pixel sequence

32bpp – A8R8G8B8 or interleaved AYUV8888



16bpp – A4R4G4B4

16bpp – A1R5G5B5

16bpp – R5G6B5

16bpp – interleaved YUV422

Planar YUV422 (UV combined)

8bpp – MONO

4bpp – MONO

2bpp – MONO

1bpp – MONO

Planar YUV420 (UV combined)

Planar YUV411 (UV combined)

The above 13 kinds of output format is same as respective input format PS.

Planar YUV422

Planar YUV420

Planar YUV411

The above 3 kinds of output format is same as input 8bpp format PS.

## Chapter 6 Image

This chapter introduces the image features of A80.

## 6.1. CSI

### 6.1.1. Overview

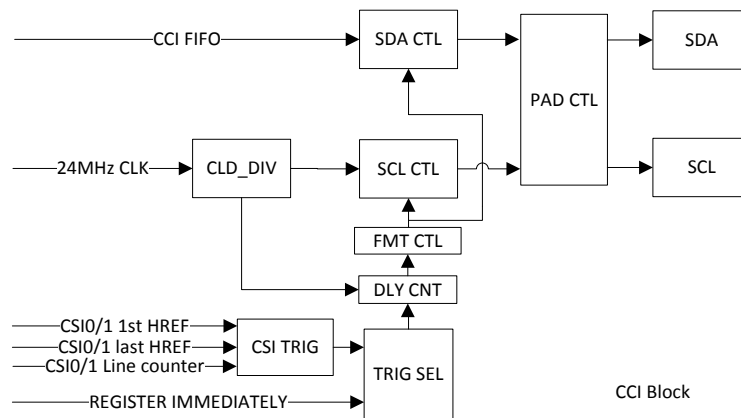
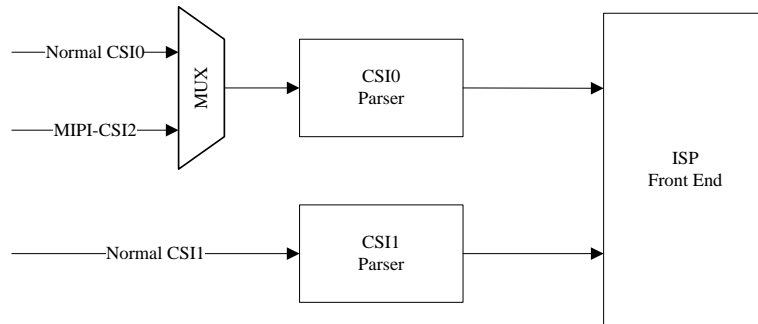
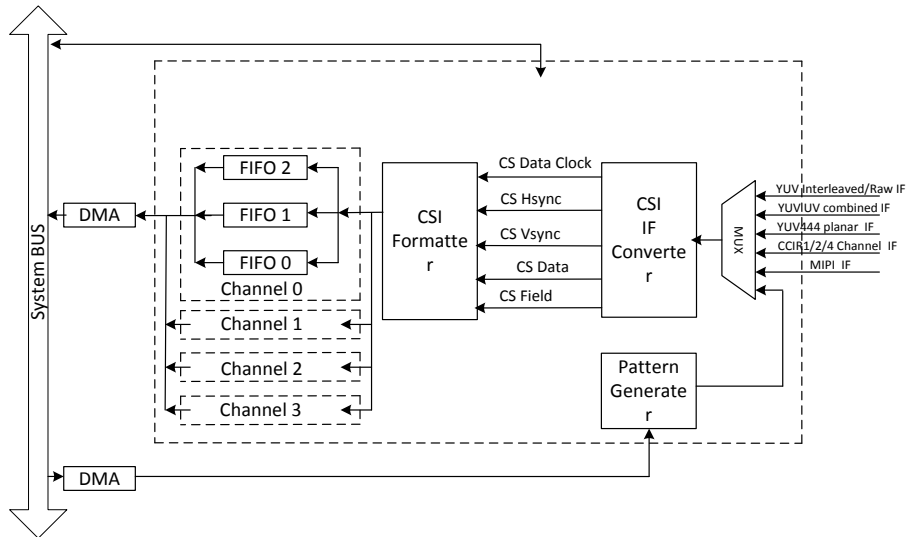
CSI:

- Support 8/10/12bit yuv422 and bayer raw CMOS sensor
- Support CCIR656 protocol for NTSC and PAL
- Support multi-channel ITU-R BT.656 time-multiplexed format
- Maximum still capture resolution to 16M (4608x3456) with yuv422 or yuv420
- Maximum video capture resolution to 1080p@60fps
- Support dual channel buffer output for display and encoder
- Parsing interleaved YCbCr 422 data into planar or MB Y, Cb, Cr output to memory
- Parsing YCbCr 420 data into planar or MB Y, Cb, Cr output to memory
- Parsing interlaced data into field or frame output to memory
- Received data double buffer support
- Crop and scale support
- H/V-flip and rotation support
- Frame rate counter statistic

CCI

- Compatible with i2c transmission in 7 bit slave ID + 1 bit R/W
- Automatic transmission
- 0/8/16/32 bit register address supported
- 8/16/32 bit data supported
- 64bytes-FIFO input CCI data supported
- Synchronized with CSI signal and delay trigger supported
- Repeated transmission with sync signal supported

6.1.2. Block Diagram



### 6.1.3. CSI Description

#### 6.1.3.1. CSI FIFO Distribution

Interface	YUYV422  Interleaved  /RAW		YUV422  UV  Combined	YUV444  Planar	YUV444  Planar to  YUV422  UV Combined	
Input format	YUV422		Raw	Raw	Raw	Raw
Output format	Planar	UV combined  / MB	Raw  /RGB  /PRGB	Raw	Raw	Raw
CH0_FIFO0	Y  pixel data	Y  pixel data	All  pixels data	Y  pixel data	Y  pixel data	Y  pixel data
CH0_FIFO1	Cb (U)  pixel data	Cb (U)  Cr (V)  pixel data	-	-	-	-
CH0_FIFO2	Cr (V)  pixel data	-	-	-	-	-
CH1_FIFO0	-	-	-	Cb (U)  Cr (V)  pixel data	Cb (U)  pixel data	Cb (U)  Cr (V)  pixel data

CH2_FIFO0	-	-	-	-	Cr(V) pixel data	-
-----------	---	---	---	---	---------------------	---

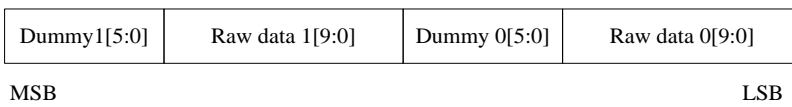
Interface	BT656 Interface		Channels		
Input format	YUV422				
Output format	Planar	UV  combined/  MB			
CH0_FIFO0	Y	Y	1	2	4
CH0_FIFO1	Cb (U)	CbCr  (UV)			
CH0_FIFO2	Cr (V)	-			
CH1_FIFO0	Y	Y	-		
CH1_FIFO1	Cb (U)	CbCr  (UV)			
CH1_FIFO2	Cr (V)	-			
CH2_FIFO0	Y	Y			
CH2_FIFO1	Cb (U)	CbCr  (UV)			
CH2_FIFO2	Cr (V)	-			
CH3_FIFO0	Y	Y			
CH3_FIFO1	Cb (U)	CbCr  (UV)			
CH3_FIFO2	Cr (V)	-			

Interface	MIPI Interface			Channels			
	Input format	YUV422/YUV420					
Output format	Planar	UV  combined/  MB	Pass-Through  /Padding				
CH0_FIFO0	Y	Y	All pixels data	1	2	3	4
CH0_FIFO1	Cb (U)	CbCr  (UV)	-				
CH0_FIFO2	Cr (V)	-	-				
CH1_FIFO0	Y	Y	All pixels data	-			
CH1_FIFO1	Cb (U)	CbCr  (UV)	-				
CH1_FIFO2	Cr (V)	-	-				
CH2_FIFO0	Y	Y	All pixels data		-		
CH2_FIFO1	Cb (U)	CbCr  (UV)	-				
CH2_FIFO2	Cr (V)	-	-				
CH3_FIFO0	Y	Y	All pixels data			-	
CH3_FIFO1	Cb (U)	CbCr	-				

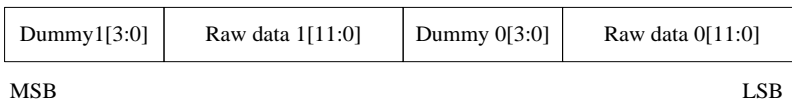
		(UV)					
CH3_FIFO2	Cr (V)	-	-				

**6.1.3.2. Pixel format Arrangement**

RAW-10:

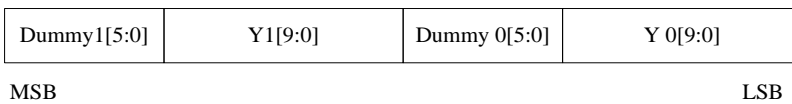


RAW-12:

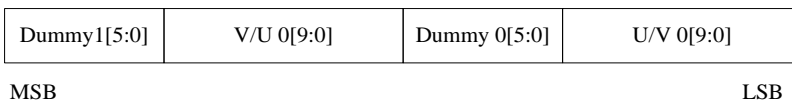


YUV-10:

Y:



UV Combined:



RGB888:



B1[7:0]	R0[7:0]	G0[7:0]	B0[7:0]
G2[7:0]	B2[7:0]	R1[7:0]	G1[7:0]
R3[7:0]	G3[7:0]	B3[7:0]	R2[7:0]

MSB

**PRGB888:**

PAD[7:0]	R0[7:0]	G0[7:0]	B0[7:0]
----------	---------	---------	---------

MSB

LSB

**RGB565:**

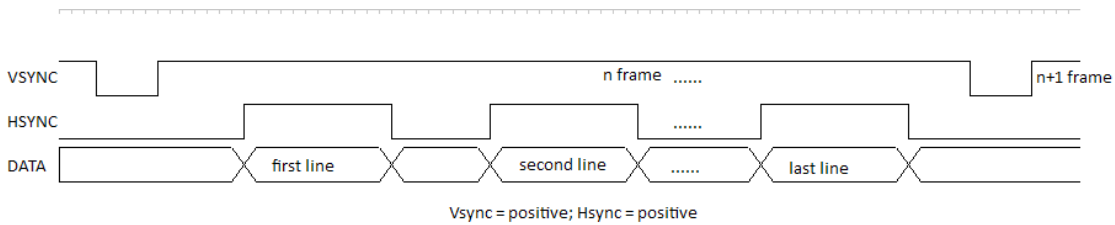
R1[4:0]	G1[5:0]	B1[4:0]	R0[4:0]	G0[5:0]	B0[4:0]
---------	---------	---------	---------	---------	---------

MSB

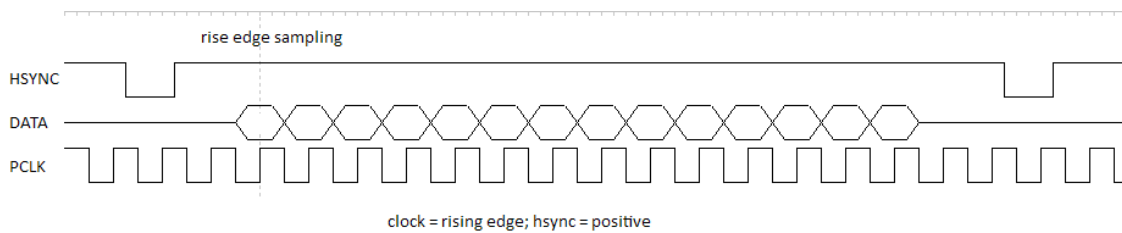
LSB

**6.1.3.3. Timing**

**CSI timing**



**Vref= positive; Href= positive**



**horizontal size setting and pixel clock timing(Href= positive)**

**CCIR656 Header Code**

**CCIR656 Header Data Bit Definition**

Data Bit	First Word(0xFF)	Second Word(0x00)	Third Word(0x00)	Fourth Word
CS D[9] (MSB)	1	0	0	1
CS D[8]	1	0	0	F
CS D[7]	1	0	0	V
CS D[6]	1	0	0	H
CS D[5]	1	0	0	P3
CS D[4]	1	0	0	P2
CS D[3]	1	0	0	P1
CS D[2]	1	0	0	P0
CS D[1]	x	x	x	x
CS D[0]	x	x	x	x

For compatibility with an 8-bit interface, CS D[1] and CS D[0] are not defined.

Decode	F	V	H	P3	P2	P1	P0
Field 1 start of active video (SAV)	0	0	0	0	0	0	0
Field 1 end of active video (EAV)	0	0	1	1	1	0	1
Field 1 SAV (digital blanking)	0	1	0	1	0	1	1
Field 1 EAV (digital blanking)	0	1	1	0	1	1	0
Field 2 SAV	1	0	0	0	1	1	1
Field 2 EAV	1	0	1	1	0	1	0
Field 2 SAV (digital blanking)	1	1	0	1	1	0	0
Field 2 EAV (digital blanking)	1	1	1	0	0	0	1

Multi-Channel:

Condition			656FVH Value			SAV-EAV Code						
Field	V-time	H-time	F	V	H	First	Second	Third	Fourth			
									Ch1	Ch2	Ch3	Ch4
EVEN	BLANK	EAV	1	1	1	0xFF	0x00	0x00	0xF0	0xF1	0xF2	0xF3
EVEN	BLANK	SAV	1	1	0	0xFF	0x00	0x00	0xE0	0xE1	0xE2	0xE3
EVEN	BLANK	EAV	1	0	1	0xFF	0x00	0x00	0xD0	0xD1	0xD2	0xD3
EVEN	BLANK	SAV	1	0	0	0xFF	0x00	0x00	0xC0	0xC1	0xC2	0xC3
ODD	BLANK	EAV	0	1	1	0xFF	0x00	0x00	0xB0	0xB1	0xB2	0xB3
ODD	BLANK	SAV	0	1	0	0xFF	0x00	0x00	0xA0	0xA1	0xA2	0xA3
ODD	ACTIVE	EAV	0	0	1	0xFF	0x00	0x00	0x90	0x91	0x92	0x93
ODD	ACTIVE	SAV	0	0	0	0xFF	0x00	0x00	0x80	0x81	0x82	0x83

#### 6.1.3.4. Offset / scale / flip function

Interface will do these three functions in sequence.

##### 6.1.3.4.1. Offset definition

Offset in horizontal and vertical can be added when receiving image. Unit is pixel.

For YUV422 format, pixel unit is a YU/YV combination.

For YUV420 format, pixel unit is a YU/YV combination in YC line, and only a Y in Y line.

For Bayer and RAW format, pixel unit is a R/G/B single component.

For RGB565, pixel unit is 2 bytes of RGB565 package.

For RGB888, pixel unit is 3 bytes of RGB combination.

##### 6.1.3.4.2. Flip definition

Both horizontal and vertical flip are supported at the same time. This function is implemented **in the process of each FIFO writing data to memory, only flipping the data of separate FIFO, not changing component to FIFO distribution.**

If horizontal flip is enabled, one or more pixels will be taken as a unit:

For YUV format, a unit of  $Y_0U_0Y_1V_1$  will parser and flip the Y component in one channel, and UV will be treated as a whole. In planar output mode, U and V will be flipped separately. In UV combined output mode, UV will be flipped as a whole. So, a sequence of  $Y_1U_0Y_0V_1$  will be.

For Bayer\_raw format, situation is much like. A GR/BG sequence will be changed to BG/RG. A unit of square has four pixels.

For RGB565/RGB888, one unit of two/three bytes of component will be flipped with original sequence.

#### 6.1.3.5. Camera Communication Interface

The CCI module support master mode i2c-compatible single read and write access to camera and related devices.

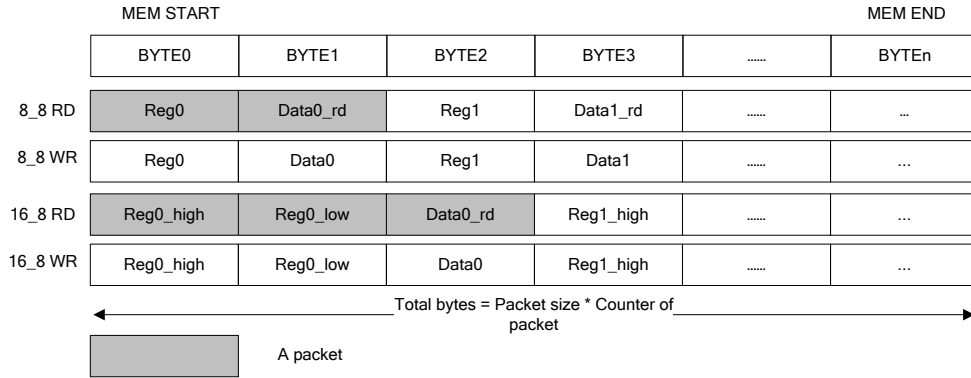
It reads a series of packet from FIFO (accessed by registers) and transmit with the format defined in specific register(or packet data).

In compact mode, format register define the slave ID, R/W flag, register address width(0/8/16/32...bit), data width(8/16/32...bit) and access counter.

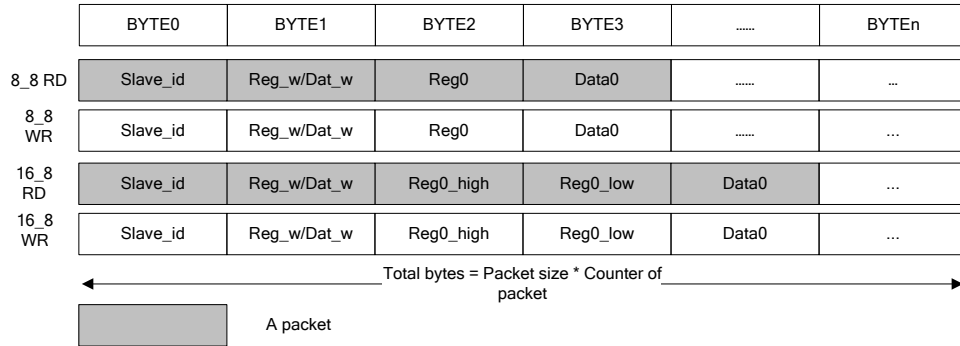
In complete mode, all data and format will be loaded from memory packet.

The access counter should be set to  $N(N > 0)$ , and it will read N packets from FIFO. The total bytes should not exceed 64 for FIFO input mode.

COMPACT MODE

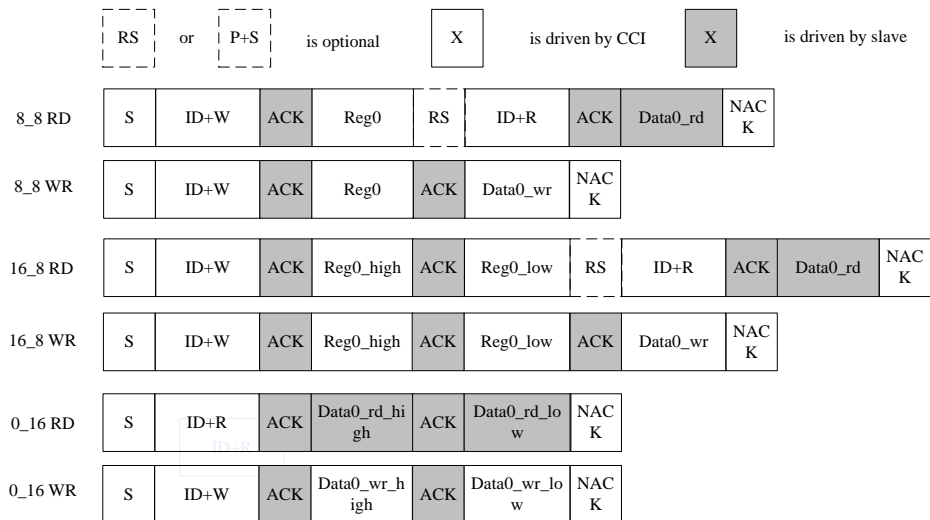


COMPLETE MODE

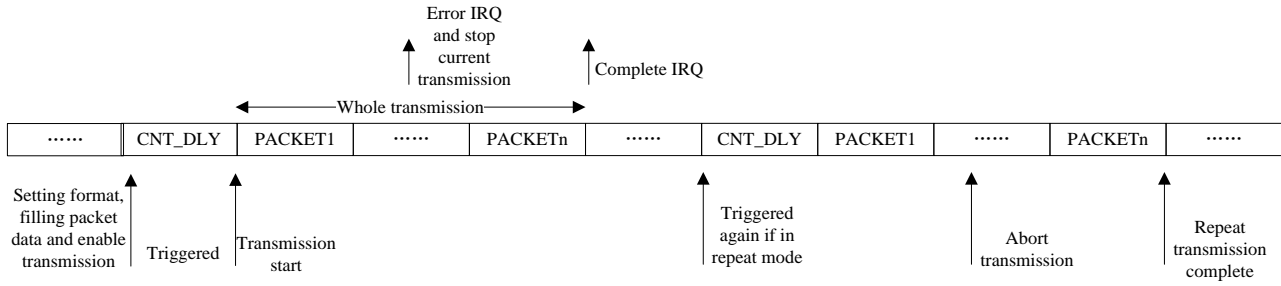


A packet is several bytes filled with register address and data (if in complete mode, slave id and width should be filled too) as the i2c access sequence defined. That is, the low address byte will be transmitted/received first. Bytes will be sent in write access, while some address will be written back with the data received in read access.

Single Access protocol supported by CCI



After set the execution bit, the module will do the transmission automatically and return the result - success or fail. If any access fail, the whole transmission will be stopped and returns the number when it fail in the access counter.



#### 6.1.4. CSI Register list

Module Name	Base Address
CSI0	0x03800000
CSI1	0x03900000

Register Name	Offset	Register name
CSI_EN_REG	0X0000	CSI Enable register
CSI_IF_CFG_REG	0X0004	CSI Interface Configuration Register
CSI_CAP_REG	0X0008	CSI Capture Register
CSI_SYNC_CNT_REG	0X000C	CSI Synchronization Counter Register
CSI_FIFO_THRS_REG	0X0010	CSI FIFO Threshold Register
CSI_BT656_HEAD_CFG_REG	0X0014	CSI BT656 Header Configuration Register
CSI_PTN_LEN_REG	0X0030	CSI Pattern Generation Length register
CSI_PTN_ADDR_REG	0X0034	CSI Pattern Generation Address register
CSI_VER_REG	0X003C	CSI Version Register
CSI_C0_CFG_REG	0X0044	CSI Channel_0 configuration register
CSI_C0_SCALE_REG	0X004C	CSI Channel_0 scale register
CSI_C0_F0_BUFA_REG	0X0050	CSI Channel_0 FIFO 0 output buffer-A address register
CSI_C0_F1_BUFA_REG	0X0058	CSI Channel_0 FIFO 1 output buffer-A address register
CSI_C0_F2_BUFA_REG	0X0060	CSI Channel_0 FIFO 2 output buffer-A address register
CSI_C0_CAP_STA_REG	0X006C	CSI Channel_0 status register
CSI_C0_INT_EN_REG	0X0070	CSI Channel_0 interrupt enable register
CSI_C0_INT_STA_REG	0X0074	CSI Channel_0 interrupt status register
CSI_C0_HSIZE_REG	0X0080	CSI Channel_0 horizontal size register
CSI_C0_VSIZE_REG	0X0084	CSI Channel_0 vertical size register
CSI_C0_BUF_LEN_REG	0X0088	CSI Channel_0 line buffer length register
CSI_C0_FLIP_SIZE_REG	0X008C	CSI Channel_0 flip size register

CSI_C0_FRM_CLK_CNT_REG	0X0090	CSI Channel_0 frame clock counter register
CSI_C0_ACC_ITNL_CLK_CNT_REG	0X0094	CSI Channel_0 accumulated and internal clock counter register
CSI_C0_FIFO_STAT_REG	0X0098	CSI Channel_0 FIFO Statistic Register
CSI_C0_PCLK_STAT_REG	0X009C	CSI Channel_0 PCLK Statistic Register
CSI_C1_CFG_REG	0X0144	CSI Channel_1 configuration register
CSI_C1_SCALE_REG	0X014C	CSI Channel_1 scale register
CSI_C1_F0_BUFA_REG	0X0150	CSI Channel_1 FIFO 0 output buffer-A address register
CSI_C1_F1_BUFA_REG	0X0158	CSI Channel_1 FIFO 1 output buffer-A address register
CSI_C1_F2_BUFA_REG	0X0160	CSI Channel_1 FIFO 2 output buffer-A address register
CSI_C1_CAP_STA_REG	0X016C	CSI Channel_1 status register
CSI_C1_INT_EN_REG	0X0170	CSI Channel_1 interrupt enable register
CSI_C1_INT_STA_REG	0X0174	CSI Channel_1 interrupt status register
CSI_C1_HSIZE_REG	0X0180	CSI Channel_1 horizontal size register
CSI_C1_VSIZE_REG	0X0184	CSI Channel_1 vertical size register
CSI_C1_BUF_LEN_REG	0X0188	CSI Channel_1 line buffer length register
CSI_C1_FLIP_SIZE_REG	0X018C	CSI Channel_1 flip size register
CSI_C1_FRM_CLK_CNT_REG	0X0190	CSI Channel_1 frame clock counter register
CSI_C1_ACC_ITNL_CLK_CNT_REG	0X0194	CSI Channel_1 accumulated and internal clock counter register
CSI_C1_FIFO_STAT_REG	0X0198	CSI Channel_1 FIFO Statistic Register
CSI_C1_PCLK_STAT_REG	0X019C	CSI Channel_1 PCLK Statistic Register
CSI_C2_CFG_REG	0X0244	CSI Channel_2 configuration register
CSI_C2_SCALE_REG	0X024C	CSI Channel_2 scale register
CSI_C2_F0_BUFA_REG	0X0250	CSI Channel_2 FIFO 0 output buffer-A address register
CSI_C2_F1_BUFA_REG	0X0258	CSI Channel_2 FIFO 1 output buffer-A address register
CSI_C2_F2_BUFA_REG	0X0260	CSI Channel_2 FIFO 2 output buffer-A address register
CSI_C2_CAP_STA_REG	0X26C	CSI Channel_2 status register



CSI_C2_INT_EN_REG	0X0270	CSI Channel_2 interrupt enable register
CSI_C2_INT_STA_REG	0X0274	CSI Channel_2 interrupt status register
CSI_C2_HSIZE_REG	0X0280	CSI Channel_2 horizontal size register
CSI_C2_VSIZE_REG	0X0284	CSI Channel_2 vertical size register
CSI_C2_BUF_LEN_REG	0X0288	CSI Channel_2 line buffer length register
CSI_C2_FLIP_SIZE_REG	0X028C	CSI Channel_2 flip size register
CSI_C2_FRM_CLK_CNT_REG	0X0290	CSI Channel_2 frame clock counter register
CSI_C2_ACC_ITNL_CLK_CNT_REG	0X0294	CSI Channel_2 accumulated and internal clock counter register
CSI_C2_FIFO_STAT_REG	0X0298	CSI Channel_2 FIFO Statistic Register
CSI_C2_PCLK_STAT_REG	0X029C	CSI Channel_2 PCLK Statistic Register
CSI_C3_CFG_REG	0X0344	CSI Channel_3 configuration register
CSI_C3_SCALE_REG	0X034C	CSI Channel_3 scale register
CSI_C3_F0_BUFA_REG	0X0350	CSI Channel_3 FIFO 0 output buffer-A address register
CSI_C3_F1_BUFA_REG	0X0358	CSI Channel_3 FIFO 1 output buffer-A address register
CSI_C3_F2_BUFA_REG	0X0360	CSI Channel_3 FIFO 2 output buffer-A address register
CSI_C3_CAP_STA_REG	0X036C	CSI Channel_3 status register
CSI_C3_INT_EN_REG	0X0370	CSI Channel_3 interrupt enable register
CSI_C3_INT_STA_REG	0X0374	CSI Channel_3 interrupt status register
CSI_C3_HSIZE_REG	0X0380	CSI Channel_3 horizontal size register
CSI_C3_VSIZE_REG	0X0384	CSI Channel_3 vertical size register
CSI_C3_BUF_LEN_REG	0X0388	CSI Channel_3 line buffer length register
CSI_C3_FLIP_SIZE_REG	0X038C	CSI Channel_3 flip size register
CSI_C3_FRM_CLK_CNT_REG	0X0390	CSI Channel_3 frame clock counter register
CSI_C3_ACC_ITNL_CLK_CNT_REG	0X0394	CSI Channel_3 accumulated and internal clock counter register
CSI_C3_FIFO_STAT_REG	0X0398	CSI Channel_3 FIFO Statistic Register
CSI_C3_PCLK_STAT_REG	0X039C	CSI Channel_3 PCLK Statistic Register

CCI_CTRL	0x3000	CCI control register
CCI_CFG	0x3004	CCI transmission config register
CCI_FMT	0x3008	CCI packet format register
CCI_BUS_CTRL	0x300C	CCI bus control register
CCI_INT_CTRL	0x3014	CCI interrupt control register
CCI_LC_TRIG	0x3018	CCI line counter trigger register
CCI_FIFO_ACC	0x3100	CCI FIFO access register
CCI_RSV_REG	0x3200	CCI reserved register

## 6.1.5. CSI Register Description

### 6.1.5.1. CSI Enable Register

Offset Address: 0x0000			Register Name: CSI_EN_REG
Bit	Read/ Write	Default/Hex	Description
31	/	/	/
30	R/W	0x0	VER_EN  CSI Version Register Read Enable:  0: Disable  1: Enable
29:24	/	/	/
23:16	R/W	0x00	PTN_CYCLE  Pattern generating cycle counter.  The pattern in dram will be generated in cycles of PTN_CYCLE+1.
15:9	/	/	/
8	R/W	0x0	SRAM_PWDN  0: SRAM in normal  1: SRAM in power down
7:3	/	/	/
4	R/W	0x0	PTN_START  CSI Pattern Generating Start

			<p>0: Finish</p> <p>other: Start</p> <p>Software writes this bit to“1” to start pattern generating from DRAM. When finished, the hardware will clear this bit to“0”automatically. Generating cycles depends on PTN_CYCLE.</p>
3	R/W	0	<p>CLK_CNT_SPL</p> <p>Sampling time for clock counter per frame</p> <p>0: Sampling clock counter every frame done</p> <p>1: Sampling clock counter every VSYNC</p>
2	R/W	0	<p>CLK_CNT</p> <p>Clock count per frame</p>
1	R/W	0	<p>PTN_GEN_EN</p> <p>Pattern Generation Enable</p>
0	R/W	0	<p>CSI_EN</p> <p>Enable</p> <p>0: Reset and disable the CSI module</p> <p>1: Enable the CSI module</p>

### 6.1.5.2. CSI Interface Configuration Register

Offset Address: 0x0004		Register Name: CSI_IF_CFG_REG	
Bit	Read/ Write	Default/Hex	Description

31:30	/	/	/
29	R/W	0	BUF_SHARE  Buffer share between csi0 and csi1 control:  0: Share disable  1: Share enable
28:24	/	/	/
23	R/W	0	Reserved
22	/	/	/
21	R/W	0	SRC_TYPE  Source type  0: Progressed  1: Interlaced
20	R/W	0	FPS_DS  Fps down sample  0: no down sample  1: 1/2 fps, only receives the first frame every 2 frames
19	R/W	0	FIELD  For YUV HV timing, Field polarity  0: negative(field=0 indicate odd, field=1 indicate even )  1: positive(field=1 indicate odd, field=0 indicate even )  For BT656 timing, Field sequence

			<p>0: Normal sequence (field 0 first)</p> <p>1: Inverse sequence (field 1 first)</p>
18	R/W	1	<p>VREF_POL</p> <p>VREF polarity</p> <p>0: negative</p> <p>1: positive</p> <p>Not apply to CCIR656 interface.</p>
17	R/W	0	<p>HERF_POL</p> <p>HREF polarity</p> <p>0: negative</p> <p>1: positive</p> <p>Not apply to CCIR656 interface.</p>
16	R/W	1	<p>CLK_POL</p> <p>Data clock type</p> <p>0: active in falling edge</p> <p>1: active in rising edge</p>
15:12	/	/	/
11:10	R/W	0	<p>SEQ_8PLUS2</p> <p>When select IF_DATA_WIDTH to be 8+2bit, odd/even pixel byte at CSI-D[11:4] will be rearranged to D[11:2]+2'b0 at the actual CSI data bus according to these sequences:</p>

			00: 6'bx+D[9:8], D[7:0]  01: D[9:2], 6'bx+D[1:0]  10: D[7:0], D[9:8]+6'bx  11: D[7:0], 6'bx+D[9:8]
9:8	R/W	0	IF_DATA_WIDTH  00: 8 bit data bus  01: 10 bit data bus  10: 12 bit data bus  11: 8+2bit data bus
7	R/W	0	MIPI_IF  MIPI Interface Enable:  0: CSI  1: MIPI
6:5	/	/	/
4:0	R/W	0	CSI_IF  YUV:  00000: YUYV422 Interleaved or RAW (All data in one data bus)  00001: Reserved  00010: Reserved

			00011: Reserved
			CCIR656:
			00100: YUYV422 Interleaved or RAW (All data in one data bus)
			00101: Reserved
			00110: Reserved
			00111: Reserved
			01100: CCIR656 2 channels (All data interleaved in one data bus)
			01101: CCIR656 4 channels (All data interleaved in one data bus)
			Others: Reserved

### 6.1.5.3. CSI Capture Register

Offset: 0x0008			Register Name: CSI_CAP_REG
Bit	Read/ Write	Default/Hex	Description
31:30	/	/	/
29:26	R/W	0x00	CH3_CAP_MASK  VSYNC number masked before capture.



25	R/W	0	<p>CH3_VCAP_ON</p> <p>Video capture control: Capture the video image data stream on channel 3.</p> <p>0: Disable video capture</p> <p>If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is written to output FIFO.</p> <p>1: Enable video capture</p> <p>The CSI starts capturing image data at the start of the next frame.</p>
24	R/W	0	<p>CH3_SCAP_ON</p> <p>Still capture control: Capture a single still image frame on channel 3.</p> <p>0: Disable still capture.</p> <p>1: Enable still capture</p> <p>The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self-clearing and always reads as a 0.</p>
23:22	/	/	/
21:18	R/W	0x00	<p>CH2_CAP_MASK</p> <p>VSYNC number masked before capture.</p>
17	R/W	0	<p>CH2_VCAP_ON</p> <p>Video capture control: Capture the video image data stream on channel 2.</p> <p>0: Disable video capture</p> <p>If video capture is in progress, the CSI stops capturing image data at the end of</p>

			<p>the current frame, and all of the current frame data is written to output FIFO.</p> <p>1: Enable video capture</p> <p>The CSI starts capturing image data at the start of the next frame.</p>
16	R/W	0	<p>CH2_SCAP_ON</p> <p>Still capture control: Capture a single still image frame on channel 2.</p> <p>0: Disable still capture.</p> <p>1: Enable still capture</p> <p>The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self-clearing and always reads as a 0.</p>
15:14	/	/	/
13:10	R/W	0x00	<p>CH1_CAP_MASK</p> <p>VSYNC number masked before capture.</p>
09	R/W	0	<p>CH1_VCAP_ON</p> <p>Video capture control: Capture the video image data stream on channel 1.</p> <p>0: Disable video capture</p> <p>If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is written to output FIFO.</p> <p>1: Enable video capture</p> <p>The CSI starts capturing image data at the start of the next frame.</p>
08	R/W	0	<p>CH1_SCAP_ON</p>

			<p>Still capture control: Capture a single still image frame on channel 1.</p> <p>0: Disable still capture.</p> <p>1: Enable still capture</p> <p>The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self-clearing and always reads as a 0.</p>
07:06	/	/	/
05:02	R/W	0x00	<p>CHO_CAP_MASK</p> <p>VSYNC number masked before capture.</p>
01	R/W	0	<p>CHO_VCAP_ON</p> <p>Video capture control: Capture the video image data stream on channel 0.</p> <p>0: Disable video capture</p> <p>If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is written to output FIFO.</p> <p>1: Enable video capture</p> <p>The CSI starts capturing image data at the start of the next frame.</p>
00	R/W	0	<p>CHO_SCAP_ON</p> <p>Still capture control: Capture a single still image frame on channel 0.</p> <p>0: Disable still capture.</p> <p>1: Enable still capture</p> <p>The CSI module starts capturing image data at the start of the next frame. The</p>

			CSI module captures only one frame of image data. This bit is self-clearing and always reads as a 0.
--	--	--	--

#### 6.1.5.4. CSI Synchronization Counter Register

Offset Address: 0x000C			Register Name: CSI_SYNC_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	/	/	/
23:00	R	0	<p>SYNC_CNT</p> <p>The counter value between VSYNC of Csi0 channel 0 and VSYNC of Csi1 channel 0 , using 24MHz.</p>

#### 6.1.5.5. CSI FIFO Threshold Register

Offset Address: 0x0010			Register Name: CSI_FIFO_THRS_REG
Bit	Read/ Write	Default/Hex	Description
31:29	/	/	/
28:26	R/W	0x0	<p>FIFO_NEARLY_FULL_TH</p> <p>The threshold of FIFO being nearly full. Indicates that the ISP should stop writing. Only valid when ISP is enabled.</p> <p>0~7: The smaller the value, the flag of FIFO being nearly full is easier to reach.</p>
25:24	R/W	0x0	<p>PTN_GEN_CLK_DIV</p> <p>Packet generator clock divider</p>

23:16	R/W	0x0f	PTN_GEN_DLY  Clocks delayed before pattern generating start.
15:12	/	/	/
11:00	R/W	0x400	FIFO_THRS  When CSI0 FIFO occupied memory exceed the threshold, dram frequency can not change.

#### 6.1.5.6. CSI BT656 Header Configuration Register

Offset Address: 0x0014			Register Name: CSI_BT656_HEAD_CFG_REG
Bit	Read/ Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x3	CH3_ID  The low 4bit of BT656 header for channel 3  Only valid in BT656 multi-channel mode
23:20	/	/	/
19:16	R/W	0x2	CH2_ID  The low 4bit of BT656 header for channel 2  Only valid in BT656 multi-channel mode
15:12	/	/	/
11:08	R/W	0x1	CH1_ID  The low 4bit of BT656 header for channel 1  Only valid in BT656 multi-channel mode

07:04	/	/	/
03:00	R/W	0x0	<p>CH0_ID</p> <p>The low 4bit of BT656 header for channel 0</p> <p>Only valid in BT656 multi-channel mode</p>

#### 6.1.5.7. CSI Pattern Generation Length Register

Offset: 0x0030			Register Name: CSI_PTN_LEN_REG
Bit	Read/ Write	Default/Hex	Description
31:0	R/W	0x0	<p>PTN_LEN</p> <p>The pattern length in byte when generating pattern.</p>

#### 6.1.5.8. CSI Pattern Generation Address Register

Offset: 0x0034			Register Name: CSI_PTN_ADDR_REG
Bit	Read/ Write	Default/Hex	Description
31:0	R/W	0x0	<p>PTN_ADDR</p> <p>The pattern DRAM address when generating pattern.</p>

#### 6.1.5.9. CSI Version Register

Offset: 0x003C			Register Name: CSI_VER_REG
Bit	Read/ Write	Default/Hex	Description

31:0	R/W	/	VER  Version of hardware circuit. Only can be readable when version register read enable is on.
------	-----	---	---

#### 6.1.5.10. CSI Channel\_0 configuration register

Offset Address: 0X0044			Register Name: CSI_CO_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	PAD_VAL  Padding value when OUTPUT_FMT is prgb888  0x00~0xff
23:20	R/W	3	INPUT_FMT  Input data format  0000: RAW stream  0001: reserved  0010: reserved  0011: YUV422  0100: YUV420  Others: reserved
19:16	R/W	0	OUTPUT_FMT  Output data format

			<p>When the input format is set RAW stream</p> <p>0000: field-raw-8</p> <p>0001: field-raw-10</p> <p>0010: field-raw-12</p> <p>0011: reserved</p> <p>0100: field-rgb565</p> <p>0101: field-rgb888</p> <p>0110: field-prgb888</p> <p>0111: field-UV-combined</p> <p>1000: frame-raw-8</p> <p>1001: frame-raw-10</p> <p>1010: frame-raw-12</p> <p>1011: reserved</p> <p>1100: frame-rgb565</p> <p>1101: frame-rgb888</p> <p>1110: frame-prgb888</p> <p>1111: frame-UV-combined</p>
--	--	--	--



		<p>When the input format is set YUV422</p> <p>0000: field planar YCbCr 422</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p> <p>0011: frame planar YCbCr 422</p> <p>0100: field planar YCbCr 422 UV combined</p> <p>0101: field planar YCbCr 420 UV combined</p> <p>0110: frame planar YCbCr 420 UV combined</p> <p>0111: frame planar YCbCr 422 UV combined</p> <p>1000: field MB YCbCr 422</p> <p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: frame MB YCbCr 422</p> <p>1100: field planar YCbCr 422 10bit UV combined</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p>
--	--	--

			<p>When the input format is set YUV420</p> <p>0000: Reserved</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p> <p>0011: Reserved</p> <p>0100: Reserved</p> <p>0101: field planar YCbCr 420 UV combined</p> <p>0110: frame planar YCbCr 420 UV combined</p> <p>0111: Reserved</p> <p>1000: Reserved</p> <p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: Reserved</p> <p>1100: Reserved</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p>
--	--	--	--

			Others: reserved
15:14	/	/	/
13	R/W	0	<p>VFLIP_EN</p> <p>Vertical flip enable</p> <p>When enabled, the received data will be arranged in vertical flip.</p> <p>0:Disable</p> <p>1:Enable</p>
12	R/W	0	<p>HFLIP_EN</p> <p>Horizontal flip enable</p> <p>When enabled, the received data will be arranged in horizontal flip.</p> <p>0:Disable</p> <p>1:Enable</p>
11:10	R/W	0	<p>FIELD_SEL</p> <p>Field select</p> <p>00: capture field 1.</p> <p>01: capture field 2.</p> <p>10: capture either field.</p> <p>11: reserved</p>
09:08	R/W	2	<p>INPUT_SEQ</p> <p>Input data sequence, only valid for YUV422 and YUV420 input format.</p>

			<p>All data interleaved in one channel:</p> <p>00: YUYV</p> <p>01: YVYU</p> <p>10: UYVY</p> <p>11: VYUY</p> <p>Y and UV in separated channel:</p> <p>x0: UV</p> <p>x1: VU</p>
07:02	/	/	/
01:00			<p>MIN_SDR_WR_SIZE</p> <p>Minimum size of SDRAM block write</p> <p>0: 256 bytes (if HFLIP is enable, always select 256 bytes)</p> <p>1: 512 bytes</p> <p>2: 1k bytes</p> <p>3: 2k bytes</p>

#### 6.1.5.11. CSI Channel\_0 scale register

Offset Address: 0X004C	Register Name: CSI_CO_SCALE_REG
------------------------	---------------------------------

Bit	Read/Write	Default/Hex	Description
31:01	/	/	/
00	R/W	0	QUART_EN  When this bit is set to 1, input image will be subsample to quarter size. Support for all input formats.

#### 6.1.5.12. CSI Channel\_0 FIFO 0 output buffer-A address register

Offset Address: 0X0050		Register Name: CSI_CO_F0_BUFA_REG	
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	COF0_BUFA  FIFO 0 output buffer-A address

#### 6.1.5.13. CSI Channel\_0 FIFO 1 output buffer-A address register

Offset Address: 0X0058		Register Name: CSI_CO_F1_BUFA_REG	
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	COF1_BUFA  FIFO 1 output buffer-A address

#### 6.1.5.14. CSI Channel\_0 FIFO 2 output buffer-A address register

Offset Address: 0X0060		Register Name: CSI_CO_F2_BUFA_REG	
------------------------	--	-----------------------------------	--

Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	COF2_BUFA  FIFO 2 output buffer-A address

#### 6.1.5.15. CSI Channel\_0 status register

Offset Address: 0X006C			Register Name: CSI_CO_CAP_STA_REG
Bit	Read/Write	Default/Hex	Description
31:03	/	/	/
02	R	0	FIELD_STA  The status of the received field  0: Field 0  1: Field 1
01	R	0	VCAP_STA  Video capture in progress  Indicates the CSI is capturing video image data (multiple frames). The bit is set at the start of the first frame after enabling video capture. When software disables video capture, it clears itself after the last pixel of the current frame is captured.
00	R	0	SCAP_STA  Still capture in progress  Indicates the CSI is capturing still image data (single frame). The bit is set at the

			<p>start of the first frame after enabling still frame capture. It clears itself after the last pixel of the first frame is captured.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means filed end.</p>
--	--	--	---

#### 6.1.5.16. CSI Channel\_0 interrupt enable register

Offset Address: 0X0070			Register Name: CSI_CO_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	<p>VS_INT_EN</p> <p>VSYNC flag</p> <p>The bit is set when VSYNC comes. Load the buffer address for the coming frame at this time.</p> <p>So after this IRQ comes, change the buffer address could only effect next frame</p>
06	R/W	0	<p>HB_OF_INT_EN</p> <p>Horizontal blank FIFO overflow</p> <p>The bit is set when 3 FIFOs still overflow after the horizontal blank.</p>
05	R/W	0	<p>MUL_ERR_INT_EN</p> <p>Multi-channel writing error</p> <p>Indicates that error has been detected for writing data to a wrong channel.</p>
04	R/W	0	<p>FIFO2_OF_INT_EN</p> <p>FIFO 2 overflow</p>

			The bit is set when the FIFO 2 become overflow.
03	R/W	0	<p>FIFO1_OF_INT_EN</p> <p>FIFO 1 overflow</p> <p>The bit is set when the FIFO 1 become overflow.</p>
02	R/W	0	<p>FIFO0_OF_INT_EN</p> <p>FIFO 0 overflow</p> <p>The bit is set when the FIFO 0 become overflow.</p>
01	R/W	0	<p>FD_INT_EN</p> <p>Frame done</p> <p>Indicates the CSI has finished capturing an image frame. Apply to video capture mode. The bit is set after each completed frame data has been written to buffer as long as video capture remains enabled.</p>
00	R/W	0	<p>CD_INT_EN</p> <p>Capture done</p> <p>Indicates the CSI has completed capturing the image data.</p> <p>For still capture, the bit is set when one frame data has been written to buffer.</p> <p>For video capture, the bit is set when the last frame has been written to buffer after video capture was disabled.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.</p>



**6.1.5.17. CSI Channel\_0 interrupt status register**

Offset Address: 0X0074			Register Name: CSI_C0_INT_STA_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	VS_PD VSYNC flag
06	R/W	0	HB_OF_PD Horizontal Blank FIFO overflow
05	R/W	0	MUL_ERR_PD Multi-channel writing error
04	R/W	0	FIFO2_OF_PD FIFO 2 overflow
03	R/W	0	FIFO1_OF_PD FIFO 1 overflow
02	R/W	0	FIFO0_OF_PD FIFO 0 overflow
01	R/W	0	FD_PD Frame done

00	R/W	0	CD_PD  Capture done
----	-----	---	---------------------------

#### 6.1.5.18. CSI Channel\_0 horizontal size register

Offset Address: 0X0080			Register Name: CSI_CO_HSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	500	HOR_LEN  Horizontal pixel number  Unit in pixel
15:13	/	/	/
12:00	R/W	0	HOR_START  Start position of the active pixel.  Unit in pixel

#### 6.1.5.19. CSI Channel\_0 vertical size register

Offset Address: 0X0084			Register Name: CSI_CO_VSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN

			Vertical line number  Unit in line
15:13	/	/	/
12:00	R/W	0	VER_START  Start position of the active line.  Unit in line

#### 6.1.5.20. CSI Channel\_0 buffer length register

Offset Address: 0X0088			Register Name: CSI_CO_BUF_LEN_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	140	BUF_LEN_C  Buffer length for C  Unit in byte
15:14	/	/	/
13:00	R/W	280	BUF_LEN  Buffer length for Y  Unit in byte

#### 6.1.5.21. CSI Channel\_0 flip size register

Offset Address: 0X008C			Register Name: CSI_CO_FLIP_SIZE_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	

31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical number of lines  In VFLIP mode
15:13	/	/	/
12:00	R/W	280	VALID_LEN  Valid components of a line  In HFLIP mode

#### 6.1.5.22. CSI Channel\_0 frame clock counter register

Offset Address: 0x0090			Register Name: CSI_CO_FRM_CLK_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	/	/	/
23:00	R	0	FRM_CLK_CNT  Counter value between every frame. For instant hardware frame rate statics.  The internal counter is added by one every 24MHz clock cycle. When frame done or VSYNC comes, the internal counter value is sampled to FRM_CLK_CNT, and cleared to 0.

#### 6.1.5.23. CSI Channel\_0 accumulated and internal clock counter register

Offset Address: 0x0094	Register Name: CSI_CO_ACC_ITNL_CLK_CNT_REG
------------------------	--

Bit	Read/ Write	Default/Hex	Description
31:24	R/W	0	<p>ACC_CLK_CNT</p> <p>The accumulated value of FRM_CLK_CNT for software frame rate statics. Every interrupt of frame done, the software check this accumulated value and clear it to 0. If the ACC_CLK_CNT is larger than 1, the software has lost frame.</p> <p>When frame done or VSYNC comes, ACC_CLK_CNT = ACC_CLK_CNT + 1, and cleared to 0 when writing 0 to this register.</p>
23:00	R	0	<p>ITNL_CLK_CNT</p> <p>The instant value of internal frame clock counter.</p> <p>When frame done interrupt comes, the software can query this counter for judging whether it is the time for updating the double buffer address registers.</p>

#### 6.1.5.24. CSI Channel\_0 FIFO Statistic Register

Offset Address: 0x0098		Register Name: CSI_CO_FIFO_STAT_REG	
Bit	Read/ Write	Default/Hex	Description
31:12	/	/	/
11:00	R	/	<p>FIFO_FRM_MAX</p> <p>Indicates the maximum depth of FIFO being occupied for whole frame. Update at every VSYNC or FRAMEDONE.</p>

**6.1.5.25. CSI Channel\_0 PCLK Statistic Register**

Offset Address: 0x009c			Register Name: CSI_CO_PCLK_STAT_REG
Bit	Read/ Write	Default/Hex	Description
31	/	/	/
30:16	R	/	PCLK_CNT_LINE_MAX  Indicates maximum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.
15	/	/	/
14:00	R	0x7fff	PCLK_CNT_LINE_MIN  Indicates minimum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.

**6.1.5.26. CSI Channel\_1 configuration register**

Offset Address: 0X0144			Register Name: CSI_C1_CFG_REG
Bit	Read/Writ e	Default/H ex	Description
31:24	R/W	0	PAD_VAL  Padding value when OUTPUT_FMT is prgb888  0x00~0xff
23:20	R/W	3	INPUT_FMT  Input data format

			0000: RAW stream  0001: reserved  0010: reserved  0011: YUV422  0100: YUV420  Others: reserved
19:16	R/W	0	OUTPUT_FMT  Output data format  When the input format is set RAW stream  0000: field-raw-8  0001: field-raw-10  0010: field-raw-12  0011: reserved  0100: field-rgb565  0101: field-rgb888  0110: field-prgb888  0111: field-UV-combined  1000: frame-raw-8

			1001: frame-raw-10  1010: frame-raw-12  1011: reserved  1100: frame-rgb565  1101: frame-rgb888  1110: frame-prgb888 1111: frame-UV-combined   When the input format is set YUV422  0000: field planar YCbCr 422  0001: field planar YCbCr 420  0010: frame planar YCbCr 420  0011: frame planar YCbCr 422  0100: field planar YCbCr 422 UV combined  0101: field planar YCbCr 420 UV combined  0110: frame planar YCbCr 420 UV combined  0111: frame planar YCbCr 422 UV combined  1000: field MB YCbCr 422
--	--	--	---



			<p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: frame MB YCbCr 422</p> <p>1100: field planar YCbCr 422 10bit UV combined</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p> <p>When the input format is set YUV420</p> <p>0000: Reserved</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p> <p>0011: Reserved</p> <p>0100: Reserved</p> <p>0101: field planar YCbCr 420 UV combined</p> <p>0110: frame planar YCbCr 420 UV combined</p> <p>0111: Reserved</p> <p>1000: Reserved</p>
--	--	--	--

			<p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: Reserved</p> <p>1100: Reserved</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p> <p>Others: reserved</p>
15:14	/	/	/
13	R/W	0	<p>VFLIP_EN</p> <p>Vertical flip enable</p> <p>When enabled, the received data will be arranged in vertical flip.</p> <p>0:Disable</p> <p>1:Enable</p>
12	R/W	0	<p>HFLIP_EN</p> <p>Horizontal flip enable</p> <p>When enabled, the received data will be arranged in horizontal flip.</p> <p>0:Disable</p>

			1:Enable
11:10	R/W	0	FIELD_SEL  Field select  00: capture field 1.  01: capture field 2.  10: capture either field.  11: reserved
09:08	R/W	2	INPUT_SEQ  Input data sequence, only valid for YUV422 and YUV420 input format.  All data interleaved in one channel:  00: YUYV  01: YVYU  10: UYVY  11: VYUY  Y and UV in separated channel:  x0: UV  x1: VU

07:02	/	/	/
01:00			<p>MIN_SDR_WR_SIZE</p> <p>Minimum size of SDRAM block write</p> <p>0: 256 bytes (if HFLIP is enable, always select 256 bytes)</p> <p>1: 512 bytes</p> <p>2: 1k bytes</p> <p>3: 2k bytes</p>

#### 6.1.5.27. CSI Channel\_1 scale register

Offset Address: 0X014C			Register Name: CSI_C1_SCALE_REG
Bit	Read/Write	Default/Hex	Description
31:01	/	/	/
00	R/W	0	<p>QUART_EN</p> <p>When this bit is set to 1, input image will be subsample to quarter size. Support for all input formats.</p>

#### 6.1.5.28. CSI Channel\_1 FIFO 0 output buffer-A address register

Offset Address: 0X0150			Register Name: CSI_C1_F0_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	<p>C1F0_BUFA</p> <p>FIFO 0 output buffer-A address</p>

**6.1.5.29. CSI Channel\_1 FIFO 1 output buffer-A address register**

Offset Address: 0X0158			Register Name: CSI_C1_F1_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C1F1_BUFA  FIFO 1 output buffer-A address

**6.1.5.30. CSI Channel\_1 FIFO 2 output buffer-A address register**

Offset Address: 0X0160			Register Name: CSI_C1_F2_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C1F2_BUFA  FIFO 2 output buffer-A address

**6.1.5.31. CSI Channel\_1 status register**

Offset Address: 0X016C			Register Name: CSI_C1_CAP_STA_REG
Bit	Read/Write	Default/Hex	Description
31:03	/	/	/
02	R	0	FIELD_STA  The status of the received field  0: Field 0

			1: Field 1
01	R	0	<p>VCAP_STA</p> <p>Video capture in progress</p> <p>Indicates the CSI is capturing video image data (multiple frames). The bit is set at the start of the first frame after enabling video capture. When software disables video capture, it clears itself after the last pixel of the current frame is captured.</p>
00	R	0	<p>SCAP_STA</p> <p>Still capture in progress</p> <p>Indicates the CSI is capturing still image data (single frame). The bit is set at the start of the first frame after enabling still frame capture. It clears itself after the last pixel of the first frame is captured.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means filed end.</p>

#### 6.1.5.32. CSI Channel\_1 interrupt enable register

Offset Address: 0X0170			Register Name: CSI_C1_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	<p>VS_INT_EN</p> <p>VSYNC flag</p> <p>The bit is set when VSYNC comes. Load the buffer address for the coming frame at this time.</p> <p>So after this IRQ comes, change the buffer address could only effect next frame</p>

06	R/W	0	<p>HB_OF_INT_EN</p> <p>Horizontal blank FIFO overflow</p> <p>The bit is set when 3 FIFOs still overflow after the horizontal blank.</p>
05	R/W	0	<p>MUL_ERR_INT_EN</p> <p>Multi-channel writing error</p> <p>Indicates that error has been detected for writing data to a wrong channel.</p>
04	R/W	0	<p>FIFO2_OF_INT_EN</p> <p>FIFO 2 overflow</p> <p>The bit is set when the FIFO 2 become overflow.</p>
03	R/W	0	<p>FIFO1_OF_INT_EN</p> <p>FIFO 1 overflow</p> <p>The bit is set when the FIFO 1 become overflow.</p>
02	R/W	0	<p>FIFO0_OF_INT_EN</p> <p>FIFO 0 overflow</p> <p>The bit is set when the FIFO 0 become overflow.</p>
01	R/W	0	<p>FD_INT_EN</p> <p>Frame done</p> <p>Indicates the CSI has finished capturing an image frame. Apply to video capture mode. The bit is set after each completed frame data has been written to buffer as long as video capture remains enabled.</p>
00	R/W	0	<p>CD_INT_EN</p>

			<p>Capture done</p> <p>Indicates the CSI has completed capturing the image data.</p> <p>For still capture, the bit is set when one frame data has been written to buffer.</p> <p>For video capture, the bit is set when the last frame has been written to buffer after video capture was disabled.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.</p>
--	--	--	---

#### 6.1.5.33. CSI Channel\_1 interrupt status register

Offset Address: 0X0174			Register Name: CSI_C1_INT_STA_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	VS_PD  VSYNC flag
06	R/W	0	HB_OF_PD  Horizontal Blank FIFO overflow
05	R/W	0	MUL_ERR_PD  Multi-channel writing error
04	R/W	0	FIFO2_OF_PD  FIFO 2 overflow
03	R/W	0	FIFO1_OF_PD



			FIFO 1 overflow
02	R/W	0	FIFO0_OF_PD FIFO 0 overflow
01	R/W	0	FD_PD Frame done
00	R/W	0	CD_PD Capture done

#### 6.1.5.34. CSI Channel\_1 horizontal size register

Offset Address: 0X0180			Register Name: CSI_C1_HSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	500	HOR_LEN Horizontal pixel number Unit in pixel
15:13	/	/	/
12:00	R/W	0	HOR_START

			Start position of the active pixel.  Unit in pixel
--	--	--	--

#### 6.1.5.35. CSI Channel\_1 vertical size register

Offset Address: 0X0184			Register Name: CSI_C1_VSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical line number  Unit in line
15:13	/	/	/
12:00	R/W	0	VER_START  Start position of the active line.  Unit in line

#### 6.1.5.36. CSI Channel\_1 buffer length register

Offset Address: 0X0188			Register Name: CSI_C1_BUF_LEN_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	140	BUF_LEN_C  Buffer length for C  Unit in byte

15:14	/	/	/
13:00	R/W	280	BUF_LEN  Buffer length for Y  Unit in byte

#### 6.1.5.37. CSI Channel\_1 flip size register

Offset Address: 0X018C			Register Name: CSI_C1_FLIP_SIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical number of lines  In VFLIP mode
15:13	/	/	/
12:00	R/W	280	VALID_LEN  Valid components of a line  In HFLIP mode

#### 6.1.5.38. CSI Channel\_1 frame clock counter register

Offset Address: 0x0190			Register Name: CSI_C1_FRM_CLK_CNT_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:00	R	0	FRM_CLK_CNT

			<p>Counter value between every frame. For instant hardware frame rate statics.</p> <p>The internal counter is added by one every 24MHz clock cycle. When frame done or VSYNC comes, the internal counter value is sampled to FRM_CLK_CNT, and cleared to 0. Then the FRM_CLK_CNT is added to ACC_CLK_CNT.</p>
--	--	--	---

#### 6.1.5.39. CSI Channel\_1 accumulated and internal clock counter register

Offset Address: 0x0194			Register Name: CSI_C1_ACC_ITNL_CLK_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	R/W	0	<p>ACC_CLK_CNT</p> <p>The accumulated value of FRM_CLK_CNT for software frame rate statics. Every interrupt of frame done, the software check this accumulated value and clear it to 0. If the ACC_CLK_CNT is larger than 1, the software has lost frame.</p> <p>When frame done or VSYNC comes, <math>ACC\_CLK\_CNT = ACC\_CLK\_CNT + 1</math>, and cleared to 0 when writing 0 to this register.</p>
23:00	R	0	<p>ITNL_CLK_CNT</p> <p>The instant value of internal frame clock counter.</p> <p>When frame done interrupt comes, the software can query this counter for judging whether it is the time for updating the double buffer address registers.</p>

#### 6.1.5.40. CSI Channel\_1 FIFO Statistic Register

Offset Address: 0x0198	Register Name: CSI_C1_FIFO_STAT_REG
------------------------	-------------------------------------

Bit	Read/ Write	Default/Hex	Description
31:12	/	/	/
11:00	R	/	FIFO_FRM_MAX  Indicates the maximum depth of FIFO being occupied for whole frame. Update at every VSYNC or FRAMEDONE.

#### 6.1.5.41. CSI Channel\_1 PCLK Statistic Register

Offset Address: 0x019c		Register Name: CSI_C1_PCLK_STAT_REG	
Bit	Read/ Write	Default/Hex	Description
31	/	/	/
30:16	R	/	PCLK_CNT_LINE_MAX  Indicates maximum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.
15	/	/	/
14:00	R	0x7fff	PCLK_CNT_LINE_MIN  Indicates minimum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.

#### 6.1.5.42. CSI Channel\_2 configuration register

Offset Address: 0X0244		Register Name: CSI_C2_CFG_REG	
Bit	Read/Writ e	Default/H ex	Description

31:24	R/W	0	PAD_VAL  Padding value when OUTPUT_FMT is prgb888  0x00~0xff
23:20	R/W	3	INPUT_FMT  Input data format  0000: RAW stream  0001: reserved  0010: reserved  0011: YUV422  0100: YUV420  Others: reserved
19:16	R/W	0	OUTPUT_FMT  Output data format  When the input format is set RAW stream  0000: field-raw-8  0001: field-raw-10  0010: field-raw-12  0011: reserved  0100: field-rgb565

			0101: field-rgb888  0110: field-prgb888  0111: field-UV-combined  1000: frame-raw-8  1001: frame-raw-10  1010: frame-raw-12  1011: reserved  1100: frame-rgb565  1101: frame-rgb888  1110: frame-prgb888  1111: frame-UV-combined   When the input format is set YUV422  0000: field planar YCbCr 422  0001: field planar YCbCr 420  0010: frame planar YCbCr 420  0011: frame planar YCbCr 422  0100: field planar YCbCr 422 UV combined
--	--	--	--

			<p>0101: field planar YCbCr 420 UV combined</p> <p>0110: frame planar YCbCr 420 UV combined</p> <p>0111: frame planar YCbCr 422 UV combined</p> <p>1000: field MB YCbCr 422</p> <p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: frame MB YCbCr 422</p> <p>1100: field planar YCbCr 422 10bit UV combined</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p> <p>When the input format is set YUV420</p> <p>0000: Reserved</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p> <p>0011: Reserved</p> <p>0100: Reserved</p>
--	--	--	--



			0101: field planar YCbCr 420 UV combined  0110: frame planar YCbCr 420 UV combined  0111: Reserved  1000: Reserved  1001: field MB YCbCr 420  1010: frame MB YCbCr 420  1011: Reserved  1100: Reserved  1101: field planar YCbCr 420 10bit UV combined  1110: Reserved  1111: Reserved  Others: reserved
15:14	/	/	/
13	R/W	0	VFLIP_EN  Vertical flip enable  When enabled, the received data will be arranged in vertical flip.  0:Disable  1:Enable

12	R/W	0	<p>HFLIP_EN</p> <p>Horizontal flip enable</p> <p>When enabled, the received data will be arranged in horizontal flip.</p> <p>0:Disable</p> <p>1:Enable</p>
11:10	R/W	0	<p>FIELD_SEL</p> <p>Field select</p> <p>00: capture field 1.</p> <p>01: capture field 2.</p> <p>10: capture either field.</p> <p>11: reserved</p>
09:08	R/W	2	<p>INPUT_SEQ</p> <p>Input data sequence, only valid for YUV422 and YUV420 input format.</p> <p>All data interleaved in one channel:</p> <p>00: YUYV</p> <p>01: YVYU</p> <p>10: UYVY</p> <p>11: VYUY</p>

			Y and UV in separated channel:  x0: UV  x1: VU
07:02	/	/	/
01:00			MIN_SDR_WR_SIZE  Minimum size of SDRAM block write  0: 256 bytes (if HFLIP is enable, always select 256 bytes)  1: 512 bytes  2: 1k bytes  3: 2k bytes

#### 6.1.5.43. CSI Channel\_2 scale register

Offset Address: 0X024C			Register Name: CSI_C2_SCALE_REG
Bit	Read/Write	Default/Hex	Description
31:01	/	/	/
00	R/W	0	QUART_EN  When this bit is set to 1, input image will be subsample to quarter size. Support for all input formats.

#### 6.1.5.44. CSI Channel\_2 FIFO 0 output buffer-A address register

Offset Address: 0X0250	Register Name: CSI_C2_F0_BUFA_REG
------------------------	-----------------------------------

Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C2F0_BUFA  FIFO 0 output buffer-A address

#### 6.1.5.45. CSI Channel\_2 FIFO 1 output buffer-A address register

Offset Address: 0X0258			Register Name: CSI_C2_F1_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C2F1_BUFA  FIFO 1 output buffer-A address

#### 6.1.5.46. CSI Channel\_2 FIFO 2 output buffer-A address register

Offset Address: 0X0260			Register Name: CSI_C2_F2_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C2F2_BUFA  FIFO 2 output buffer-A address

#### 6.1.5.47. CSI Channel\_2 status register

Offset Address: 0X026C			Register Name: CSI_C2_CAP_STA_REG
Bit	Read/Write	Default/Hex	Description
31:03	/	/	/

02	R	0	<p>FIELD_STA</p> <p>The status of the received field</p> <p>0: Field 0</p> <p>1: Field 1</p>
01	R	0	<p>VCAP_STA</p> <p>Video capture in progress</p> <p>Indicates the CSI is capturing video image data (multiple frames). The bit is set at the start of the first frame after enabling video capture. When software disables video capture, it clears itself after the last pixel of the current frame is captured.</p>
00	R	0	<p>SCAP_STA</p> <p>Still capture in progress</p> <p>Indicates the CSI is capturing still image data (single frame). The bit is set at the start of the first frame after enabling still frame capture. It clears itself after the last pixel of the first frame is captured.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means filed end.</p>

#### 6.1.5.48. CSI Channel\_2 interrupt enable register

Offset Address: 0X0270			Register Name: CSI_C2_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	VS_INT_EN

			<p>VSYNC flag</p> <p>The bit is set when VSYNC comes. Load the buffer address for the coming frame at this time.</p> <p>So after this IRQ comes, change the buffer address could only effect next frame</p>
06	R/W	0	<p>HB_OF_INT_EN</p> <p>Horizontal blank FIFO overflow</p> <p>The bit is set when 3 FIFOs still overflow after the horizontal blank.</p>
05	R/W	0	<p>MUL_ERR_INT_EN</p> <p>Multi-channel writing error</p> <p>Indicates that error has been detected for writing data to a wrong channel.</p>
04	R/W	0	<p>FIFO2_OF_INT_EN</p> <p>FIFO 2 overflow</p> <p>The bit is set when the FIFO 2 become overflow.</p>
03	R/W	0	<p>FIFO1_OF_INT_EN</p> <p>FIFO 1 overflow</p> <p>The bit is set when the FIFO 1 become overflow.</p>
02	R/W	0	<p>FIFO0_OF_INT_EN</p> <p>FIFO 0 overflow</p> <p>The bit is set when the FIFO 0 become overflow.</p>
01	R/W	0	<p>FD_INT_EN</p> <p>Frame done</p>

			Indicates the CSI has finished capturing an image frame. Apply to video capture mode. The bit is set after each completed frame data has been written to buffer as long as video capture remains enabled.
00	R/W	0	<p>CD_INT_EN</p> <p>Capture done</p> <p>Indicates the CSI has completed capturing the image data.</p> <p>For still capture, the bit is set when one frame data has been written to buffer.</p> <p>For video capture, the bit is set when the last frame has been written to buffer after video capture was disabled.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.</p>

#### 6.1.5.49. CSI Channel\_2 interrupt status register

Offset Address: 0X0274			Register Name: CSI_C2_INT_STA_REG
Bit	Read/Write	Default/Hex	Description
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:08	/	/	/
07	R/W	0	<p>VS_PD</p> <p>VSYNC flag</p>
06	R/W	0	<p>HB_OF_PD</p> <p>Horizontal Blank FIFO overflow</p>
05	R/W	0	MUL_ERR_PD

			Multi-channel writing error
04	R/W	0	FIFO2_OF_PD  FIFO 2 overflow
03	R/W	0	FIFO1_OF_PD  FIFO 1 overflow
02	R/W	0	FIFO0_OF_PD  FIFO 0 overflow
01	R/W	0	FD_PD  Frame done

#### 6.1.5.50. CSI Channel\_2 horizontal size register

Offset Address: 0X0280			Register Name: CSI_C2_HSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	500	HOR_LEN  Horizontal pixel number  Unit in pixel
15:13	/	/	/



12:00	R/W	0	HOR_START  Start position of the active pixel.  Unit in pixel
-------	-----	---	---

#### 6.1.5.51. CSI Channel\_2 vertical size register

Offset Address: 0X0284			Register Name: CSI_C2_VSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical line number  Unit in line
15:13	/	/	/
12:00	R/W	0	VER_START  Start position of the active line.  Unit in line

#### 6.1.5.52. CSI Channel\_2 buffer length register

Offset Address: 0X0288			Register Name: CSI_C2_BUF_LEN_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	140	BUF_LEN_C  Buffer length for C

			Unit in byte
15:14	/	/	/
13:00	R/W	280	BUF_LEN  Buffer length for Y  Unit in byte

#### 6.1.5.53. CSI Channel\_2 flip size register

Offset Address: 0X028C			Register Name: CSI_C2_FLIP_SIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical number of lines  In VFLIP mode
15:13	/	/	/
12:00	R/W	280	VALID_LEN  Valid components of a line  In HFLIP mode

#### 6.1.5.54. CSI Channel\_2 frame clock counter register

Offset Address: 0x0290			Register Name: CSI_C2_FRM_CLK_CNT_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/

23:00	R	0	<p>FRM_CLK_CNT</p> <p>Counter value between every frame. For instant hardware frame rate statics.</p> <p>The internal counter is added by one every 24MHz clock cycle. When frame done or VSYNC comes, the internal counter value is sampled to FRM_CLK_CNT, and cleared to 0. Then the FRM_CLK_CNT is added to ACC_CLK_CNT.</p>
-------	---	---	--

#### 6.1.5.55. CSI Channel\_2 accumulated and internal clock counter register

Offset Address: 0x0294			Register Name: CSI_C2_ACC_ITNL_CLK_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	R/W	0	<p>ACC_CLK_CNT</p> <p>The accumulated value of FRM_CLK_CNT for software frame rate statics. Every interrupt of frame done, the software check this accumulated value and clear it to 0. If the ACC_CLK_CNT is larger than 1, the software has lost frame.</p> <p>When frame done or VSYNC comes, <math>ACC\_CLK\_CNT = ACC\_CLK\_CNT + 1</math>, and cleared to 0 when writing 0 to this register.</p>
23:00	R	0	<p>ITNL_CLK_CNT</p> <p>The instant value of internal frame clock counter.</p> <p>When frame done interrupt comes, the software can query this counter for judging whether it is the time for updating the double buffer address registers.</p>

#### 6.1.5.56. CSI Channel\_2 FIFO Statistic Register

Offset Address: 0x0298			Register Name: CSI_C2_FIFO_STAT_REG
Bit	Read/ Write	Default/Hex	Description
31:12	/	/	/
11:00	R	/	FIFO_FRM_MAX  Indicates the maximum depth of FIFO being occupied for whole frame. Update at every VSYNC or FRAMEDONE.

#### 6.1.5.57. CSI Channel\_2 PCLK Statistic Register

Offset Address: 0x029c			Register Name: CSI_C2_PCLK_STAT_REG
Bit	Read/ Write	Default/Hex	Description
31	/	/	/
30:16	R	/	PCLK_CNT_LINE_MAX  Indicates maximum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.
15	/	/	/
14:00	R	0x7fff	PCLK_CNT_LINE_MIN  Indicates minimum pixel clock counter value for each line.  Update at every VSYNC or FRAMEDONE.

#### 6.1.5.58. CSI Channel\_3 configuration register

Offset Address: 0X0344			Register Name: CSI_C3_CFG_REG
------------------------	--	--	-------------------------------

Bit	Read/Write	Default/Hex	Description
31:24	R/W	0	PAD_VAL  Padding value when OUTPUT_FMT is prgb888  0x00~0xff
23:20	R/W	3	INPUT_FMT  Input data format  0000: RAW stream  0001: reserved  0010: reserved  0011: YUV422  0100: YUV420  Others: reserved
19:16	R/W	0	OUTPUT_FMT  Output data format  When the input format is set RAW stream  0000: field-raw-8  0001: field-raw-10  0010: field-raw-12

			<p>0011: reserved</p> <p>0100: field-rgb565</p> <p>0101: field-rgb888</p> <p>0110: field-prgb888</p> <p>0111: field-UV-combined</p> <p>1000: frame-raw-8</p> <p>1001: frame-raw-10</p> <p>1010: frame-raw-12</p> <p>1011: reserved</p> <p>1100: frame-rgb565</p> <p>1101: frame-rgb888</p> <p>1110: frame-prgb888</p> <p>1111: frame-UV-combined</p> <p>When the input format is set YUV422</p> <p>0000: field planar YCbCr 422</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p>
--	--	--	---

			<p>0011: frame planar YCbCr 422</p> <p>0100: field planar YCbCr 422 UV combined</p> <p>0101: field planar YCbCr 420 UV combined</p> <p>0110: frame planar YCbCr 420 UV combined</p> <p>0111: frame planar YCbCr 422 UV combined</p> <p>1000: field MB YCbCr 422</p> <p>1001: field MB YCbCr 420</p> <p>1010: frame MB YCbCr 420</p> <p>1011: frame MB YCbCr 422</p> <p>1100: field planar YCbCr 422 10bit UV combined</p> <p>1101: field planar YCbCr 420 10bit UV combined</p> <p>1110: Reserved</p> <p>1111: Reserved</p> <p>When the input format is set YUV420</p> <p>0000: Reserved</p> <p>0001: field planar YCbCr 420</p> <p>0010: frame planar YCbCr 420</p>
--	--	--	--

			0011: Reserved  0100: Reserved  0101: field planar YCbCr 420 UV combined  0110: frame planar YCbCr 420 UV combined  0111: Reserved  1000: Reserved  1001: field MB YCbCr 420  1010: frame MB YCbCr 420  1011: Reserved  1100: Reserved  1101: field planar YCbCr 420 10bit UV combined  1110: Reserved  1111: Reserved  Others: reserved
15:14	/	/	/
13	R/W	0	VFLIP_EN  Vertical flip enable  When enabled, the received data will be arranged in vertical flip.



			0:Disable  1:Enable
12	R/W	0	HFLIP_EN  Horizontal flip enable  When enabled, the received data will be arranged in horizontal flip.  0:Disable  1:Enable
11:10	R/W	0	FIELD_SEL  Field select  00: capture field 1.  01: capture field 2.  10: capture either field.  11: reserved
09:08	R/W	2	INPUT_SEQ  Input data sequence, only valid for YUV422 and YUV420 input format.  All data interleaved in one channel:  00: YUYV  01: YVYU

			10: UYVY  11: VYUY  Y and UV in separated channel:  x0: UV  x1: VU
07:02	/	/	/
01:00			MIN_SDR_WR_SIZE  Minimum size of SDRAM block write  0: 256 bytes (if HFLIP is enable, always select 256 bytes)  1: 512 bytes  2: 1k bytes  3: 2k bytes

#### 6.1.5.59. CSI Channel\_3 scale register

Offset Address: 0X034C			Register Name: CSI_C3_SCALE_REG
Bit	Read/Write	Default/Hex	Description
31:01	/	/	/
00	R/W	0	QUART_EN  When this bit is set to 1, input image will be subsample to quarter size. Support for all input formats.

**6.1.5.60. CSI Channel\_3 FIFO 0 output buffer-A address register**

Offset Address: 0X0350			Register Name: CSI_C3_F0_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C3F0_BUFA  FIFO 0 output buffer-A address

**6.1.5.61. CSI Channel\_3 FIFO 1 output buffer-A address register**

Offset Address: 0X0358			Register Name: CSI_C3_F1_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C3F1_BUFA  FIFO 1 output buffer-A address

**6.1.5.62. CSI Channel\_3 FIFO 2 output buffer-A address register**

Offset Address: 0X0360			Register Name: CSI_C3_F2_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:00	R/W	0	C3F2_BUFA  FIFO 2 output buffer-A address

### 6.1.5.63. CSI Channel\_3 status register

Offset Address: 0X036C			Register Name: CSI_C3_CAP_STA_REG
Bit	Read/Write	Default/Hex	Description
31:03	/	/	/
02	R	0	<p>FIELD_STA</p> <p>The status of the received field</p> <p>0: Field 0</p> <p>1: Field 1</p>
01	R	0	<p>VCAP_STA</p> <p>Video capture in progress</p> <p>Indicates the CSI is capturing video image data (multiple frames). The bit is set at the start of the first frame after enabling video capture. When software disables video capture, it clears itself after the last pixel of the current frame is captured.</p>
00	R	0	<p>SCAP_STA</p> <p>Still capture in progress</p> <p>Indicates the CSI is capturing still image data (single frame). The bit is set at the start of the first frame after enabling still frame capture. It clears itself after the last pixel of the first frame is captured.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.</p>

**6.1.5.64. CSI Channel\_3 interrupt enable register**

Offset Address: 0X0370			Register Name: CSI_C3_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:08	/	/	/
07	R/W	0	VS_INT_EN  VSYNC flag  The bit is set when VSYNC comes. Load the buffer address for the coming frame at this time.  So after this IRQ comes, change the buffer address could only effect next frame
06	R/W	0	HB_OF_INT_EN  Horizontal blank FIFO overflow  The bit is set when 3 FIFOs still overflow after the horizontal blank.
05	R/W	0	MUL_ERR_INT_EN  Multi-channel writing error  Indicates that error has been detected for writing data to a wrong channel.
04	R/W	0	FIFO2_OF_INT_EN  FIFO 2 overflow  The bit is set when the FIFO 2 become overflow.
03	R/W	0	FIFO1_OF_INT_EN  FIFO 1 overflow  The bit is set when the FIFO 1 become overflow.

02	R/W	0	<p>FIFO0_OF_INT_EN</p> <p>FIFO 0 overflow</p> <p>The bit is set when the FIFO 0 become overflow.</p>
01	R/W	0	<p>FD_INT_EN</p> <p>Frame done</p> <p>Indicates the CSI has finished capturing an image frame. Apply to video capture mode. The bit is set after each completed frame data has been written to buffer as long as video capture remains enabled.</p>
00	R/W	0	<p>CD_INT_EN</p> <p>Capture done</p> <p>Indicates the CSI has completed capturing the image data.</p> <p>For still capture, the bit is set when one frame data has been written to buffer.</p> <p>For video capture, the bit is set when the last frame has been written to buffer after video capture was disabled.</p> <p>For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.</p>

#### 6.1.5.65. CSI Channel\_3 interrupt status register

Offset Address: 0X0374		Register Name: CSI_C3_INT_STA_REG	
Bit	Read/Write	Default/Hex	Description
	e	ex	
Bit	Read/Write	Default/Hex	Description
	e	ex	

31:08	/	/	/
07	R/W	0	VS_PD  VSYNC flag
06	R/W	0	HB_OF_PD  Horizontal Blank FIFO overflow
05	R/W	0	MUL_ERR_PD  Multi-channel writing error
04	R/W	0	FIFO2_OF_PD  FIFO 2 overflow
03	R/W	0	FIFO1_OF_PD  FIFO 1 overflow
02	R/W	0	FIFO0_OF_PD  FIFO 0 overflow
01	R/W	0	FD_PD  Frame done

#### 6.1.5.66. CSI Channel\_3 horizontal size register

Offset Address: 0X0380	Register Name: CSI_C3_HSIZE_REG
------------------------	---------------------------------

Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	500	HOR_LEN  Horizontal pixel number  Unit in pixel
15:13	/	/	/
12:00	R/W	0	HOR_START  Start position of the active pixel.  Unit in pixel

#### 6.1.5.67. CSI Channel\_3 vertical size register

Offset Address: 0X0384			Register Name: CSI_C3_VSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN  Vertical line number  Unit in line
15:13	/	/	/
12:00	R/W	0	VER_START  Start position of the active line.  Unit in line



**6.1.5.68. CSI Channel\_3 buffer length register**

Offset Address: 0X0388			Register Name: CSI_C3_BUF_LEN_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	140	BUF_LEN_C Buffer length for C Unit in byte
15:14	/	/	/
13:00	R/W	280	BUF_LEN Buffer length for Y Unit in byte

**6.1.5.69. CSI Channel\_3 flip size register**

Offset Address: 0X038C			Register Name: CSI_C3_FLIP_SIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	1E0	VER_LEN Vertical number of lines In VFLIP mode
15:13	/	/	/
12:00	R/W	280	VALID_LEN

			Valid components of a line  In HFLIP mode
--	--	--	---

#### 6.1.5.70. CSI Channel\_3 frame clock counter register

Offset Address: 0x0390			Register Name: CSI_C3_FRM_CLK_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	/	/	/
23:00	R	0	FRM_CLK_CNT  Counter value between every frame. For instant hardware frame rate statics.  The internal counter is added by one every 24MHz clock cycle. When frame done or VSYNC comes, the internal counter value is sampled to FRM_CLK_CNT, and cleared to 0. Then the FRM_CLK_CNT is added to ACC_CLK_CNT.

#### 6.1.5.71. CSI Channel\_3 accumulated and internal clock counter register

Offset Address: 0x0394			Register Name: CSI_C3_ACC_ITNL_CLK_CNT_REG
Bit	Read/ Write	Default/Hex	Description
31:24	R/W	0	ACC_CLK_CNT  The accumulated value of FRM_CLK_CNT for software frame rate statics. Every interrupt of frame done, the software check this accumulated value and clear it to 0. If the ACC_CLK_CNT is larger than 1, the software has lost frame.

			When frame done or VSYNC comes, ACC_CLK_CNT = ACC_CLK_CNT + 1, and cleared to 0 when writing 0 to this register.
23:00	R	0	<p>ITNL_CLK_CNT</p> <p>The instant value of internal frame clock counter.</p> <p>When frame done interrupt comes, the software can query this counter for judging whether it is the time for updating the double buffer address registers.</p>

#### 6.1.5.72. CSI Channel\_3 FIFO Statistic Register

Offset Address: 0x0398			Register Name: CSI_C3_FIFO_STAT_REG
Bit	Read/ Write	Default/Hex	Description
31:12	/	/	/
11:00	R	/	<p>FIFO_FRM_MAX</p> <p>Indicates the maximum depth of FIFO being occupied for whole frame. Update at every VSYNC or framedone.</p>

#### 6.1.5.73. CSI Channel\_3 PCLK Statistic Register

Offset Address: 0x039c			Register Name: CSI_C3_PCLK_STAT_REG
Bit	Read/ Write	Default/Hex	Description
31	/	/	/
30:16	R	/	<p>PCLK_CNT_LINE_MAX</p> <p>Indicates maximum pixel clock counter value for each line.</p>

			Update at every VSYNC or framedone.
15	/	/	/
14:00	R	0x7fff	<p>PCLK_CNT_LINE_MIN</p> <p>Indicates minimum pixel clock counter value for each line.</p> <p>Update at every VSYNC or framedone.</p>

#### 6.1.5.74. CCI Control Register

Offset Address: 0x3000			Register Name: CCI_CTRL
Bit	Read/ Write	Default/Hex	Description
31	R/W	0	<p>SINGLE_TRAN</p> <p>0: Transmission idle</p> <p>1: Start single transmission</p> <p>Automatically cleared to '0' when finished. Abort current transmission immediately if changing from '1' to '0'. If slave not respond for the expected status over the time defined by TIMEOUT, current transmission will stop. PACKET_CNT will return the sequence number when transmission is fail. All format setting and data will be loaded from registers and FIFO when transmission started.</p>
30	R/W	0	<p>REPEAT_TRAN</p> <p>0: transmission idle</p> <p>1: repeated transmission</p> <p>When this bit is set to 1, transmission repeats when trigger signal (such as</p>

			<p>VSYNC/ VCAP is done) repeats.</p> <p>If changing this bit from '1' to '0' during transmission, the current transmission will be guaranteed then stop.</p>
29	R/W	0	<p>RESTART_MODE</p> <p>0: RESTART</p> <p>1: STOP+START</p> <p>Define the CCI action after sending register address.</p>
28	R/W	0	<p>READ_TRAN_MODE</p> <p>0: send slave ID + W</p> <p>1: do not send slave ID + W</p> <p>Setting this bit to 1 if reading from a slave with zero width register.</p>
27:24	R	0	<p>TRAN_RESULT</p> <p>000: OK</p> <p>001: FAIL</p> <p>Other: Reserved</p>
23:16	R	/	<p>CCI_STA</p> <p>0x00: bus error</p> <p>0x08: START condition transmitted</p> <p>0x10: Repeated START condition transmitted</p> <p>0x18: Address + Write bit transmitted, ACK received</p>

			0x20: Address + Write bit transmitted, ACK not received  0x28: Data byte transmitted in master mode, ACK received  0x30: Data byte transmitted in master mode, ACK not received  0x38: Arbitration lost in address or data byte  0x40: Address + Read bit transmitted, ACK received  0x48: Address + Read bit transmitted, ACK not received  0x50: Data byte received in master mode, ACK received  0x58: Data byte received in master mode, ACK not received  0x01: Timeout when sending 9 <sup>th</sup> SCL clock  Other: Reserved
15:2	/	/	/
1	R/W	0	SOFT_RESET  0: normal  1: reset
0	R/W	0	CCI_EN  0: Module disable  1: Module enable

#### 6.1.5.75. CCI Transmission Configuration Register

Offset Address: 0x3004	Register Name: CCI_CFG
------------------------	------------------------

Bit	Read/ Write	Default/Hex	Description
31:24	R/W	0x10	<p>TIMEOUT_N</p> <p>When sending the 9<sup>th</sup> clock, assert fail signal when slave device did not response after <math>N * F_{SCL}</math> cycles. And software must do a reset to CCI module and send a stop condition to slave.</p>
23:16	R/W	0x00	<p>INTERVAL</p> <p>Define the interval between each packet in <math>40 * F_{SCL}</math> cycles. 0~255</p>
15	R/W	0	<p>PACKET_MODE</p> <p>Select where to load slave id / data width</p> <p>0: Compact mode</p> <p>1: Complete mode</p> <p>In compact mode, slave id/register width / data width will be loaded from CCI_FMT register, only address and data read from memory.</p> <p>In complete mode, they will be loaded from packet memory.</p>
14:8	/	/	/
7	R/W	0	Reserved
6:4	R/W	0	<p>TRIG_MODE</p> <p>Transmit mode:</p> <p>000: Immediately, no trigger</p> <p>001: Reserved</p>

			010: CSI0 interrupt trigger 011: CSI1 interrupt trigger
3:0	R/W	0	CSI_TRIG  CSI interrupt trigger signal select:  0000: First HREF start  0001: Last HREF done  0010: Line counter trigger  other: Reserved

#### 6.1.5.76. CCI Packet Format Register

Offset Address: 0x3008			Register Name: CCI_FMT
Bit	Read/ Write	Default/Hex	Description
31:25	R/W	0	SLV_ID  7bit address
24	R/W	0	CMD  0: write  1: read
23:20	R/W	1	ADDR_BYTE  How many bytes be sent as address  0~15



19:16	R/W	1	<p>DATA_BYTE</p> <p>How many bytes be sent/received as data</p> <p>1~15</p> <p>Normally use ADDR_DATA with 0_2, 1_1, 1_2, 2_1, 2_2 access mode. If DATA_BYTE is 0, transmission will not start. In complete mode, the ADDR_BYTE and DATA_BYTE are defined in a byte's high/low 4bit.</p>
15:0	R/W	1	<p>PACKET_CNT</p> <p>FIFO data is transmitted as PACKET_CNT packets in current format.</p> <p>Total bytes not exceed 32bytes.</p>

#### 6.1.5.77. CCI Bus Control Register

Offset Address: 0x300C			Register Name: CCI_BUS_CTRL
Bit	Read/ Write	Default/Hex	Description
31:16	R/W	0	<p>DLY_CYC</p> <p>0~65535 F<sub>SCL</sub> cycles between each transmission</p>
15	R/W	0	<p>DLY_TRIG</p> <p>0: disable</p> <p>1: execute transmission after internal counter delay when triggered</p>
14:12	R/W	0x2	<p>CLK_N</p> <p>CCI bus sampling clock F0=24MHz/2^CLK_N</p>
11:8	R/W	0x5	CLK_M

			CCI output SCL frequency is $F_{SCL}=F/10=(F0/(CLK\_M+1))/10$
7	R	/	SCL_STA SCL current status
6	R	/	SDA_STA SDA current status
5	R/W	0	SCL_PEN SCL PAD enable
4	R/W	0	SDA_PEN SDA PAD enable
3	R/W	0	SCL_MOV SCL manual output value
2	R/W	0	SDA_MOV SDA manual output value
1	R/W	0	SCL_MOE SCL manual output en
0	R/W	0	SDA_MOE SDA manual output en

#### 6.1.5.78. CCI Interrupt Control Register

Offset Address: 0x3014		Register Name: CCI_INT_CTRL	
Bit	Read/ Write	Default/Hex	Description
31:18	/	/	/

17	R/W	0	S_TRAN_ERR_INT_EN
16	R/W	0	S_TRAN_COM_INT_EN
15:2	/	/	/
1	R/W	0	S_TRAN_ERR_PD
0	R/W	0	S_TRAN_COM_PD

#### 6.1.5.79. CCI Line Counter Trigger Control Register

Offset Address: 0x3018			Register Name: CCI_LC_TRIG
Bit	Read/ Write	Default/Hex	Description
31:13	/	/	/
12:0	R/W	0	LN_CNT  0~8191: line counter send trigger when 1 <sup>st</sup> ~8192 <sup>th</sup> line is received.

#### 6.1.5.80. CCI FIFO Access Register

Offset Address: 0x3100~0x313f			Register Name: CCI_FIFO_ACC
Bit	Read/ Write	Default/Hex	Description
31:0	R/W	0	DATA_FIFO  From 0x100 to 0x13f, CCI data FIFO is 64bytes, used in FIFO input mode. CCI transmission read/write data from/to FIFO in byte.

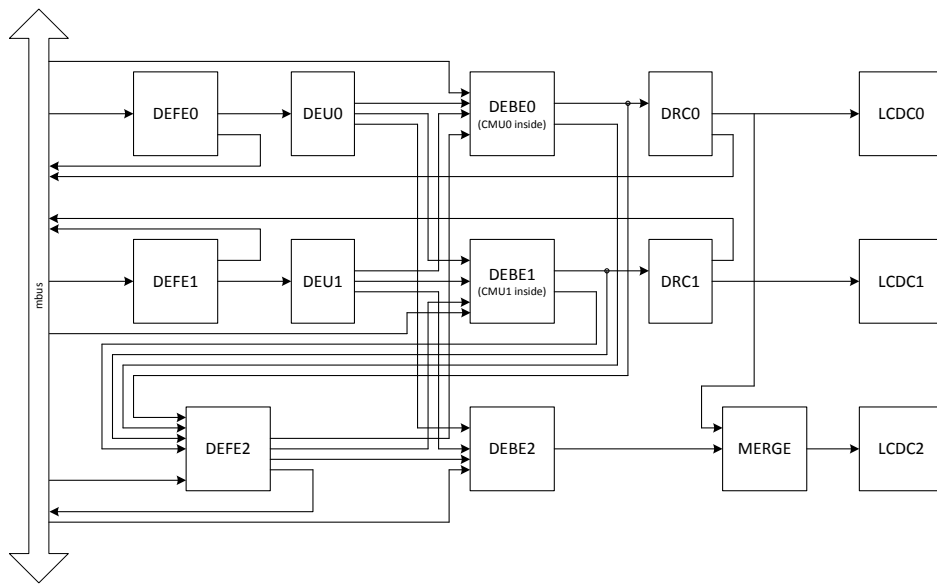
## Chapter 7 Display

This chapter describes the A80 display system from following perspectives:

- Display engine system
- TCON
- DEFE
- DEBE
- CMU
- DEU
- DRC

## 7.1. Display Engine System

### 7.1.1. Display Engine Block Diagram



**Display Subsystem Main Data Flow**

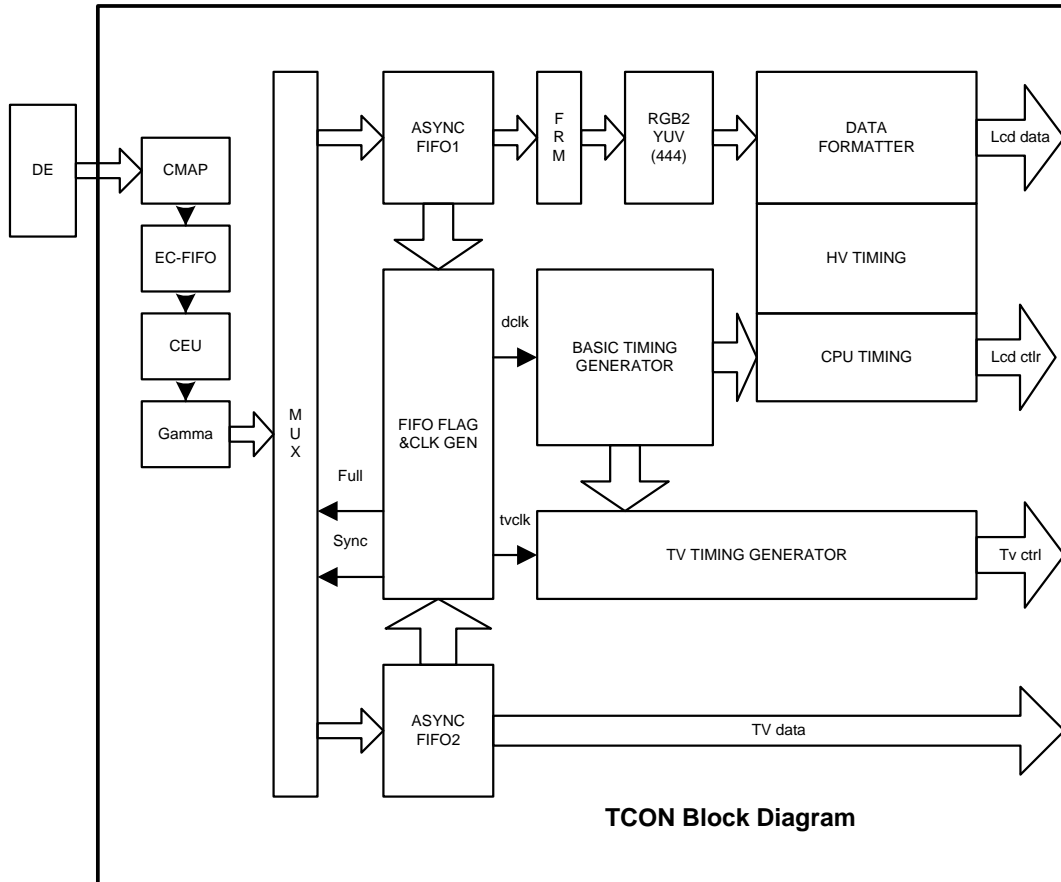
## 7.2. TCON

### 7.2.1. Overview

The LCD0 module is used for LCD panel, and LCD1 module is used for HDTV.

- LCD0 supports LVDS interface with single/dual link, up to 1920x1200@60fps
- LCD0 supports RGB interface with DE/SYNC mode, up to 2048x1536@60fps
- LCD0 supports serial RGB/dummy RGB/CCIR656 interface, up to 1280x720@60fps
- LCD0 supports i80 interface with 18/16/9/8 bit, support TE, up to 1280x720@60fps
- LCD0 supports pixel format: RGB888, RGB666 and RGB565
- LCD0 dither function from RGB666/RGB565 to RGB888
- LCD0 color map function and color enhance function
- LCD0 gamma correction with R/G/B channel independence
- LCD1 supports up to full HDTV timing for HDMI
- LCD1 gamma correction with R/G/B channel independence

7.2.2. Block Diagram



TCON Block Diagram

### 7.2.3. TCON Structure

#### 7.2.3.1. Input Port and Stream FIFO

Main Signal	I/O Type	Definition And Description
RGB[23..0]	I	Digital RGB data input from the host(three channels)
Full	O	Full signal from input FIFO(three channels)
SRC_SYN	O	Source reset signal(two channels)
SRC_STA		When it asserts, the corresponding FIFO and counter will be clear

#### 7.2.3.2. FIFO1/FIFO2 Attribute:

Type: Async Width: 8bit\*3 Depth: 64

LCD controller has 2 input sources: DE0 and DE1. Note that FIFO1 and FIFO2 can select the same input source. The 24-bit width FIFO is the bridge between host and the timing generator. When the data is out by the host, it will be stored in the FIFO. And when the timing generator wants to drives the next pixel, data will be fetched from FIFO. The key function of the FIFO is to guarantee the synchronization between input stream and the output pixels.

Generally, the host can encode graphic data faster than the timing controller can accept, so it has to make a handshake (assert a full flag) to decelerate the host data transmission speed.

In a worse case, too much traffic may cause the host to not encode the graphic data fast enough, a FIFO underflow happens. Each time the timing generator fetches next pixel data, it can only get the last valid one. At the same time, a 19-bit counter is used to record how many data is missed. Later, FIFO has to discard enough input data to make the 19-bit counter decrease to zero. If the counter value reaches the number of a whole frame pixel, it will reset to zero.

Each input channel has a full flag to prevent FIFO overflow, and its logic as:

When FIFO is full, full flag will assert.

When the source is not selected by FIFO, or FIFO in disable state, full flag always asserts.

During source reset timing (LCD/TV set a reset signal to its source), FIFO full flag asserts.

If FIFO1 and FIFO2 select the same source, full flag is their combine logic.



When SRC\_SYN is asserted, the corresponding FIFO data are discarded, and the counter will be reset. SRC\_STA will be valid at back porch time every frame. This signal tell the DE to start decode data for display.

### 7.2.3.3. Frame Synchronization

SRC\_SYN is used to sync the input source. When it's asserted, input drivers (DE ) should terminate the current transfer and begin at the first pixel data of the display frame(interlace mode should begin at odd field).

Reset signal will assert at the beginning of first blank line every field by LCD timing controller.  
Reset signal will assert at the beginning of first blank line every odd field by TV timing controller.

There is 1 SRC\_SYN signal for each input source, and its logic is as:

If source is not used, SRC\_SYN is always invalid.

If source outputs to LCD, its SRC\_SYN signal is drive by LCD timing controller; if the source outputs to TV, its SRC\_SYN signal is drive by TV timing controller.

If LCD and TV select the same source or both LCD and TV select DE as input source, its SRC\_SYN is valid only when both LCD and TV assert its own reset signal.

### 7.2.3.4. DE\_TCON\_BRIDGE

This module connects DE output and LCD CH1 input, it buffer it input pixels and output pixel data by interlace or no-interlace mode.

Signal	I/O Type	Definition And Description
DECLK	I	Clock
SYNC	I	Reset signal for buffer
WE1	I	Write signal from source
READY1	O	Ready signal for source
WE2	O	Write signal for CH1 input
READY2	O	Ready signal from CH1
INTERLACE_EN	I	Interlace output enable
FIELD	I	Output field request
SIZEX	I	Source picture X axis size
DI[23..0]	I	Input data bus

DO[23..0]	O	Output data bus
-----------	---	-----------------

#### 7.2.3.5. Function description

This module acts as a 768x8x3 FIFO for DE1 data output cache and TCON CH1 input buffer.

When "INTERLACE\_EN" is true, this FIFO discards every odd/even line according to the "FIELD" signal.

If "FIELD" is true, FIFO discards all even line data;

If "FIELD" is false, FIFO discards all odd line data;

#### 7.2.3.6. Clock Generator

This module is used to generate DCLK for LCD .

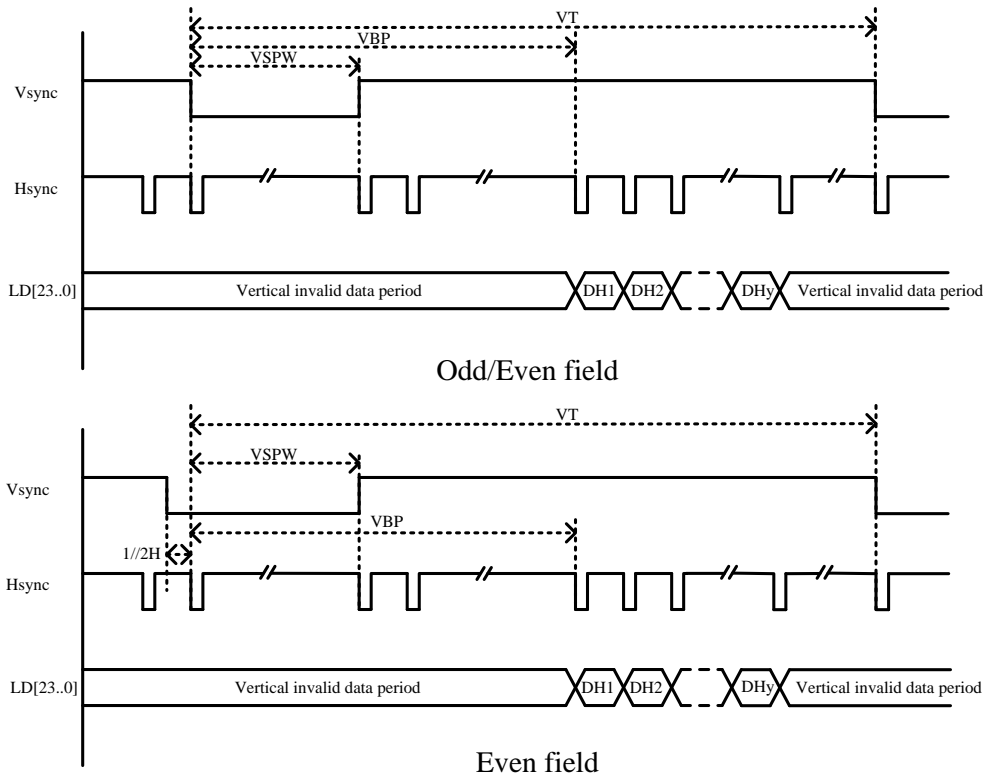
#### 7.2.4. HV interface

HV I/F is also known as Sync + DE mode, which is widely used in TFT LCD module for PMP/MP4 applications.

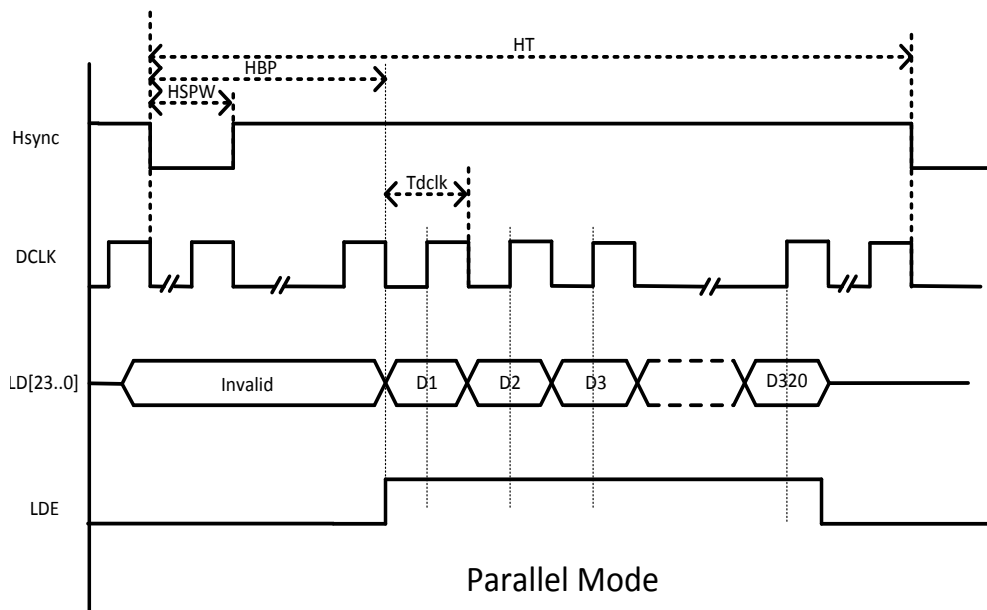
Its signals are defined as:

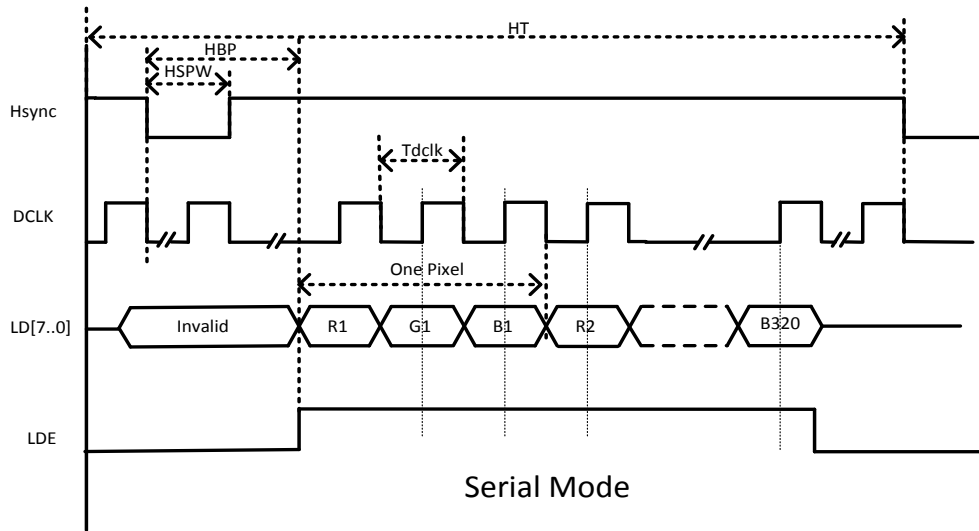
Main Signal	I/O Type	Definition And Description
VSYNC	O	Vertical sync, indicates one new frame
HSYNC	O	Horizontal sync, indicate one new scan line
DCLK	O	Dot clock, pixel data are sync by this clock
LDE	O	LCD data enable
LD[23..0]	O	18Bit RGB/YUV output from input FIFO for panel

Vertical Timing



Horizontal Timing





### 7.2.5. CCIR interface

When in HV serial YUV output mode, its timing is CCIR656/601 compatible. SAV add right before active area every line; EAV add right after active area every line.

Its logic is:

F = "0" for Field 1 F = "1" for Field 2;

V = "1" during vertical blanking, V="0" during active video;

H = "0" at SAV H = "1" at EAV;

P3-P0 = protection bits

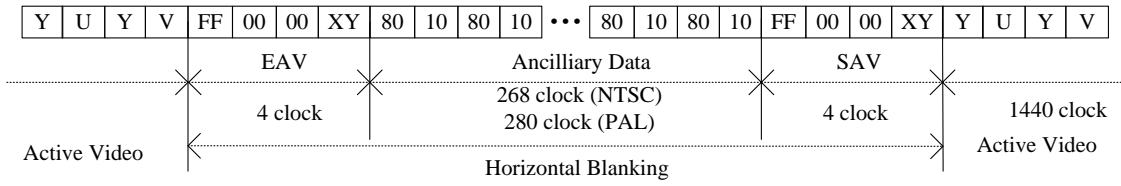
$P3 = V \oplus H$     $P2 = F \oplus H$     $P1 = F \oplus V$     $P0 = F \oplus V \oplus H$

Where  $\oplus$  represents the exclusive-OR function

The 4-byte SAV/EAV sequences are:

	8-bit Data							
	D7 (MSB)	D6	D5	D4	D3	D2	D1	D0 (LSB)
preamble	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
Status word	1	F	V	H	P3	P2	P1	P0

Panel interface Timing:

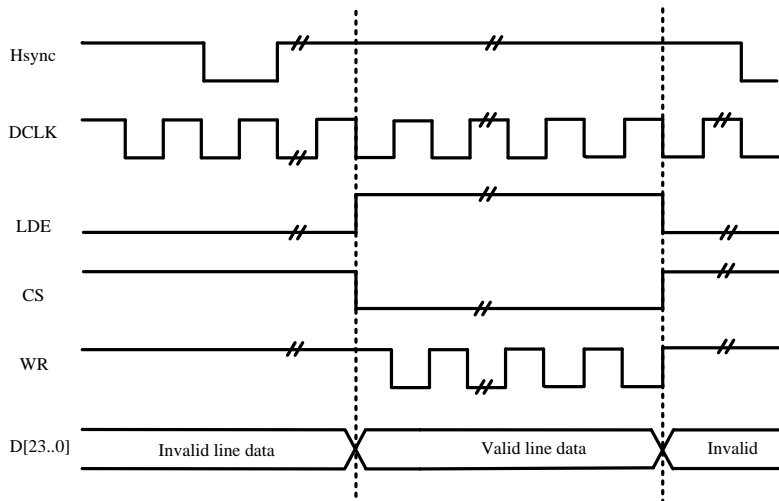


### 7.2.6. CPU interface

CPU I/F LCD panel is most common interface for small size, low resolution LCD panels.

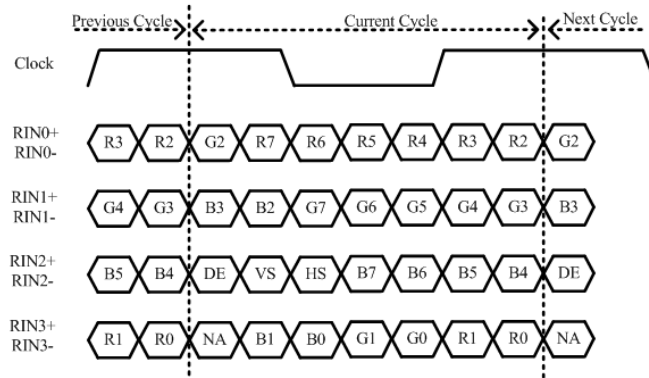
Main Signal	I/O type	Definition And Description
CS	O	Chip select, active low
WR	O	Write strobe, active low
RD	O	Read strobe, active low
A1	O	Address bit, controlled by "LCD_CPU I/F" BIT26/25
D[23..0]	I/O	Digital RGB output signal

The following figure illustrates the relationship between basic timing and CPU timing. When CPU I/F is in IDLE state, it can generate WR/RD timing by setting "LCD\_CPU I/F". WR is 180 degree delay of DCLK; CS is active when pixel data are valid; RD is always set to 1; A1 are set by "LCD\_CPU I/F".

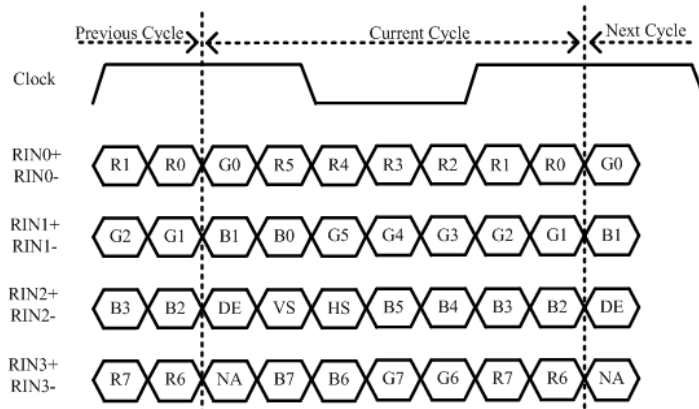


## 7.2.7. LVDS interface

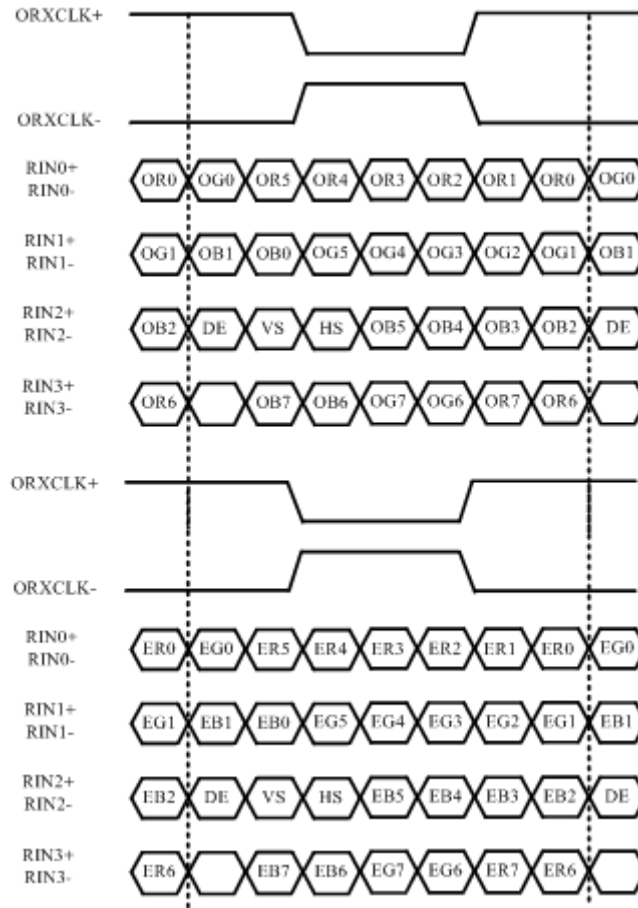
### 7.2.7.1. Single channel: JEDIA mode



### 7.2.7.2. Single channel: NS mode



**7.2.7.3. Dual Channels: NS Mode**



**7.2.7.4. Control signal and data port mapping**

Please refer to [Appendix](#).

**7.2.7.5. Gamma correction**

This module correct the RGB input data of DE.

A 256x8x3 Byte register file is used to store the gamma table. The following is the layout.

Offset	Value
0x400,0x401,0x402	{B0[7:0],G0[7:0],R0[7:0]}
0x404,0x405,0x406	{B1[7:0],G1[7:0],R1[7:0]}
.....	.....
0x4FC,0x4FD,0xFE	{B255[7:0],G255[7:0],R255[7:0]}

### 7.2.7.6. Color Enhance Unit

Function: This module enhance color data from DE0.

$$R_O = R_r R_i + R_g G_i + R_b B_i + R_c \quad (R_{\min} \leq R_O \leq R_{\max})$$

$$G_O = G_r R_i + G_g G_i + G_b B_i + G_c \quad (G_{\min} \leq G_O \leq G_{\max})$$

$$B_O = B_r R_i + B_g G_i + B_b B_i + B_c \quad (B_{\min} \leq B_O \leq B_{\max})$$

note:

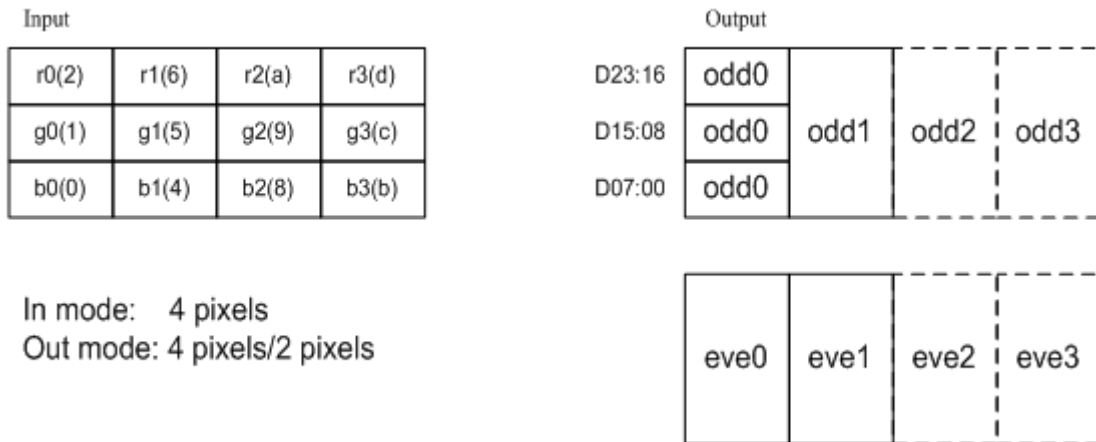
$$R_r, R_g, R_b, G_r, G_g, G_b, B_r, B_g, B_b \quad \text{is} \quad s13(-16,16)$$

$$R_c, G_c, B_c \quad \text{is} \quad s19(-16384,16384)$$

### 7.2.7.7. Color Mapping Unit

This module map color data from DE0

Every 4 input pixels as an unit. an unit is divided into 12 bytes. Output byte can select one of those 12 bytes. Note that even line and odd line can be different, and output can be 12 bytes(4 pixels) or reduce to 6 bytes(2 pixels).





### 7.2.8. TCON Register List

Module Name	Base Address
TCON0	0x03C00000

Register Name	Offset	Description
TCON_GCTL_REG	0x000	TCON global control register
TCON_GINT0_REG	0x004	TCON global interrupt register0
TCON_GINT1_REG	0x008	TCON global interrupt register1
TCON0_FRM_CTL_REG	0x010	TCON FRM control register
TCON0_CTL_REG	0x040	TCON0 control register
TCON0_DCLK_REG	0x044	TCON0 data clock register
TCON0_BASIC0_REG	0x048	TCON0 basic timing register0
TCON0_BASIC1_REG	0x04C	TCON0 basic timing register1
TCON0_BASIC2_REG	0x050	TCON0 basic timing register2
TCON0_BASIC3_REG	0x054	TCON0 basic timing register3
TCON0_HV_IF_REG	0x058	TCON0 HV panel interface register
TCON0_CPU_IF_REG	0x060	TCON0 CPU panel interface register
TCON0_CPU_WR_REG	0x064	TCON0 CPU panel write data register
TCON0_CPU_RD0_REG	0x068	TCON0 CPU panel read data register0
TCON0_CPU_RD1_REG	0x06C	TCON0 CPU panel read data register1
TCON0_LVDS_IF_REG	0x084	TCON0 LVDS panel interface register
TCON0_IO_POL_REG	0x088	TCON0 IO polarity register
TCON0_IO_TRI_REG	0x08C	TCON0 IO control register
TCON_ECC_FIFO_REG	0x0F8	TCON ECC FIFO register
TCON_CEU_CTL_REG	0x100	TCON CEU control register

TCON_CEU_COEF_MUL_REG	$0x110+N*0x04$	TCON CEU coefficient register0  (N=0,1,2,4,5,6,8,9,10)
TCON_CEU_COEF_ADD_REG	$0x11C+N*0x10$	TCON CEU coefficient register1  (N=0,1,2)
TCON_CEU_COEF_RANG_REG	$0x140+N*0x04$	TCON CEU coefficient register2  (N=0,1,2)
TCON0_CPU_TRI0_REG	0x160	TCON0 CPU panel trigger register0
TCON0_CPU_TRI1_REG	0x164	TCON0 CPU panel trigger register1
TCON0_CPU_TRI2_REG	0x168	TCON0 CPU panel trigger register2
TCON0_CPU_TRI3_REG	0x16C	TCON0 CPU panel trigger register3
TCON0_CPU_TRI4_REG	0x170	TCON0 CPU panel trigger register4
TCON0_CPU_TRI5_REG	0x174	TCON0 CPU panel trigger register5
TCON_CMAP_CTL_REG	0x180	TCON color map control register
TCON_CMAP_ODD0_REG	0x190	TCON color map odd line register0
TCON_CMAP_ODD1_REG	0x194	TCON color map odd line register1
TCON_CMAP_EVEN0_REG	0x198	TCON color map even line register0
TCON_CMAP_EVEN1_REG	0x19C	TCON color map even line register1

### 7.2.9. TCON0 Register Description

#### 7.2.9.1. TCON global control register

Offset: 0x000			Register Name: TCON_GCTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON0_En  0: disable  1: enable  When it's disabled, the module will be reset to idle state.
30	R/W	0	TCON0_Gamma_En  0: disable  1: enable
29:1	/	/	/
0	R/W	0	Reserved

#### 7.2.9.2. TCON global interrupt register0

Offset: 0x004			Register Name: TCON_GINT0_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON0_Vb_Int_En  0: disable  1: enable

30	R/W	0	TCON1_Vb_Int_En  0: disable  1: enable
29	R/W	0	TCON0_Line_Int_En  0: disable  1: enable
28	R/W	0	TCON1_Line_Int_En  0: disable  1: enable
27	R/W	0	TCON0_Tri_Finish_Int_En  0: disable  1: enable
26:	R/W	0	TCON0_Tri_Counter_Int_En  0: disable  1: enable
25:16	/	/	/
15	R/W	0	TCON0_Vb_Int_Flag  Assert at inactive area.  Write 0 to clear it.
14	R/W	0	TCON1_Vb_Int_Flag  Assert at inactive area.

			Write 0 to clear it.
13	R/W	0	TCON0_Line_Int_Flag  Assert when SY0 match the current TCON0 scan line.  Write 0 to clear it.
12	R/W	0	TCON1_Line_Int_Flag  trigger when SY1 match the current TCON1 scan line  Write 0 to clear it.
11	R/W	0	TCON0_Tri_Finish_Int_Flag  trigger when CPU trigger mode finish  Write 0 to clear it.
10	R/W	0	TCON0_Tri_Counter_Int_Flag  trigger when tri counter reach this value  Write 0 to clear it.
9	R/W	0	TCON0_Tri_Underflow_Flag  only used in DSI video mode, tri when sync by DSI but not finish  Write 0 to clear it.
8:0	/	/	/

### 7.2.9.3. TCON global interrupt register1

Offset: 0x008			Register Name: TCON_GINT1_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/

27:16	R/W	0	<p>TCON0_Line_Int_Num</p> <p>Scan line for TCON0 line interrupt (inactive lines are calculated).</p> <p>Note: SY0 is writable only when LINE_TRG0 disable.</p>
15:12	/	/	/
11:0	R/W	0	<p>TCON1_Line_Int_Num</p> <p>Scan line for TCON1 line interrupt (inactive lines are calculated).</p> <p>Note: SY1 is writable only when LINE_TRG1 disable.</p>

#### 7.2.9.4. TCON FRM control register

Offset: 0x010			Register Name: TCON0_FRM_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	<p>TCON0_FRM_En</p> <p>0:disable</p> <p>1:enable</p>
30:7	/	/	/
6	R/W	0	<p>TCON0_FRM_Mode_R</p> <p>0: 6bit FRM output</p> <p>1: 5bit FRM output</p>
5	R/W	0	<p>TCON0_Frm_Mode_G</p> <p>0: 6bit FRM output</p> <p>1: 5bit FRM output</p>

4	R/W	0	TCON0_Frm_Mode_B  0: 6bit FRM output  1: 5bit FRM output
3:2	/	/	/
1:0	R/W	0	TCON0_FRM_Test  00: FRM  01: half 5/6bit, half FRM  10: half 8bit, half FRM  11: half 8bit, half 5/6bit

#### 7.2.9.5. TCON0 control register

Offset: 0x040			Register Name: TCON0_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON0_En  0: disable  1: enable  Note: It executes at the beginning of the first blank line of TCON0 timing.
30:29	/	/	/
28	R/W	0	TCON0_Work_Mode  0: normal  1: dynamic freq

27:26	/	/	/
25:24	R/W	0	<p>TCON0_IF</p> <p>00: HV(Sync+DE)</p> <p>01: 8080 I/F</p> <p>1x:reserved</p>
23	R/W	0	<p>TCON0_RB_Swap</p> <p>0: default</p> <p>1: swap RED and BLUE data at FIFO1</p>
22	R/W	0	reserved
21	R/W	0	<p>TCON0_FIFO1_Rst</p> <p>Write 1 and then 0 at this bit will reset FIFO 1</p> <p>Note: 1 holding time must more than 1 DCLK</p>
20	R/W	0	reserved
19:9	/	/	/
8:4	R/W	0	<p>TCON0_Start_Delay</p> <p>STA delay</p> <p>NOTE: valid only when TCON0_EN == 1</p>
3	/	/	/
2:0	R/W	0	<p>TCON0_SRC_SEL:</p> <p>000: DE0</p> <p>001: reserved</p>



			010: reserved
			011: reserved
			100: Test Data all 0
			101: Test Data all 1
			11x: reserved

#### 7.2.9.6. TCON0 data clock register

Offset: 0x044			Register Name: TCON0_DCLK REG
Bit	Read/ Write	Default/ Hex	Description
31:28	R/W	0	<p>TCON0_Dclk_En</p> <p>LCLK_EN[3:0] :TCON0 clock enable</p> <p>4'h0, 'h4,4'h6,4'ha7:dclk_en=0;dclk1_en=0;dclk2_en=0;dclkm2_en=0;</p> <p>4'h1: dclk_en = 1; dclk1_en = 0; dclk2_en = 0; dclkm2_en = 0;</p> <p>4'h2: dclk_en = 1; dclk1_en = 0; dclk2_en = 0; dclkm2_en = 1;</p> <p>4'h3: dclk_en = 1; dclk1_en = 1; dclk2_en = 0; dclkm2_en = 0;</p> <p>4'h5: dclk_en = 1; dclk1_en = 0; dclk2_en = 1; dclkm2_en = 0;</p> <p>4'h8,4'h9,4'ha,4'hb,4'hc,4'hd,4'he,4'hf:</p> <p style="padding-left: 40px;">dclk_en = 1;</p> <p style="padding-left: 40px;">dclk1_en = 1;</p>

			$dclk2\_en = 1;$  $dclkm2\_en = 1;$
27:7	/	/	/
6:0	R/W	0	TCON0_Dclk_Div  $Tdclk = Tsclk * DCLKDIV$  Note:  1.if dclk1&dclk2 used, DCLKDIV >=6  2.if dclk only, DCLKDIV >=1

#### 7.2.9.7. TCON0 basic timing register0

Offset: 0x048			Register Name: TCON0_BASIC0_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27:16	R/W	0	TCON0_X  Panel width is X+1
15:12	/	/	/
11:0	R/W	0	TCON0_Y  Panel height is Y+1

#### 7.2.9.8. TCON0 basic timing register1

Offset: 0x04C			Register Name: TCON0_BASIC1_REG
Bit	Read/ Write	Default/ Hex	Description

31	R/W	0	Reserved
30:29	/	/	/
28:16	R/W	0	<p>HT</p> <p><math>T_{ht} = (HT+1) * T_{dclk}</math></p> <p>Computation</p> <p>1) parallel: <math>HT = X + BLANK</math></p> <p>Limitation:</p> <p>1) parallel : <math>HT \geq (HBP + 1) + (X+1) + 2</math></p> <p>2) serial 1: <math>HT \geq (HBP + 1) + (X+1) * 3 + 2</math></p> <p>3) serial 2: <math>HT \geq (HBP + 1) + (X+1) * 3/2 + 2</math></p>
15:12	/	/	/
11:0	R/W	0	<p>HBP</p> <p>horizontal back porch (in dclk)</p> <p><math>T_{hbp} = (HBP + 1) * T_{dclk}</math></p>

### 7.2.9.9. TCON0 basic timing register2

Offset: 0x050			Register Name: TCON0_BASIC2_REG
Bit	Read/ Write	Default/ Hex	Description
31:29	/	/	/
28:16	R/W	0	<p>VT</p> <p><math>TVT = (VT)/2 * T_{hsync}</math></p>

			Note: $VT/2 \geq (VBP+1) + (Y+1) + 2$
15:12	/	/	/
11:0	R/W	0	VBP  $T_{vbp} = (VBP + 1) * T_{hsync}$

### 7.2.9.10. TCON0 basic timing register3

Offset: 0x054			Register Name: TCON0_BASIC3_REG
Bit	Read/ Write	Default/ Hex	Description
31:26	/	/	/
25:16	R/W	0	HSPW  $T_{hspw} = (HSPW+1) * T_{dclk}$  Note: $HT > (HSPW+1)$
15:10	/	/	/
9:0	R/W	0	VSPW  $T_{vspw} = (VSPW+1) * T_{hsync}$  Note: $VT/2 > (VSPW+1)$

### 7.2.9.11. TCON0 HV panel interface register

Offset: 0x058			Register Name: TCON0_HV_IF_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	R/W	0	HV_Mode  0000: 24bit/1cycle parallel mode

			1000: 8bit/3cycle RGB serial mode(RGB888)  1010: 8bit/4cycle Dummy RGB(DRGB)  1011: 8bit/4cycle RGB Dummy(RGBD)  1100: 8bit/2cycle YUV serial mode(CCIR656)
27:26	R/W	0	RGB888_SM0  Serial RGB888 mode Output sequence at odd lines of the panel (line 1, 3, 5, 7...)  00: R→G→B  01: B→R→G  10: G→B→R  11: R→G→B
25:24	R/W	0	RGB888_SM1  Serial RGB888 mode Output sequence at even lines of the panel (line 2, 4, 6, 8...)  00: R→G→B  01: B→R→G  10: G→B→R  11: R→G→B
23:22	R/W	0	YUV_SM  serial YUV mode Output sequence 2-pixel-pair of every scan line  00: YUYV

			01: YVYU  10: UYVY  11: VYUY
21:20	R/W	0	YUV EAV/SAV F line delay  0:F toggle right after active video line  1:delay 2 line(CCIR PAL)  2:delay 3 line(CCIR NTSC)  3:reserved
19:0	/	/	/

#### 7.2.9.12. TCON0 CPU panel interface register

Offset: 0x060			Register Name: TCON0_CPU_IF_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	R/W	0	CPU_Mode  0000: 18bit/256K mode  0010: 16bit mode0  0100: 16bit mode1  0110: 16bit mode2  1000: 16bit mode3  1010: 9bit mode

			<p>1100: 8bit 256K mode</p> <p>1110: 8bit 65K mode</p> <p>xxx1: 24bit for DSI</p>
27	/	/	/
26	R/W	0	<p>DA</p> <p>pin A1 value in 8080 mode auto/flash states</p>
25	R/W	0	<p>CA</p> <p>pin A1 value in 8080 mode WR/RD execute</p>
24	R/W	0	Reserved
23	R	0	<p>Wr_Flag</p> <p>0:write operation is finishing</p> <p>1:write operation is pending</p>
22	R	0	<p>Rd_Flag</p> <p>0:read operation is finishing</p> <p>1:read operation is pending</p>
21:18	/	/	/
17	R/W	0	<p>AUTO</p> <p>auto Transfer Mode:</p> <p>If it's 1, all the valid data during this frame are write to panel.</p> <p>Note: This bit is sampled by VYSNC</p>
16	R/W	0	FLUSH

			<p>direct transfer mode:</p> <p>If it's enabled, FIFO1 is regardless of the HV timing, pixels data keep being transferred unless the input FIFO was empty.</p> <p>Data output rate control by DCLK.</p>
15:6	/	/	/
5:4	R/W	0	<p>Trigger_SYNC_Mode</p> <p>0: start frame flush immediately by bit1.</p> <p>1: start frame flush sync to TE PIN. rising by bit1.</p> <p>2: start frame flush sync to TE PIN. falling by bit1.</p> <p>When set as 1 or 2, io0 is map as TE input.</p>
3	R/W	0	<p>Trigger_FIFO_BIST_En</p> <p>0: disable</p> <p>1: enable</p> <p>Entry address is 0xFF8</p>
2	R/W	0	<p>Trigger_FIFO_En</p> <p>0:enable</p> <p>1:disable</p>
1	R/W	0	<p>Trigger_Start</p> <p>Write '1' to start a frame flush, write'0' has no effect.</p> <p>This flag indicated frame flush is running.</p> <p>Software must make sure write '1' only when this flag is '0'.</p>



0	R/W	0	Trigger_En  0: trigger mode disable  1: trigger mode enable
---	-----	---	---

#### 7.2.9.13. TCON0 CPU panel write data register

Offset: 0x064			Register Name: TCON0_CPU_WR_REG
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/
23:0	W	0	Data_Wr  Data write on 8080 bus, launch a write operation on i8080 bus.

#### 7.2.9.14. TCON0 CPU panel read data register0

Offset: 0x068			Register Name: TCON0_CPU_RDO_REG
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/
23:0	R	/	Data_Rd0  Data read on 8080 bus, launch a new read operation on 8080 bus.

#### 7.2.9.15. TCON0 CPU panel read data register1

Offset: 0x06C			Register Name: TCON0_CPU_RD1_REG
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/

23:0	R	/	Data_Rd1  Data read on 8080 bus, without a new read operation on 8080 bus.
------	---	---	--

#### 7.2.9.16. TCON0 LVDS panel interface register

Offset: 0x084			Register Name: TCON0_LVDS_IF_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON0_LVDS_En  0: disable  1: enable
30	R/W	0	TCON0_LVDS_Link_Sel  0: single link  1: dual link
29	R/W	0	TCON0_LVDS_Even_Odd_Dir  0: normal  1: reverse
28	R/W	0	TCON0_LVDS_Dir  1: normal  2: reverse  NOTE: LVDS direction
27	R/W	0	TCON0_LVDS_Mode

			0: NS mode 1: JEIDA mode
26	R/W	0	TCON0_LVDS_BitWidth 0: 24bit 1: 18bit
25	R/W	0	TCON0_LVDS_DeBug_En 0: disable 1: enable
24	R/W	0	TCON0_LVDS_DeBug_Mode 0: mode0 1: mode1
23	R/W	0	TCON0_LVDS_Correct_Mode 0: mode0 1: mode1
22:21	/	/	/
20	R/W	0	TCON0_LVDS_Clk_Sel 0: MIPI PLL 1: TCON0 CLK
19:5	/	/	/
4	R/W	0	TCON0_LVDS_CLK_Polarity 0: reverse

			1: normal
3:0	R/W	0	TCON0_LVDS_Data_Polarity  0: reverse  1: normal

### 7.2.9.17. TCON0 IO polarity register

Offset: 0x088			Register Name: TCON0_IO_POL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	IO_Output_Sel  0: normal output  1: register output  when set as '1', d[23:0], io0, io1,io3 sync to dclk
30:28	R/W	0	DCLK_Sel  000: used DCLK0(normal phase offset)  001: used DCLK1(1/3 phase offset)  010: used DCLK2(2/3 phase offset)  101: DCLK0/2 phase 0  100: DCLK0/2 phase 90  reserved  6phase: 00-14-20-04-10-24

27	R/W	0	IO3_Inv  0: not invert  1: invert
26	R/W	0	IO2_Inv  0: not invert  1: invert
25	R/W	0	IO1_Inv  0: not invert  1: invert
24	R/W	0	IO0_Inv  0: not invert  1: invert
23:0	R/W	0	Data_Inv  TCON0 output port D[23:0] polarity control, with independent bit control:  0s: normal polarity  1s: invert the specify output

#### 7.2.9.18. TCON0 IO control register

Offset: 0x08C			Register Name: TCON0_IO_TRI_REG
Bit	Read/ Write	Default/ Hex	Description
31:29	/	/	/

28	/	/	<p>RGB_Endian</p> <p>0: normal</p> <p>1: bits invert</p>
27	R/W	1	<p>IO3_Output_Tri_En</p> <p>1: disable</p> <p>0: enable</p>
26	R/W	1	<p>IO2_Output_Tri_En</p> <p>1: disable</p> <p>0: enable</p>
25	R/W	1	<p>IO1_Output_Tri_En</p> <p>1: disable</p> <p>0: enable</p>
24	R/W	1	<p>IO0_Output_Tri_En</p> <p>1: disable</p> <p>0: enable</p>
23:0	R/W	0xFFFFFFFF	<p>Data_Output_Tri_En</p> <p>TCON0 output port D[23:0] output enable, with independent bit control:</p> <p>1s: disable</p> <p>0s: enable</p>

**7.2.9.19. TCON ECC FIFO register**

Offset: 0x0F8			Register Name: TCON_ECC_FIFO_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	/	ECC_FIFO_BIST_EN  0: disable  1: enable
30	R/W	/	ECC_FIFO_ERR_FLAG
29:24	/	/	/
23:16	R/W	/	ECC_FIFO_ERR_BITS
15:9	/	/	/
8	R/W	/	ECC_FIFO_BLANK_EN  0: disable ECC function in blanking  1: enable ECC function in blanking  ECC function is tent to trigged in blanking area at hv mode, set '0' when in hv mode
7:0	R/W	/	ECC_FIFO_SETTING  Note: bit3 0 enable, 1 disable

**7.2.9.20. TCON CEU control register**

Offset: 0x100			Register Name: TCON_CEU_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	CEU_En

			0: bypass 1: enable
30:0	/	/	/

### 7.2.9.21. TCON CEU coefficient register0

Offset: 0x110+N*0x04 (N=0,1,2,4,5,6,8,9,10)			Register Name: TCON_CEU_COEF_MUL_REG
Bit	Read/ Write	Default/ Hex	Description
31:13	/	/	/
12:0	R/W	0	CEU_Coef_Mul_Value  signed 13bit value, range of (-16,16)  N=0: Rr  N=1: Rg  N=2: Rb  N=4: Gr  N=5: Gg  N=6: Gb  N=8: Br  N=9: Bg  N=10: Bb



**7.2.9.22. TCON CEU coefficient register1**

Offset: 0x11C+N*0x10  (N=0,1,2)			Register Name: TCON_CEU_COEF_ADD_REG
Bit	Read/ Write	Default/ Hex	Description
31:19	/	/	/
18:0	R/W	0	CEU_Coef_Add_Value  signed 19bit value, range of (-16384, 16384)  N=0: Rc  N=1: Gc  N=2: Bc

**7.2.9.23. TCON CEU coefficient register2**

Offset: 0x140+N*0x04  (N=0,1,2)			Register Name: TCON_CEU_COEF_RANG_REG
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/
23:16	R/W	0	CEU_Coef_Range_Min  unsigned 8bit value, range of [0,255]
15:8	/	/	/
7:0	R/W	0	CEU Coef_Range_Max  unsigned 8bit value, range of [0,255]

#### 7.2.9.24. TCON0 CPU panel trigger register0

Offset: 0x160			Register Name: TCON0_CPU_TRI0_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27:16	R/W	0	Block_Space  should be set >20*pixel_cycle
15:12	/	/	/
11:0	R/W	0	Block_Size

#### 7.2.9.25. TCON0 CPU panel trigger register1

Offset: 0x164			Register Name: TCON0_CPU_TRI1_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R	0	Block_Current_Num
15:0	R/W	0	Block_Num

#### 7.2.9.26. TCON0 CPU panel trigger register2

Offset: 0x168			Register Name: TCON0_CPU_TRI2_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0x20	Start_Delay  $Tdly = (Start\_Delay + 1) * be\_clk * 8$
15	R/W	0	Trans_Start_Mode  0: ecc_fifo+tri_fifo

			1: tri_fifo
14:13	R/W	0	Sync_Mode  0x: auto  10: 0  11: 1
12:0	R/W	0	Trans_Start_Set

### 7.2.9.27. TCON0 CPU panel trigger register3

Offset: 0x16C			Register Name: TCON0_CPU_TRIS_REG
Bit	Read/ Write	Default/ Hex	Description
31:30	/	/	/
29:28	R/W	0	Tri_Int_Mode  00: disable  01: counter mode  10: te rising mode  11: te falling mode  when set as 01, Tri_Counter_Int occur in cycle of $(Count\_N+1) \times (Count\_M+1) \times 4$ dclk.  when set as 10 or 11, io0 is map as TE input.
27:24	/	/	/
23:8	R/W	0	Counter_N
7:0	R/W	0	Counter_M

**7.2.9.28. TCON0 CPU panel trigger register4**

Offset: 0x170			Register Name: TCON0_CPU_TRI4_REG
Bit	Read/ Write	Default/ Hex	Description
31:29	/	/	
28	R/W	0	Plug_Mode_En  0: disable  1:enable
27:25	/	/	/
24	R/W	0	A1  Valid in first Block
23:0	R/W	0	D23-D0  Valid in first Block

**7.2.9.29. TCON0 CPU panel trigger register5**

Offset: 0x174			Register Name: TCON0_CPU_TRI5_REG
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/
24	R/W	0	A1  Valid in Block except first
23:0	R/W	0	D23-D0  Valid in Block except first

### 7.2.9.30. TCON color map control register

Offset: 0x180			Register Name: TCON_CMAP_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	Color_Map_En  0: bypass  1: enable  This module only work when X is divided by 4
30:1	/	/	/
0	R/W	0	Out_Format  0: 4 pixel output mode: Out0 -> Out1 -> Out2 -> Out3  1: 2 pixel output mode: Out0 -> Out1

### 7.2.9.31. TCON color map odd line register0

Offset: 0x190			Register Name: TCON_CMAP_ODD0_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0	Out_Odd1
15:0	R/W	0	Out_Odd0  bit15-12: Reserved  bit11-08: Out_Odd0[23:16]  bit07-04: Out_Odd0[15:8]

			bit03-00: Out_Odd0[7:0]  0x0: in_b0  0x1: in_g0  0x2: in_r0  0x3: reserved  0x4: in_b1  0x5: in_g1  0x6: in_r1  0x7: reserved  0x8: in_b2  0x9: in_g2  0xa: in_r2  0xb: reserved  0xc: in_b3  0xd: in_g3  0xe: in_r3  0xf: reserved
--	--	--	---

**7.2.9.32. TCON color map odd line register1**

Offset: 0x194			Register Name: TCON_CMAP_ODD1_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0	Out_Odd3
15:0	R/W	0	Out_Odd2

**7.2.9.33. TCON color map even line register0**

Offset: 0x198			Register Name: TCON_CMAP_EVEN0_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0	Out_Even1
15:0	R/W	0	Out_Even0

**7.2.9.34. TCON color map even line register1**

Offset: 0x19C			Register Name: TCON_CMAP_EVEN1_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0	Out_Even3
15:0	R/W	0	Out_Even2

### 7.2.10. TCON1 Register List

Module Name	Base Address
TCON1	0x03C10000

Register Name	Offset	Description
TCON_GCTL_REG	0x000	TCON global control register
TCON_GINT0_REG	0x004	TCON global interrupt register0
TCON_GINT1_REG	0x008	TCON global interrupt register1
TCON1_CTL_REG	0x090	TCON1 control register
TCON1_BASIC0_REG	0x094	TCON1 basic timing register0
TCON1_BASIC1_REG	0x098	TCON1 basic timing register1
TCON1_BASIC2_REG	0x09C	TCON1 basic timing register2
TCON1_BASIC3_REG	0x0A0	TCON1 basic timing register3
TCON1_BASIC4_REG	0x0A4	TCON1 basic timing register4
TCON1_BASIC5_REG	0x0A8	TCON1 basic timing register5
TCON1_PS_SYNC_REG	0x0B0	TCON1 sync register
TCON1_IO_POL_REG	0x0F0	TCON1 IO polarity register
TCON1_IO_TRI_REG	0x0F4	TCON1 IO control register
TCON_ECC_FIFO_REG	0x0F8	TCON ECC FIFO register
TCON_CEU_CTL_REG	0x100	TCON CEU control register
TCON_CEU_COEF_MUL_REG	0x110+N*0x04	TCON CEU coefficient register0  (N=0,1,2,4,5,6,8,9,10)
TCON_CEU_COEF_ADD_REG	0x11C+N*0x10	TCON CEU coefficient register1  (N=0,1,2)



TCON_CEU_COEF_RANG_REG	$0x140+N*0x04$	TCON CEU coefficient register2  (N=0,1,2)
------------------------	----------------	---

## 7.2.11. TCON 1 Register Description

### 7.2.11.1. TCON global control register

Offset: 0x000			Register Name: TCON_GCTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON_En  0: disable  1: enable  When it's disabled, the module will be reset to idle state.
30	R/W	0	TCON_Gamma_En  0: disable  1: enable
29:1	/	/	/
0	R/W	0	IO_Map_Sel  0: TCON0  1: TCON1  Note: this bit determines which IO_INV/IO_TRI are valid

### 7.2.11.2. TCON global interrupt register0

Offset: 0x004			Register Name: TCON_GINT0_REG
Bit	Read/ Write	Default/ Hex	Description

	Write	Hex	
31	R/W	0	TCON0_Vb_Int_En  0: disable  1: enable
30	R/W	0	TCON1_Vb_Int_En  0: disable  1: enable
29	R/W	0	TCON0_Line_Int_En  0: disable  1: enable
28	R/W	0	TCON1_Line_Int_En  0: disable  1: enable
27	R/W	0	TCON0_Tri_Finish_Int_En  0: disable  1: enable
26:	R/W	0	TCON0_Tri_Counter_Int_En  0: disable  1: enable
25:16	/	/	/
15	R/W	0	TCON0_Vb_Int_Flag

			Assert at inactive area.  Write 0 to clear it.
14	R/W	0	TCON1_Vb_Int_Flag  Assert at inactive area.  Write 0 to clear it.
13	R/W	0	TCON0_Line_Int_Flag  trigger when SY0 match the current TCON0 scan line  Write 0 to clear it.
12	R/W	0	TCON1_Line_Int_Flag  trigger when SY1 match the current TCON1 scan line  Write 0 to clear it.
11	R/W	0	TCON0_Tri_Finish_Int_Flag  trigger when cpu trigger mode finish  Write 0 to clear it.
10	R/W	0	TCON0_Tri_Counter_Int_Flag  trigger when tri counter reache this value  Write 0 to clear it.
9	R/W	0	TCON0_Tri_Underflow_Flag  only used in DSI video mode, tri when sync by DSI but not finish  Write 0 to clear it.
8:0	/	/	/

**7.2.11.3. TCON global interrupt register1**

Offset: 0x008			Register Name: TCON_GINT1_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27:16	R/W	0	TCON0_Line_Int_Num  Scan line for TCON0 line interrupt (inactive lines are calculated).  Note: SY0 is writable only when LINE_TRG0 disable.
15:12	/	/	/
11:0	R/W	0	TCON1_Line_Int_Num  Scan line for TCON1 line interrupt (inactive lines are calculated).  Note: SY1 is writable only when LINE_TRG1 disable.

**7.2.11.4. TCON1 control register**

Offset: 0x090			Register Name: TCON1_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	TCON1_En  0: disable  1: enable
30:21	/	/	/
20	R/W	0	Interlace_En  0:disable

			1:enable
19:9	/	/	/
8:4	R/W	0	Start_Delay  This is for DE1 and DE2
3:2	/	/	/
1:0	R/W	0	TCON1_Src_Sel  00: DE 0  01: DE 1  1x: BLUE data(FIFO2 disable, RGB=0000FF)

#### 7.2.11.5. TCON1 basic timing register0

Offset: 0x094			Register Name: TCON1_BASIC0_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27:16	R/W	0	TCON1_XI  source width is X+1
15:12	/	/	/
11:0	R/W	0	TCON1_YI  source height is Y+1

#### 7.2.11.6. TCON1 basic timing register1

Offset: 0x098			Register Name: TCON1_BASIC1_REG
Bit	Read/ Write	Default/ Hex	Description

31:28	/	/	/
27:16	R/W	0	LS_XO  width is LS_XO+1
15:12	/	/	/
11:0	R/W	0	LS_YO  width is LS_YO+1  NOTE: this version LS_YO = TCON1_YI

#### 7.2.11.7. TCON1 basic timing register2

Offset: 0x09C			Register Name: TCON1_BASIC2_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27:16	R/W	0	TCON1_XO  width is TCON1_XO+1
15:12	/	/	/
11:0	R/W	0	TCON1_YO  height is TCON1_YO+1

#### 7.2.11.8. TCON1 basic timing register3

Offset: 0x0A0			Register Name: TCON1_BASIC3_REG
Bit	Read/ Write	Default/ Hex	Description
31:29	/	/	/
28:16	R/W	0	HT

			horizontal total time $T_{ht} = (HT+1) * T_{dclk}$
15:12	/	/	/
11:0	R/W	0	HBP  horizontal back porch  $T_{hbp} = (HBP +1) * T_{dclk}$

#### 7.2.11.9. TCON1 basic timing register4

Offset: 0x0A4			Register Name: TCON1_BASIC4_REG
Bit	Read/ Write	Default/ Hex	Description
31:29	/	/	/
28:16	R/W	0	VT  horizontal total time (in HD line)  $T_{vt} = VT/2 * T_{ht}$
15:12	/	/	/
11:0	R/W	0	VBP  horizontal back porch (in HD line)  $T_{vbp} = (VBP +1) * T_{ht}$

#### 7.2.11.10. TCON1 basic timing register5

Offset: 0x0A8			Register Name: TCON1_BASIC5_REG
Bit	Read/ Write	Default/ Hex	Description



31:26	/	/	/
25:16	R/W	0	HSPW  horizontal Sync Pulse Width (in dclk)  $T_{hspw} = (HSPW+1) * T_{dclk}$  Note: $HT > (HSPW+1)$
15:10	/	/	/
9:0	R/W	0	VSPW  vertical Sync Pulse Width (in lines)  $T_{vspw} = (VSPW+1) * T_{ht}$  Note: $VT/2 > (VSPW+1)$

#### 7.2.11.11. TCON1 sync register

Offset: 0x0B0			Register Name: TCON1_PS_SYNC_REG
Bit	Read/ Write	Default/ Hex	Description
31:16	R/W	0	SYNC_X
15:0	R/W	0	SYNC_Y

#### 7.2.11.12. TCON1 IO polarity register

Offset: 0x0F0			Register Name: TCON1_IO_POL_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27	R/W	0	IO3_Inv

			0: not invert 1: invert
26	R/W	0	IO2_Inv 0: not invert 1: invert
25	R/W	0	IO1_Inv 0: not invert 1: invert
24	R/W	0	IO0_Inv 0: not invert 1: invert
23:0	R/W	0	Data_Inv TCON1 output port D[23:0] polarity control, with independent bit control: 0s: normal polarity 1s: invert the specify output

### 7.2.11.13. TCON1 IO control register

Offset: 0x0F4			Register Name: TCON1_IO_TRI_REG
Bit	Read/ Write	Default/ Hex	Description
31:28	/	/	/
27	R/W	1	IO3_Output_Tri_En

			1: disable 0: enable
26	R/W	1	IO2_Output_Tri_En  1: disable 0: enable
25	R/W	1	IO1_Output_Tri_En  1: disable 0: enable
24	R/W	1	IO0_Output_Tri_En  1: disable 0: enable
23:0	R/W	0xFFFFFFFF	Data_Output_Tri_En  TCON1 output port D[23:0] output enable, with independent bit control:  1s: disable 0s: enable

#### 7.2.11.14. TCON ECC FIFO register

Offset: 0x0F8			Register Name: TCON_ECC_FIFO_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	/	ECC_FIFO_BIST_EN

			0: disable  1: enable
30	R/W	/	ECC_FIFO_ERR_FLAG
29:24	/	/	/
23:16	R/W	/	ECC_FIFO_ERR_BITS
15:9	/	/	/
8	R/W	/	ECC_FIFO_BLANK_EN  0: disable ECC function in blanking  1: enable ECC function in blanking  ECC function is tent to trigged in blanking area at HV mode, set '0' when in HV mode
7:0	R/W	/	ECC_FIFO_SETTING  Note: bit3 0 enable, 1 disable

#### 7.2.11.15. TCON CEU control register

Offset: 0x100			Register Name: TCON_CEU_CTL_REG
Bit	Read/ Write	Default/ Hex	Description
31	R/W	0	CEU_en  0: bypass  1: enable
30:0	/	/	/

**7.2.11.16. TCON CEU coefficient register0**

Offset: 0x110+N*0x04  (N=0,1,2,4,5,6,8,9,10)		Register Name: TCON_CEU_COEF_MUL_REG	
Bit	Read/ Write	Default/ Hex	Description
31:13	/	/	/
12:0	R/W	0	CEU_Coef_Mul_Value  signed 13bit value, range of (-16,16)  N=0: Rr  N=1: Rg  N=2: Rb  N=4: Gr  N=5: Gg  N=6: Gb  N=8: Br  N=9: Bg  N=10: Bb

**7.2.11.17. TCON CEU coefficient register1**

Offset: 0x11C+N*0x10  (N=0,1,2)		Register Name: TCON_CEU_COEF_ADD_REG	
---------------------------------------	--	--------------------------------------	--

Bit	Read/ Write	Default/ Hex	Description
31:19	/	/	/
18:0	R/W	0	CEU_Coef_Add_Value  signed 19bit value, range of (-16384, 16384)  N=0: Rc  N=1: Gc  N=2: Bc

#### 7.2.11.18. TCON CEU coefficient register2

Offset: 0x140+N*0x04  (N=0,1,2)		Register Name: TCON_CEU_COEF_RANG_REG	
Bit	Read/ Write	Default/ Hex	Description
31:24	/	/	/
23:16	R/W	0	CEU_Coef_Range_Min  unsigned 8bit value, range of [0,255]
15:8	/	/	/
7:0	R/W	0	CEU Coef_Range_Max  unsigned 8bit value, range of [0,255]

## 7.3. DEFE

### 7.3.1. Overview

The display engine front-end (DEFE) provides image resizing function for display engine. It receives data from DRAM, performs the image resizing function, and outputs to DEBE module.

The DEFE can receive ARGB/YUV420/YUV422/YUV411 data format, and then converts to ARGB8888 for display. Horizontal and vertical direction scaling are implemented independently.

The DEFE features:

- Support YUV444/ YUV422/ YUV420/ YUV411/ ARGB8888/ARGB4444/RGB565 and ARGB1555 data format
- Support 1/16× to 32× resize ratio
- Support 32-phase 8-tap horizontal anti-alias filter, 32-phase 4-tap vertical anti-alias filter
- Support 1080p 3D format content

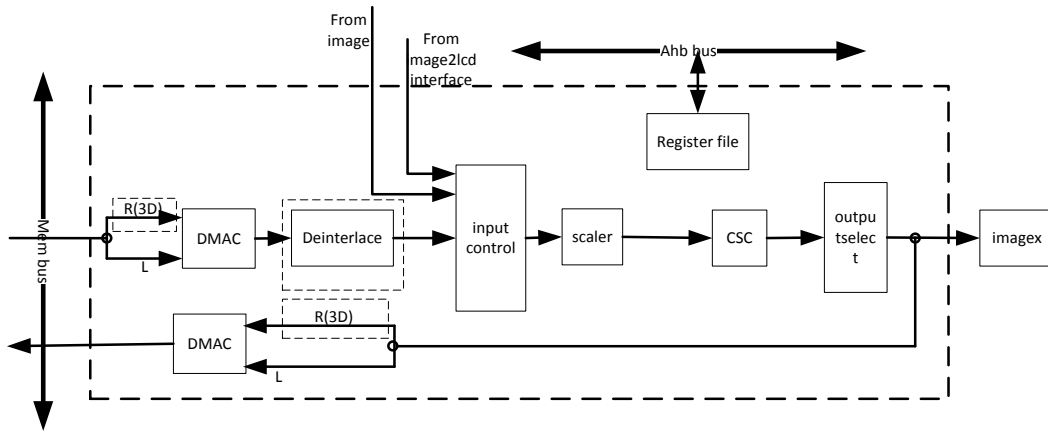
DEFE0 and DEFE1 features:

- Support input size up to 4096×4096 for YUV422/YUV420/YUV411, 2560×2560 for RGB and YUV444
- Support output size up to 4096×4096 for YUV422/YUV420/YUV411, 2560×2560 for RGB and YUV444
- Support de-interlacing

DEFE2 features:

- Support input size up to 4096×4096 for all formats
- Support output size up to 4096×4096 for all formats
- Support data from DEBE

7.3.2. Block Diagram



DEFE Block Diagram



### 7.3.3. DEFE Register List

Module Name	Base Address
DEFE0	0x03100000
DEFE1	0x03140000
DEFE2	0x03180000

Register Name	Offset	Description
DEFE_EN_REG	0x0000	DEFE Module Enable Register
DEFE_FRM_CTRL_REG	0x0004	DEFE Frame Process Control Register
DEFE_BYPASS_REG	0x0008	DEFE CSC By-Pass Register
DEFE_AGTH_SEL_REG	0x000C	DEFE Algorithm Selection Register
DEFE_LINT_CTRL_REG	0x0010	DEFE Line Interrupt Control Register
DEFE_3D_PRELUMA_REG	0x001C	DEFE 3D Pre-Luma Buffer Address Register
DEFE_BUF_ADDR0_REG	0x0020	DEFE Input Channel 0 Buffer Address Register
DEFE_BUF_ADDR1_REG	0x0024	DEFE Input Channel 1 Buffer Address Register
DEFE_BUF_ADDR2_REG	0x0028	DEFE Input Channel 2 Buffer Address Register
DEFE_FIELD_CTRL_REG	0x002C	DEFE Field Sequence Register
DEFE_TB_OFF0_REG	0x0030	DEFE Channel 0 Tile-Based Offset Register
DEFE_TB_OFF1_REG	0x0034	DEFE Channel 1 Tile-Based Offset Register
DEFE_TB_OFF2_REG	0x0038	DEFE Channel 2 Tile-Based Offset Register
DEFE_3D_PRECHROMA_REG	0x003C	DEFE 3D Pre Chroma Buffer Address Register
DEFE_LINESTRD0_REG	0x0040	DEFE Channel 0 Pitch Register
DEFE_LINESTRD1_REG	0x0044	DEFE Channel 1 Pitch Register
DEFE_LINESTRD2_REG	0x0048	DEFE Channel 2 Pitch Register
DEFE_INPUT_FMT_REG	0x004C	DEFE Input Format Register
DEFE_WB_ADDR0_REG	0x0050	DEFE Write Back Address0 Register

DEFE_WB_ADDR1_REG	0x0054	DEFE Write Back Address1 Register
DEFE_WB_ADDR2_REG	0x0058	DEFE Write Back Address2 Register
DEFE_OUTPUT_FMT_REG	0x005C	DEFE Output Format Register
DEFE_INT_EN_REG	0x0060	DEFE Interrupt Enable Register
DEFE_INT_STATUS_REG	0x0064	DEFE Interrupt Status Register
DEFE_STATUS_REG	0x0068	DEFE Status Register
DEFE_CSC_COEF00_REG	0x0070	DEFE CSC Coefficient 00 Register
DEFE_CSC_COEF01_REG	0x0074	DEFE CSC Coefficient 01 Register
DEFE_CSC_COEF02_REG	0x0078	DEFE CSC Coefficient 02 Register
DEFE_CSC_COEF03_REG	0x007C	DEFE CSC Coefficient 03 Register
DEFE_CSC_COEF10_REG	0x0080	DEFE CSC Coefficient 10 Register
DEFE_CSC_COEF11_REG	0x0084	DEFE CSC Coefficient 11 Register
DEFE_CSC_COEF12_REG	0x0088	DEFE CSC Coefficient 12 Register
DEFE_CSC_COEF13_REG	0x008C	DEFE CSC Coefficient 13 Register
DEFE_CSC_COEF20_REG	0x0090	DEFE CSC Coefficient 20 Register
DEFE_CSC_COEF21_REG	0x0094	DEFE CSC Coefficient 21 Register
DEFE_CSC_COEF22_REG	0x0098	DEFE CSC Coefficient 22 Register
DEFE_CSC_COEF23_REG	0x009C	DEFE CSC Coefficient 23 Register
DEFE_DI_CTRL_REG	0x00A0	DEFE De-interlacing Control Register
DEFE_DI_DIAGINTP_REG	0x00A4	DEFE De-interlacing Diag-Interpolate Register
DEFE_DI_TEMPDIFF_REG	0x00A8	DEFE De-interlacing Temp-Difference Register
DEFE_DI_LUMA_TH_REG	0x00AC	DEFE De-interlacing Luma Motion Threshold Register
DEFE_DI_SPATCOMP_REG	0x00B0	DEFE De-interlacing Spatial Compare Register
DEFE_DI_CHROMADIFF_REG	0x00B4	DEFE De-interlacing Chroma Diff Register
DEFE_DI_PRELUMA_REG	0x00B8	DEFE De-interlacing Pre-Frame Luma Address Register
DEFE_DI_PRECHROMA_REG	0x00BC	DEFE De-interlacing Pre-Frame Chroma Address Register
DEFE_DI_TILEFLAG0_REG	0x00C0	DEFE De-interlacing Tile Flag0 Address Register

DEFE_DI_TILEFLAG1_REG	0x00C4	DEFE De-interlacing Tile Flag1 Address Register
DEFE_DI_FLAGLINESTRD_REG	0x00C8	DEFE De-interlacing Tile Flag Pitch Register
DEFE_DI_FLAG_SEQ_REG	0x00CC	DEFE De-interlacing Flag Sequence Register
DEFE_WB_LINESTRD_EN_REG	0x00D0	DEFE Write Back Pitch Enable Register
DEFE_WB_LINESTRD0_REG	0x00D4	DEFE Write Back Channel 3 Pitch Register
DEFE_WB_LINESTRD1_REG	0x00D8	DEFE Write Back Channel 4 Pitch Register
DEFE_WB_LINESTRD2_REG	0x00DC	DEFE Write Back Channel 5 Pitch Register
DEFE_3D_CTRL_REG	0x00E0	DEFE 3D Mode Control Register
DEFE_3D_BUF_ADDR0_REG	0x00E4	DEFE 3D Channel 0 Buffer Address Register
DEFE_3D_BUF_ADDR1_REG	0x00E8	DEFE 3D Channel 1 Buffer Address Register
DEFE_3D_BUF_ADDR2_REG	0x00EC	DEFE 3D Channel 2 Buffer Address Register
DEFE_3D_TB_OFF0_REG	0x00F0	DEFE 3D Channel 0 Tile-Based Offset Register
DEFE_3D_TB_OFF1_REG	0x00F4	DEFE 3D Channel 1 Tile-Based Offset Register
DEFE_3D_TB_OFF2_REG	0x00F8	DEFE 3D Channel 2 Tile-Based Offset Register
DEFE_3D_WB_LINESTRD_REG	0x00FC	DEFE 3D Write Back Line-Stride Register
DEFE_CH0_INSIZE_REG	0x0100	DEFE Channel 0 Input Size Register
DEFE_CH0_OUTSIZE_REG	0x0104	DEFE Channel 0 Output Size Register
DEFE_CH0_HORZFACT_REG	0x0108	DEFE Channel 0 Horizontal Factor Register
DEFE_CH0_VERTFACT_REG	0x010C	DEFE Channel 0 Vertical factor Register
DEFE_CH0_HORZPHASE_REG	0x0110	DEFE Channel 0 Horizontal Initial Phase Register
DEFE_CH0_VERTPHASE0_REG	0x0114	DEFE Channel 0 Vertical Initial Phase 0 Register
DEFE_CH0_VERTPHASE1_REG	0x0118	DEFE Channel 0 Vertical Initial Phase 1 Register
DEFE_CH0_HORZTAP0_REG	0x0120	DEFE Channel 0 Horizontal Tap Offset 0 Register
DEFE_CH0_HORZTAP1_REG	0x0124	DEFE Channel 0 Horizontal Tap Offset 1 Register
DEFE_CH0_VERTTAP_REG	0x0128	DEFE Channel 0 Vertical Tap Offset Register
DEFE_CH1_INSIZE_REG	0x0200	DEFE Channel 1 Input Size Register
DEFE_CH1_OUTSIZE_REG	0x0204	DEFE Channel 1 Output Size Register
DEFE_CH1_HORZFACT_REG	0x0208	DEFE Channel 1 Horizontal Factor Register

DEFE_CH1_VERTFACT_REG	0x020C	DEFE Channel 1 Vertical factor Register
DEFE_CH1_HORZPHASE_REG	0x0210	DEFE Channel 1 Horizontal Initial Phase Register
DEFE_CH1_VERTPHASE0_REG	0x0214	DEFE Channel 1 Vertical Initial Phase 0 Register
DEFE_CH1_VERTPHASE1_REG	0x0218	DEFE Channel 1 Vertical Initial Phase 1 Register
DEFE_CH1_HORZTAP0_REG	0x0220	DEFE Channel 1 Horizontal Tap Offset 0 Register
DEFE_CH1_HORZTAP1_REG	0x0224	DEFE Channel 1 Horizontal Tap Offset 1 Register
DEFE_CH1_VERTTAP_REG	0x0228	DEFE Channel 1 Vertical Tap Offset Register
DEFE_CH0_HORZCOEF0_REGN	0x0400+N*4	DEFE Channel 0 Horizontal Filter Coefficient Register N=0:31
DEFE_CH0_HORZCOEF1_REGN	0x0480+N*4	DEFE Channel 0 Horizontal Filter Coefficient Register N=0:31
DEFE_CH0_VERTCOEF_REGN	0x0500+N*4	DEFE Channel 0 Vertical Filter Coefficient Register N=0:31
DEFE_CH1_HORZCOEF0_REGN	0x0600+N*4	DEFE Channel 1 Horizontal Filter Coefficient Register N=0:31
DEFE_CH1_HORZCOEF1_REGN	0x0680+N*4	DEFE Channel 1 Horizontal Filter Coefficient Register N=0:31
DEFE_CH1_VERTCOEF_REGN	0x0700+N*4	DEFE Channel 1 Vertical Filter Coefficient Register N=0:31
DEFE_CH3_HORZCOEF0_REGN	0x0800+N*4	DEFE Channel 3 Horizontal Filter Coefficient Register N=0:31
DEFE_CH3_HORZCOEF1_REGN	0x0880+N*4	DEFE Channel 3 Horizontal Filter Coefficient Register N=0:31
DEFE_CH3_VERTCOEF_REGN	0x0900+N*4	DEFE Channel 3 Vertical Filter Coefficient Register N=0:31
DEFE_IN_ADD_HIGH_REG	0x0B00	DEFE Input Channel Buffer Address High bits Register
DEFE_3D_ADD_HIGH_REG	0x0B04	DEFE 3D Channel Buffer Address High bits Register
DEFE_DI_ADD_HIGH_REG	0x0B08	DEFE De-interlacing Buffer Address High bits Register

DEFE_DI_3D_ADD_HIGH_REG	0x0B0C	DEFE 3D De-interlacing Buffer Address High bits Register
DEFE_WB_ADD_HIGH_REG	0x0B10	DEFE Write Back Address High bits Register

### 7.3.4. DEFE Register Description

#### 7.3.4.1. DEFE Module Enable Register

Offset: 0x0			Register Name: DEFE_EN_REG
Bit	Read/Wri te	Default/H ex	Description
31:1	/	/	/
0	R/W	0x0	<p>EN</p> <p>DEFE enable</p> <p>0: Disable</p> <p>1: Enable</p> <p>When DEFE enable bit is disabled, the clock of DEFE module will be disabled</p> <p>If this bit is transition from 0 to 1, the frame process control register and the interrupt enable register will be initialed to default value, and the state machine of the module is reset</p>

#### 7.3.4.2. DEFE Frame Process Control Register

Offset: 0x4			Register Name: DEFE_FRM_CTRL_REG
Bit	Read/Wri te	Default/H ex	Description
31:17	/	/	/
16	R/W	0x0	<p>FRM_START</p> <p>Frame start &amp; reset control</p>

			<p>0: reset</p> <p>1: start</p> <p>If the bit is written to zero, the whole state machine and data paths of DEFE module will be reset.</p> <p>When the bit is written to 1, DEFE will start a new frame process.</p>
15	/	/	/
14:12	R/W	0x0	<p>IN_CTRL</p> <p>DEFE input source control</p> <p>000: from dram</p> <p>100: from DEBE0 interface of DEBE2lcd (don't influence the interface timing of DEBE)</p> <p>101: from DEBE1 interface of DEBE2lcd(don't influence the interface timing of DEBE)</p> <p>110: from DEBE0(influence the interface timing of DEBE)</p> <p>111: from DEBE1(influence the interface timing of DEBE)</p> <p>Other: reserved</p> <p>Note: Only for DEFE2. DEFE0/1 must set to 0.</p>
11	R/W	0x0	OUT_CTRL

			DEFE output control  0: enable DEFE output to DEBE  1: disable DEFE output to DEBE  If DEFE write back function is enable, DEFE output to DEBE isn't recommended.
10	/	/	/
9:8	R/W	0x0	OUT_PORT_SEL  DEFE output port select  00: DEBE0  01: DEBE1  02: DEBE2  other: reserved  Note: Only for DEFE2. DEFE0/1 must set to 0.
7:3	/	/	/
2	R/W	0x0	WB_EN  Write back enable  0: Disable  1: Enable



			<p>If output to DEBE is enable, the writing back process will start when write back enable bit is set and a new frame processing begins. The bit will be self-cleared when writing-back frame process starts.</p>
1	R/W	0x0	<p>COEF_RDY_EN</p> <p>Filter coefficients ready enable</p> <p>0: not ready</p> <p>1: filter coefficients configuration ready</p> <p>In order to avoid the noise, you have to ensure the same set filter coefficients are used in one frame, so the filter coefficients are buffered, the programmer can change the coefficients in any time. When the filter coefficients setting is finished, the programmer should set the bit if the programmer need the new coefficients in next scaling frame.</p> <p>When the new frame start, the bit will be self-cleared.</p>
0	R/W	0x0	<p>REG_RDY_EN</p> <p>Register ready enable</p> <p>0: not ready</p> <p>1: registers configuration ready</p> <p>As same as filter coefficients configuration, in order to ensure the display be correct, the correlative display configuration registers are buffered too, the</p>

			<p>programmer also can change the value of correlative registers in any time. When the registers setting is finished, the programmer should set the bit if the programmer need the new configuration in next scaling frame.</p> <p>When the new frame start, the bit will also be self-cleared.</p>
--	--	--	---

### 7.3.4.3. DEFE CSC By-Pass Register

Offset: 0x8			Register Name: DEFE_BYPASS_REG
Bit	Read/Wri te	Default/H ex	Description
31:30	R/W	0x0	<p>SRAM_MAP_SEL</p> <p>Internal SRAM mapping select</p> <p>0: normal mode, in/out maximum width 1920 for all input formats</p> <p>1: in/out maximum width 4096 only for input yuv420</p> <p>2: in/out maximum width 2560 for all input format</p> <p>3: reserved</p> <p>These bits are valid only when de-interlacing function closed, and source comes from dram. 3D column mode is valid only in normal mode.</p> <p>Note: Only for DEFE0/1. DEFE2 should set to 0.</p>
29:2	/	/	/
1	R/W	0x0	<p>CSC_BYPASS_EN</p> <p>CSC by-pass enable</p>

			0: CSC enable  1: CSC will be by-passed  Actually, in order ensure the module working be correct, This bit only can be set when input data format is the same as output data format (both YUV or both RGB)
0	/	/	/

#### 7.3.4.4. DEFE Algorithm Selection Register

Offset: 0xC			Register Name: DEFE_AGTH_SEL_REG
Bit	Read/Wri te	Default/H ex	Description
31:9	/	/	/
8	R/W	0x0	LINEBUF_AGTH  DEFE line buffer algorithm select  0: horizontal filtered result  1: original data
7:0	/	/	/

#### 7.3.4.5. DEFE Line Interrupt Control Register

Offset: 0x10			Register Name: DEFE_LINT_CTRL_REG
Bit	Read/Wri te	Default/H ex	Description
31:28	/	/	/
27:16	R	0x0	CURRENT_LINE

15	R/W	0x0	FIELD_SEL  Field select  0: each field  1: end field(field counter in reg0x2c)
14:13	/	/	/
12:0	R/W	0x0	TRIG_LINE  Trigger line number of line interrupt

#### 7.3.4.6. DEFE 3D Pre-Luma Buffer Address Register

Offset: 0x1C			Register Name: DEFE_3D_PRELUMA_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PREFRM_ADDR  Pre-frame buffer address of luma for 3D right image.  Note: Only for DEFE0/1, invalid for DEFE2.

#### 7.3.4.7. DEFE Input Channel 0 Buffer Address Register

Offset: 0x20			Register Name: DEFE_BUF_ADDR0_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	BUF_ADDR  DEFE frame buffer address

			<p>In tile-based type:</p> <p>The address is the start address of the line in the first tile used to generating output frame.</p> <p>In non-tile-based type:</p> <p>The address is the start address of the first line.</p>
--	--	--	---

#### 7.3.4.8. DEFE Input Channel 1 Buffer Address Register

Offset: 0x24			Register Name: DEFE_BUF_ADDR1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	<p>BUF_ADDR</p> <p>DEFE frame buffer address</p> <p>In tile-based type:</p> <p>The address is the start address of the line in the first tile used to generating output frame.</p> <p>In non-tile-based type:</p> <p>The address is the start address of the first line.</p>

**7.3.4.9. DEFE Input Channel 2 Buffer Address Register**

Offset: 0x28			Register Name: DEFE_BUF_ADDR2_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	BUF_ADDR  DEFE frame buffer address  In tile-based type:  The address is the start address of the line in the first tile used to generating output frame.  In non-tile-based type:  The address is the start address of the first line.

**7.3.4.10. DEFE Field Sequence Register**

Offset: 0x2C			Register Name: DEFE_FIELD_CTRL_REG
Bit	Read/Wri te	Default/H ex	Description
31:30	/	/	/
29:24	R/W	0x20	FIR_OFFSET  FIR compute initial value
23:13	/	/	/
12	R/W	0x0	FIELD_LOOP_MOD

			Field loop mode  0: the last field  1: the full frame
11	/	/	/
10:8	R/W	0x0	VALID_FIELD_CNT  Valid field counter bit  the valid value = this value + 1
7:0	R/W	0x0	FIELD_CNT  Field counter  each bit specify a field to display  0:top field  1:bottom field

#### 7.3.4.11. DEFE Channel 0 Tile-Based Offset Register

Offset: 0x30			Register Name: DEFE_TB_OFF0_REG
Bit	Read/Wri te	Default/H ex	Description
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the end tile
15:13	/	/	/

12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0  The x offset of the top-left point in the first tile

#### 7.3.4.12. DEFE Channel 1 Tile-Based Offset Register

Offset: 0x34			Register Name: DEFE_TB_OFF1_REG
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the end tile
15:13	/	/	/
12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0  The x offset of the top-left point in the first tile



**7.3.4.13. DEFE Channel 2 Tile-Based Offset Register**

Offset: 0x38			Register Name: DEFE_TB_OFF2_REG
Bit	Read/Wri te	Default/H ex	Description
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the end tile
15:13	/	/	/
12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0  The x offset of the top-left point in the first tile

**7.3.4.14. DEFE 3D Pre-Chroma Buffer Address Register**

Offset: 0x3C			Register Name: DEFE_3D_PRECHROMA_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PREFRM_ADDR  Pre-frame buffer address of chroma for 3D right image.

			Note: Only for DEFE0/1, invalid for DEFE2.
--	--	--	--

**7.3.4.15. DEFE Channel 0 Pitch Register**

Offset: 0x40			Register Name: DEFE_LINESTRD0_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	<p>PITCH</p> <p>In tile-based type</p> <p>The pitch is the distance from the start of the end line in one tile to the start of the first line in next tile(here next tile is in vertical direction)</p> <p>In non-tile-based type</p> <p>The pitch is the distance from the start of one line to the start of the next line.</p>

**7.3.4.16. DEFE Channel 1 Pitch Register**

Offset: 0x44			Register Name: DEFE_LINESTRD1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	<p>PITCH</p> <p>In tile-based type</p> <p>The pitch is the distance from the start of the end line in one tile to the start of the first line in next tile(here next tile is in vertical direction)</p>

			<p>In non-tile-based type</p> <p>The pitch is the distance from the start of one line to the start of the next line.</p>
--	--	--	--

#### 7.3.4.17. DEFE Channel 2 Pitch Register

Offset: 0x48			Register Name: DEFE_LINESTRD2_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	<p>PITCH</p> <p>In tile-based type</p> <p>The pitch is the distance from the start of the end line in one tile to the start of the first line in next tile(here next tile is in vertical direction)</p> <p>In non-tile-based type</p> <p>The pitch is the distance from the start of one line to the start of the next line.</p>

#### 7.3.4.18. DEFE Input Format Register

Offset: 0x4C			Register Name: DEFE_INPUT_FMT_REG
Bit	Read/Wri te	Default/H ex	Description
31:17	/	/	/
16	R/W	0x0	<p>BYTE_SEQ</p> <p>Input data byte sequence selection</p>

			0: P3P2P1P0(word)  1: P0P1P2P3(word)
15	/	/	/
14	R/W	0x0	A_COEF_SEL  Alpha fir coefficient select  0: the same with channel 0(G or Y)  1: the individual coefficients
13	/	/	/
12	R/W	0x0	SCAN_MOD  Scanning Mode selection  0: non-interlace  1: interlace
11	/	/	/
10:8	R/W	0x0	DATA_MOD  Input data mode selection  000: non-tile-based planar data mode  001: interleaved data mode  010: non-tile-based UV combined data mode  100: tile-based planar data mode  110: tile-based UV combined data mode0(32x32)

			111: tile-based UV combined data mode1(128x32)  other: reserved
7	/	/	/
6:4	R/W	0x0	DATA_FMT  Input component data format  In non-tile-based planar data mode:  000: YUV 4:4:4  001: YUV 4:2:2  010: YUV 4:2:0  011: YUV 4:1:1  101: RGB888  Other: Reserved  In interleaved data mode:  000: YUV 4:4:4  001: YUV 4:2:2  100: RGB565  101: ARGB8888  110: ARGB4444

			<p>111: ARGB1555</p> <p>Other: reserved</p> <p>In non-tile-based UV combined data mode:</p> <p>001: YUV 4:2:2</p> <p>010: YUV 4:2:0</p> <p>011: YUV 4:1:1</p> <p>Other: reserved</p> <p>In tile-based planar data mode:</p> <p>001: YUV 4:2:2</p> <p>010: YUV 4:2:0</p> <p>011: YUV 4:1:1</p> <p>Other: Reserved</p> <p>In tile-based UV combined data mode0 (32x32):</p> <p>001: YUV 4:2:2</p> <p>010: YUV 4:2:0</p>
--	--	--	---

			011: YUV 4:1:1  Other: reserved  In tile-based UV combined data mode1 (128x32):  001: YUV 4:2:2  010: YUV 4:2:0  011: YUV 4:1:1  Other: reserved
3:2	/	/	/
1:0	R/W	0x0	DATA_PS  Pixel sequence  In interleaved YUV422 data mode:  00: Y1V0Y0U0  01: V0Y1U0Y0  10: Y1U0Y0V0  11: U0Y1V0Y0  In interleaved YUV444 data mode:

			00: VUYA  01: AYUV  Other: reserved   In UV combined data mode: (UV component)  00: V1U1V0U0  01: U1V1U0V0  Other: reserved   In interleaved ARGB8888 data mode:  00: BGRA  01: ARGB  Other: reserved   In interleaved RGB565 data mode:  00: RGB565  01: BGR565  Other: reserved
--	--	--	--



			<p>In interleaved ARGB4444 data mode:</p> <p>00: ARGB4444</p> <p>01: BGRA4444</p> <p>Other: reserved</p> <p>In interleaved ARGB1555 data mode:</p> <p>00: ARGB1555</p> <p>01: BGRA5551</p> <p>Other: reserved</p>
--	--	--	---

#### 7.3.4.19. DEFE Write Back Address0 Register

Offset: 0x50			Register Name: DEFE_WB_ADDR0_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	WB_ADDR  Write-back address setting for output data.

#### 7.3.4.20. DEFE Write Back Address1 Register

Offset: 0x54			Register Name: DEFE_WB_ADDR1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	WB_ADDR  Write-back address setting for output data.

#### 7.3.4.21. DEFE Write Back Address2 Register

Offset: 0x58			Register Name: DEFE_WB_ADDR2_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	WB_ADDR  Write-back address setting for output data.

#### 7.3.4.22. DEFE Output Format Register

Offset: 0x5C			Register Name: DEFE_OUTPUT_FMT_REG
Bit	Read/Wri te	Default/H ex	Description
31:9	/	/	/
8	R/W	0x0	BYTE_SEQ  Output data byte sequence selection  0: P3P2P1P0(word)

			<p>1: POP1P2P3(word)</p> <p>For ARGB, when this bit is 0, the byte sequence is BGRA, and when this bit is 1, the byte sequence is ARGB;</p>
7	R/W	0x0	<p>ALPHA_EN</p> <p>Output alpha enable</p> <p>0: disable, output alpha value = 0xff</p> <p>1: enable</p>
6:5	R/W	0x0	<p>DATA_PS</p> <p>Output data pixel sequence</p> <p>output UV combined data mode: (UV component)</p> <p>00: V1U1V0U0</p> <p>01: U1V1U0V0</p> <p>Other: reserved</p> <p>Other data format will ignore ps.</p>
4	R/W	0x0	<p>SCAN_MOD</p> <p>Output interlace enable</p> <p>0: disable</p> <p>1: enable</p> <p>When output interlace enable, scaler selects YUV initial phase according to LCD</p>

			field signal
3:0	R/W	0x0	<p>DATA_FMT</p> <p>Data format</p> <p>0000: non-tile-based planar RGB888 conversion data format</p> <p>0001: interleaved BGRA8888 conversion data format</p> <p>0010: interleaved ARGB8888 conversion data format</p> <p>0100: non-tile-based planar YUV 444</p> <p>0101: non-tile-based planar YUV 420</p> <p>0110: non-tile-based planar YUV 422</p> <p>0111: non-tile-based planar YUV 411</p> <p>1101: non-tile-based UV combined YUV 420</p> <p>1110: non-tile-based UV combined YUV 422</p> <p>1111: non-tile-based UV combined YUV 411</p> <p>Other: Reserved</p> <p>Note: When output format is YUV420, only support input format YUV and non-interleaved mode.</p>

**7.3.4.23. DEFE Interrupt Enable Register**

Offset: 0x60			Register Name: DEFE_INT_EN_REG
Bit	Read/Wri te	Default/H ex	Description

31:11	/	/	/
10	R/W	0x0	REG_LOAD_EN Register ready load interrupt enable
9	R/W	0x0	LINE_EN Line interrupt enable
8	/	/	/
7	R/W	0x0	WB_EN Write-back end interrupt enable  0: Disable  1: Enable
6:0	/	/	/

#### 7.3.4.24. DEFE Interrupt Status Register

Offset: 0x64			Register Name: DEFE_INT_STATUS_REG
Bit	Read/Wri te	Default/H ex	Description
31:11	/	/	/
10	R/W	0x0	REG_LOAD_STATUS Register ready load interrupt status
9	R/W	0x0	LINE_STATUS Line interrupt status
8	/	/	/
7	R/W	0x0	WB_STATUS Write-back end interrupt status

6:0	/	/	/
-----	---	---	---

#### 7.3.4.25. DEFE Status Register

Offset: 0x68			Register Name: DEFE_STATUS_REG
Bit	Read/Wri te	Default/H ex	Description
31:29	/	/	/
28:16	R	0x0	LINE_ON_SYNC  Line number(when sync reached)
15	R/W	0x0	WB_ERR_SYNC  Sync reach flag when capture in process
14	R/W	0x0	WB_ERR_LOSEDATA  Lose data flag when capture in process
13	/	/	/
12	R	0x0	WB_ERR_STATUS  write-back error status  0: valid write back  1: un-valid write back  This bit is cleared through write 0 to reset/start bit in frame control register
11:6	/	/	/
5	R	0x0	LCD_FIELD  LCD field status  0: top field

			1: bottom field
4	R	0x0	<p>DRAM_STATUS</p> <p>Access dram status</p> <p>0: idle</p> <p>1: busy</p> <p>This flag indicates whether DEFE is accessing dram</p>
3	/	/	/
2	R	0x0	<p>CFG_PENDING</p> <p>Register configuration pending</p> <p>0: no pending</p> <p>1: configuration pending</p> <p>This bit indicates the registers for the next frame has been configured. This bit will be set when configuration ready bit is set and this bit will be cleared when a new frame process begin.</p>
1	R	0x0	<p>WB_STATUS</p> <p>Write-back process status</p> <p>0: write-back end or write-back disable</p> <p>1: write-back in process</p> <p>This flag indicates that a full frame has not been written back to memory. The bit</p>

			will be set when write-back enable bit is set, and be cleared when write-back process end.
0	R	0x0	<p>FRM_BUSY</p> <p>Frame busy.</p> <p>This flag indicates that the frame is being processed.</p> <p>The bit will be set when frame process reset &amp; start is set, and be cleared when frame process reset or disabled.</p>

#### 7.3.4.26. DEFE CSC Coefficient 00 Register

Offset: 0x70			Register Name: DEFE_CSC_COEF00_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	<p>COEF</p> <p>the Y/G coefficient</p> <p>the value equals to coefficient*2<sup>10</sup></p>

#### 7.3.4.27. DEFE CSC Coefficient 01 Register

Offset: 0x74			Register Name: DEFE_CSC_COEF01_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF



			the Y/G coefficient  the value equals to coefficient*2 <sup>10</sup>
--	--	--	--

#### 7.3.4.28. DEFE CSC Coefficient 02 Register

Offset: 0x78			Register Name: DEFE_CSC_COEF02_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF  the Y/G coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.3.4.29. DEFE CSC Coefficient 03 Register

Offset: 0x7C			Register Name: DEFE_CSC_COEF03_REG
Bit	Read/Wri te	Default/H ex	Description
31:14	/	/	/
13:0	R/W	0x0	CONT  the Y/G constant  the value equals to coefficient*2 <sup>4</sup>

#### 7.3.4.30. DEFE CSC Coefficient 10 Register

Offset: 0x80			Register Name: DEFE_CSC_COEF10_REG
Bit	Read/Wri te	Default/H ex	Description

31:13	/	/	/
12:0	R/W	0x0	COEF  the U/R coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.3.4.31. DEFE CSC Coefficient 11 Register

Offset: 0x84			Register Name: DEFE_CSC_COEF11_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF  the U/R coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.3.4.32. DEFE CSC Coefficient 12 Register

Offset: 0x88			Register Name: DEFE_CSC_COEF12_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF  the U/R coefficient  the value equals to coefficient*2 <sup>10</sup>

**7.3.4.33. DEFE CSC Coefficient 13 Register**

Offset: 0x8C			Register Name: DEFE_CSC_COEF13_REG
Bit	Read/Wri te	Default/H ex	Description
31:14	/	/	/
13:00	R/W	0x0	CONT  the U/R constant  the value equals to coefficient*2 <sup>4</sup>

**7.3.4.34. DEFE CSC Coefficient 20 Register**

Offset: 0x90			Register Name: DEFE_CSC_COEF20_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF  the V/B coefficient  the value equals to coefficient*2 <sup>10</sup>

**7.3.4.35. DEFE CSC Coefficient 21 Register**

Offset: 0x94			Register Name: DEFE_CSC_COEF21_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF

			the V/B coefficient  the value equals to coefficient*2 <sup>10</sup>
--	--	--	--

#### 7.3.4.36. DEFE CSC Coefficient 22 Register

Offset: 0x98			Register Name: DEFE_CSC_COEF22_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12:0	R/W	0x0	COEF  the V/B coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.3.4.37. DEFE CSC Coefficient 23 Register

Offset: 0x9C			Register Name: DEFE_CSC_COEF23_REG
Bit	Read/Wri te	Default/H ex	Description
31:14	/	/	/
13:00	R/W	0x0	CONT  the V/B constant  the value equals to coefficient*2 <sup>4</sup>

#### 7.3.4.38. DEFE De-interlacing Control Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xA0			Register Name: DEFE_DI_CTRL_REG
Bit	Read/Wri te	Default/H ex	Description

	te	ex	
31:26	/	/	/
25	R/W	0x0	TEMPDIFF_EN  Temporal difference compare enable  0: disable  1: enable
24	R/W	0x0	DIAGINTP_EN  De-interlacing diagonal interpolate enable  0: disable  1: enable
23:18	/	/	/
17:16	R/W	0x0	MOD  De-interlacing mode select  00: weave  01: bob  10: reserved  11: pixel-motion-adaptive
15:1	/	/	/
0	R/W	0x0	EN  De-interlacing enable  0: de-interlacing disable

			1: de-interlacing enable
--	--	--	--------------------------

### 7.3.4.39. DEFE De-interlacing Diag-Interpolate Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xA4			Register Name: DEFE_DI_DIAGINTP_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x8	TH3  Diagintp_th3
23:15	/	/	/
14:8	R/W	0x5	TH1  Diagintp_th1
7	/	/	/
6:0	R/W	0x28	TH0  Diagintp_th0

### 7.3.4.40. DEFE De-interlacing Temp-Difference Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xA8			Register Name: DEFE_DI_TEMPDIFF_REG
Bit	Read/Wri te	Default/H ex	Description
31:27	/	/	/
26:16	R/W	0xF	DIRECT_DITHER_TH
15	/	/	/
14:8	R/W	0xA	AMBIGUITY_TH
7	/	/	/

6:0	R/W	0xA	SAD_CENTRAL_TH
-----	-----	-----	----------------

#### 7.3.4.41. DEFE De-interlacing Luma Motion Threshold Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xAC			Register Name: DEFE_DI_LUMA_TH_REG
Bit	Read/Wri te	Default/H ex	Description
31:26	/	/	/
25:24	R/W	0x3	Pixel_Static_TH
23:16	R/W	0x6	AvgLumaShifter
15:8	R/W	0x10	MaxLumaTh
7:0	R/W	0x9	MinLumaTh

#### 7.3.4.42. DEFE De-interlacing Spatial Compare Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xB0			Register Name: DEFE_DI_SPATCOMP_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x14	TH3  spatial_th3
15:8	/	/	/
7:0	R/W	0xA	TH2  spatial_th2

#### 7.3.4.43. DEFE De-interlacing Chroma Diff Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xB4			Register Name: DEFE_DI_CHROMADIFF_REG
Bit	Read/Wri te	Default/H ex	Description
31:30	/	/	/
29:24	R/W	0x1F	CHROMA Chroma burst length
23:22	/	/	/
21:16	R/W	0x1F	LUMA Luma burst length
15:8	/	/	/
7:0	R/W	0x5	CHROMA_DIFF_TH

#### 7.3.4.44. DEFE De-interlacing Pre-Frame Luma Address Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xB8			Register Name: DEFE_DI_PRELUMA_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PREFRM_ADDR Pre-frame buffer address of luma

#### 7.3.4.45. DEFE De-interlacing Pre-Frame Chroma Address Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xBC			Register Name: DEFE_DI_PRECHROMA_REG
Bit	Read/Wri te	Default/H ex	Description



31:0	R/W	0x0	PREFRM_ADDR Pre-frame buffer address of chroma
------	-----	-----	---

#### 7.3.4.46. DEFE De-interlacing Tile Flag0 Address Register

Note: Only for DEFE0/1. Invalid for DEFE2.

Offset: 0xC0			Register Name: DEFE_DI_TILEFLAG0_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	TILE_FLAG_ADDR0 Current frame tile flag buffer address

#### 7.3.4.47. DEFE De-interlacing Tile Flag1 Address Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xC4			Register Name: DEFE_DI_TILEFLAG1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	TILE_FLAG_ADDR1 Current frame tile flag buffer address

#### 7.3.4.48. DEFE De-interlacing Tile Flag Pitch Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xC8			Register Name: DEFE_DI_FLAGLINESTRD_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x200	TILE_FLAG_LINESTRD

			tile flag line-stride
--	--	--	-----------------------

#### 7.3.4.49. DEFE De-interlacing Flag Sequence Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xCC			Register Name: DEFE_DI_SEQ_REG
Bit	Read/Wri te	Default/H ex	Description
31:13	/	/	/
12	R/W	0x0	FIELD_LOOP_MOD  Field loop mode for de-interlace flag  0: the last field; 1: the full frame
11	/	/	/
10:8	R/W	0x0	VALID_FIELD_CNT  Valid field counter bit for de-interlace flag  the valid value = this value + 1:
7:0	R/W	0x0	FIELD_CNT  Field counter for de-interlace flag  each bit specify a field to display, 0: flag0, 1: flag1

#### 7.3.4.50. DEFE Write Back Pitch Enable Register

Offset: 0xD0			Register Name: DEFE_WB_LINESTRD_EN_REG
Bit	Read/Wri te	Default/H ex	Description
31:1	/	/	/

0	R/W	0x0	EN  Write back pitch enable  0: disable  1: enable
---	-----	-----	--

#### 7.3.4.51. DEFE Write Back Channel 3 Pitch Register

Offset: 0xD4			Register Name: DEFE_WB_LINESTRD0_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PITCH  Ch3 write back pitch

#### 7.3.4.52. DEFE Write Back Channel 4 Pitch Register

Offset: 0xD8			Register Name: DEFE_WB_LINESTRD1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PITCH  Ch4 write back pitch

#### 7.3.4.53. DEFE Write Back Channel 5 Pitch Register

Offset: 0xDC			Register Name: DEFE_WB_LINESTRD2_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	PITCH  Ch5 write back pitch

**7.3.4.54. DEFE 3D Mode Control Register**

Offset: 0xE0			Register Name: DEFE_3D_CTRL_REG
Bit	Read/Wri te	Default/H ex	Description
31:26	/	/	/
25:24	R/W	0x0	TB_OUT_MOD_FIELD  Top/bottom output mode field number  0: left or left 1st field(determined by reg0x2c)  1: right or right 1st field  2: left 2nd field  3: right 2nd field
23:19	/	/	/
18:16	R/W	0x0	CI_OUT_MOD  3D column interleaved mode  0: CI_1  1: CI_2  2: CI_3  3: CI_4  Other: reserved
15:13	/	/	/
12	R/W	0x0	TB_OUT_SCAN_MOD

			Output top/bottom scan mode selection  0: progressive  1: interlace
11	R/W	0x0	LI_IN_EN  3D input line interleaved enable
10	R/W	0x0	SS_OUT_EN  3D output side by side mode enable
9	/	/	/
8	R/W	0x0	CI_OUT_EN  3D Column interleaved mode output enable
7:2	/	/	/
1:0	R/W	0x0	MOD_SEL  3D mode select  00: normal output mode(2D mode)  01: 3D side by side/line interleaved/column interleaved output mode  10: 3D top/bottom output mode  11: reserved  When 3D mode is enable, DEFE will enter 3D mode(source will be composed of left and right frame, output will be composed of left and right frame).

#### 7.3.4.55. DEFE 3D Channel 0 Buffer Address Register

Offset: 0xE4	Register Name: DEFE_3D_BUF_ADDR0_REG
--------------	--------------------------------------

Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	RIGHT_CH0_ADDR  3D mode channel 0 buffer address  This address is the start address of right image in 3D mode

#### 7.3.4.56. DEFE 3D Channel 1 Buffer Address Register

Offset: 0xE8			Register Name: DEFE_3D_BUF_ADDR1_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	RIGHT_CH1_ADDR  3D mode channel 1 buffer address  This address is the start address of right image in 3D mode

#### 7.3.4.57. DEFE 3D Channel 2 Buffer Address Register

Offset: 0xEC			Register Name: DEFE_3D_BUF_ADDR2_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	RIGHT_CH2_ADDR  3D mode channel 2 buffer address  This address is the start address of right image in 3D mode

#### 7.3.4.58. DEFE 3D Channel 0 Tile-Based Offset Register

Offset: 0xF0			Register Name: DEFE_3D_TB_OFF0_REG
Bit	Read/Wri te	Default/H ex	Description

	te	ex	
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the first tile
15:13	/	/	/
12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0  The x offset of the top-left point in the first tile  This value is the start offset of right image in 3D mode

#### 7.3.4.59. DEFE 3D Channel 1 Tile-Based Offset Register

Offset: 0xF4			Register Name: DEFE_3D_TB_OFF1_REG
Bit	Read/Wri te	Default/H ex	Description
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the first tile
15:13	/	/	/
12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0

			The x offset of the top-left point in the first tile  This value is the start offset of right image in 3D mode
--	--	--	--

#### 7.3.4.60. DEFE 3D Channel 2 Tile-Based Offset Register

Offset: 0xF8			Register Name: DEFE_3D_TB_OFF2_REG
Bit	Read/Wri te	Default/H ex	Description
31:23	/	/	/
22:16	R/W	0x0	X_OFFSET1  The x offset of the bottom-right point in the first tile
15:13	/	/	/
12:8	R/W	0x0	Y_OFFSET0  The y offset of the top-left point in the first tile
7	/	/	/
6:0	R/W	0x0	X_OFFSET0  The x offset of the top-left point in the first tile  This value is the start offset of right image in 3D mode

#### 7.3.4.61. DEFE 3D Write Back Line-Stride Register

Offset: 0xFC			Register Name: DEFE_3D_WB_STRD_REG
Bit	Read/Wri te	Default/H ex	Description
31:0	R/W	0x0	WB_STRIDE  Write back stride length



			The stride length is the distance between the first point of left image and the first point of right image for 3D top-bottom mode write back
--	--	--	--

#### 7.3.4.62. DEFE Channel 0 Input Size Register

Offset: 0x100			Register Name: DEFE_CH0_INSIZE_REG
Bit	Read/Wri te	Default/H ex	Description
31:29	/	/	/
28:16	R/W	0x0	IN_HEIGHT  Input image Y/G component height  Input image height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	0x0	IN_WIDTH  Input image Y/G component width  The image width = The value of these bits add 1  Limitation: In DEFE0/1, when line buffer result selection is original data, the maximum width is 4096 for yuv420 format and 2560 for others. In DEFE2, limitation is 4096 for all input format.

#### 7.3.4.63. DEFE Channel 0 Output Size Register

Offset: 0x104	Register Name: DEFE_CH0_OUTSIZE_REG
---------------	-------------------------------------

Bit	Read/Wri te	Default/H ex	Description
31:29	/	/	/
28:16	R/W	0x0	OUT_HEIGHT  Output layer Y/G component height  The output layer height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	0x0	OUT_WIDTH  Output layer Y/G component width  The output layer width = The value of these bits add 1  Limitation: In DEFE0/1, when line buffer result selection is horizontal filtered result, the maximum width is 4096 for yuv420 and 2560 for others. In DEFE2, limitation is 4096 for all input format.

#### 7.3.4.64. DEFE Channel 0 Horizontal Factor Register

Offset: 0x108			Register Name: DEFE_CH0_HORZFACT_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x0	FACTOR_INT  The integer part of the horizontal scaling ratio

			the horizontal scaling ratio = input width/output width
15:0	R/W	0x0	FACTOR_FRAC  The fractional part of the horizontal scaling ratio  the horizontal scaling ratio = input width/output width

#### 7.3.4.65. DEFE Channel 0 Vertical factor Register

Offset: 0x10C			Register Name: DEFE_CH0_VERTFACT_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x0	FACTOR_INT  The integer part of the vertical scaling ratio  the vertical scaling ratio = input height/output height
15:0	R/W	0x0	FACTOR_FRAC  The fractional part of the vertical scaling ratio  the vertical scaling ratio = input height /output height

**7.3.4.66. DEFE Channel 0 Horizontal Initial Phase Register**

Offset: 0x110			Register Name: DEFE_CH0_HORZPHASE_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/
19:0	R/W	0x0	PHASE  Y/G component initial phase in horizontal (complement)  This value equals to initial phase * $2^{16}$

**7.3.4.67. DEFE Channel 0 Vertical Initial Phase 0 Register**

Offset: 0x114			Register Name: DEFE_CH0_VERTPHASE0_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/
19:0	R/W	0x0	PHASE  Y/G component initial phase in vertical for top field (complement)  This value equals to initial phase * $2^{16}$

**7.3.4.68. DEFE Channel 0 Vertical Initial Phase 1 Register**

Offset: 0x118			Register Name: DEFE_CH0_VERTPHASE1_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/
19:0	R/W	0x0	PHASE

			Y/G component initial phase in vertical for bottom field (complement)  This value equals to initial phase * 2 <sup>16</sup>
--	--	--	---

#### 7.3.4.69. DEFE Channel 0 Horizontal Tap Offset 0 Register

Offset: 0x120			Register Name: DEFE_CH0_HORZTAPO_REG
Bit	Read/Wri te	Default/H ex	Description
31	/	/	/
30:24	R/W	0x1	TAP3  Tap 3 offset in horizontal
23	/	/	/
22:16	R/W	0x1	TAP2  Tap 2 offset in horizontal
15	/	/	/
14:8	R/W	0x1	TAP1  Tap 1 offset in horizontal
7	/	/	/
6:0	R/W	0x7D	TAP0  Tap 0 offset in horizontal

#### 7.3.4.70. DEFE Channel 0 Horizontal Tap Offset 1 Register

Offset: 0x124			Register Name: DEFE_CH0_HORZTAP1_REG
Bit	Read/Wri te	Default/H ex	Description
31	/	/	/

30:24	R/W	0x1	TAP7 Tap 7 offset in horizontal
23	/	/	/
22:16	R/W	0x1	TAP6 Tap 6 offset in horizontal
15	/	/	/
14:8	R/W	0x1	TAP5 Tap 5 offset in horizontal
7	/	/	/
6:0	R/W	0x1	TAP4 Tap 4 offset in horizontal

#### 7.3.4.71. DEFE Channel 0 Vertical Tap Offset Register

Offset: 0x128			Register Name: DEFE_CH0_VERTTAP_REG
Bit	Read/Wri te	Default/H ex	Description
31	/	/	/
30:24	R/W	0x1	TAP3 Tap 3 offset in vertical
23	/	/	/
22:16	R/W	0x1	TAP2 Tap 2 offset in vertical
15	/	/	/
14:8	R/W	0x1	TAP1

			Tap 1 offset in vertical
7	/	/	/
6:0	R/W	0x7F	TAP0  Tap 0 offset in vertical

#### 7.3.4.72. DEFE Channel 1 Input Size Register

Offset: 0x200			Register Name: DEFE_CH1_INSIZE_REG
Bit	Read/Wri te	Default/H ex	Description
31:29	/	/	/
28:16	R/W	0x0	IN_HEIGHT  Input image U/R component height  Input image height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	0x0	IN_WIDTH  Input image U/R component width  The image width = The value of these bits add 1  Limitation: In DEFE0/1, when line buffer result selection is original data, the maximum width is 4096 for yuv420 format and 2560 for others. In DEFE2, limitation is 4096 for all input format.

**7.3.4.73. DEFE Channel 1 Output Size Register**

Offset: 0x204			Register Name: DEFE_CH1_OUTSIZE_REG
Bit	Read/Wri te	Default/H ex	Description
31:29	/	/	/
28:16	R/W	0x0	OUT_HEIGHT  Output layer U/R component height  The output layer height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	0x0	OUT_WIDTH  Output layer U/R component width  The output layer width = The value of these bits add 1  Limitation: In DEFE0/1, when line buffer result selection is horizontal filtered result, the maximum width is 4096 for yuv420 and 2560 for others. In DEFE2, limitation is 4096 for all input format.

**7.3.4.74. DEFE Channel 1 Horizontal Factor Register**

Offset: 0x208			Register Name: DEFE_CH1_HORZFACT_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/



23:16	R/W	0x0	FACTOR_INT  The integer part of the horizontal scaling ratio  the horizontal scaling ratio = input width/output width
15:0	R/W	0x0	FACTOR_FRAC  The fractional part of the horizontal scaling ratio  the horizontal scaling ratio = input width/output width

#### 7.3.4.75. DEFE Channel 1 Vertical factor Register

Offset: 0x20C			Register Name: DEFE_CH1_VERTFACT_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x0	FACTOR_INT  The integer part of the vertical scaling ratio  the vertical scaling ratio = input height/output height
15:0	R/W	0x0	FACTOR_FRAC  The fractional part of the vertical scaling ratio  the vertical scaling ratio = input height /output height

--	--	--	--

#### 7.3.4.76. DEFE Channel 1 Horizontal Initial Phase Register

Offset: 0x210			Register Name: DEFE_CH1_HORZPHASE_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/
19:0	R/W	0x0	PHASE  U/R component initial phase in horizontal (complement)  This value equals to initial phase * $2^{16}$

#### 7.3.4.77. DEFE Channel 1 Vertical Initial Phase 0 Register

Offset: 0x214			Register Name: DEFE_CH1_VERTPHASE0_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/
19:0	R/W	0x0	PHASE  U/R component initial phase in vertical for top field (complement)  This value equals to initial phase * $2^{16}$

#### 7.3.4.78. DEFE Channel 1 Vertical Initial Phase 1 Register

Offset: 0x218			Register Name: DEFE_CH1_VERTPHASE1_REG
Bit	Read/Wri te	Default/H ex	Description
31:20	/	/	/

19:0	R/W	0x0	PHASE  U/R component initial phase in vertical for bottom field (complement)  This value equals to initial phase * 2 <sup>16</sup>
------	-----	-----	--

#### 7.3.4.79. DEFE Channel 1 Horizontal Tap Offset 0 Register

Offset: 0x220			Register Name: DEFE_CH1_HORZTAPO_REG
Bit	Read/Wri te	Default/H ex	Description
31	/	/	/
30:24	R/W	0x1	TAP3  Tap 3 offset in horizontal
23	/	/	/
22:16	R/W	0x1	TAP2  Tap 2 offset in horizontal
15	/	/	/
14:8	R/W	0x1	TAP1  Tap 1 offset in horizontal
7	/	/	/
6:0	R/W	0x7D	TAP0  Tap 0 offset in horizontal

#### 7.3.4.80. DEFE Channel 1 Horizontal Tap Offset 1 Register

Offset: 0x224			Register Name: DEFE_CH1_HORZTAP1_REG
Bit	Read/Wri	Default/H	Description

	te	ex	
31	/	/	/
30:24	R/W	0x1	TAP7 Tap 7 offset in horizontal
23	/	/	/
22:16	R/W	0x1	TAP6 Tap 6 offset in horizontal
15	/	/	/
14:8	R/W	0x1	TAP5 Tap 5 offset in horizontal
7	/	/	/
6:0	R/W	0x1	TAP4 Tap 4 offset in horizontal

#### 7.3.4.81. DEFE Channel 1 Vertical Tap Offset Register

Offset: 0x228			Register Name: DEFE_CH1_VERTTAP_REG
Bit	Read/Wri te	Default/H ex	Description
31	/	/	/
30:24	R/W	0x1	TAP3 Tap 3 offset in vertical
23	/	/	/
22:16	R/W	0x1	TAP2 Tap 2 offset in vertical
15	/	/	/

14:8	R/W	0x1	TAP1  Tap 1 offset in vertical
7	/	/	/
6:0	R/W	0x7F	TAP0  Tap 0 offset in vertical

#### 7.3.4.82. DEFE Channel 0 Horizontal Filter Coefficient Register

Offset: 0x400+N*4			Register Name: DEFE_CH0_HORZCOEF0_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP3  Horizontal tap3 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP2  Horizontal tap2 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP1  Horizontal tap1 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	TAP0  Horizontal tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>
-----	-----	-----	---

#### 7.3.4.83. DEFE Channel 0 Horizontal Filter Coefficient Register

Offset: 0x480+N*4			Register Name: DEFE_CH0_HORZCOEF1_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP7  Horizontal tap7 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP6  Horizontal tap6 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP5  Horizontal tap5 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	<p>TAP4</p> <p>Horizontal tap4 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
-----	-----	-----	---

#### 7.3.4.84. DEFE Channel 0 Vertical Filter Coefficient Register

Offset: 0x500+N*4			Register Name: DEFE_CH0_VERTCOEF_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	<p>TAP3</p> <p>Vertical tap3 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
23:16	R/W	0x0	<p>TAP2</p> <p>Vertical tap2 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
15:8	R/W	0x0	<p>TAP1</p> <p>Vertical tap1 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>

7:0	R/W	0x0	<p>TAP0</p> <p>Vertical tap0 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
-----	-----	-----	---

#### 7.3.4.85. DEFE Channel 1 Horizontal Filter Coefficient Register

Offset: 0x600+N*4			Register Name: DEFE_CH1_HORZCOEF0_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	<p>TAP3</p> <p>Horizontal tap3 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
23:16	R/W	0x0	<p>TAP2</p> <p>Horizontal tap2 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
15:8	R/W	0x0	<p>TAP1</p> <p>Horizontal tap1 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>



7:0	R/W	0x0	TAP0  Horizontal tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>
-----	-----	-----	---

#### 7.3.4.86. DEFE Channel 1 Horizontal Filter Coefficient Register

Offset: 0x680+N*4			Register Name: DEFE_CH1_HORZCOEF1_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP7  Horizontal tap7 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP6  Horizontal tap6 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP5  Horizontal tap5 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	TAP4  Horizontal tap4 coefficient  The value equals to coefficient*2 <sup>6</sup>
-----	-----	-----	---

#### 7.3.4.87. DEFE Channel 1 Vertical Filter Coefficient Register

Offset: 0x700+N*4			Register Name: DEFE_CH1_VERTCOEF_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP3  Vertical tap3 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP2  Vertical tap2 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP1  Vertical tap1 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	TAP0  Vertical tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>
-----	-----	-----	---

#### 7.3.4.88. DEFE Channel 3 Horizontal Filter Coefficient Register

Offset: 0x800+N*4			Register Name: DEFE_CH3_HORZCOEF0_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP3  Horizontal tap3 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP2  Horizontal tap2 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP1  Horizontal tap1 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	TAP0  Horizontal tap0 coefficient  The value equals to coefficient*2 <sup>6</sup>
-----	-----	-----	---

#### 7.3.4.89. DEFE Channel 3 Horizontal Filter Coefficient Register

Offset: 0x880+N*4			Register Name: DEFE_CH3_HORZCOEF1_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TAP7  Horizontal tap7 coefficient  The value equals to coefficient*2 <sup>6</sup>
23:16	R/W	0x0	TAP6  Horizontal tap6 coefficient  The value equals to coefficient*2 <sup>6</sup>
15:8	R/W	0x0	TAP5  Horizontal tap5 coefficient  The value equals to coefficient*2 <sup>6</sup>

7:0	R/W	0x0	<p>TAP4</p> <p>Horizontal tap4 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
-----	-----	-----	---

#### 7.3.4.90. DEFE Channel 3 Vertical Filter Coefficient Register

Offset: 0x900+N*4			Register Name: DEFE_CH3_VERTCOEF_REGN
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	<p>TAP3</p> <p>Vertical tap3 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
23:16	R/W	0x0	<p>TAP2</p> <p>Vertical tap2 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
15:8	R/W	0x0	<p>TAP1</p> <p>Vertical tap1 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>

7:0	R/W	0x0	<p>TAP0</p> <p>Vertical tap0 coefficient</p> <p>The value equals to coefficient*2<sup>6</sup></p>
-----	-----	-----	---

#### 7.3.4.91. DEFE Input Channel Buffer Address High bits Register

Offset: 0xB00			Register Name: DEFE_IN_ADD_HIGH_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x0	<p>CH2_HIGH_BIT</p> <p>Channel 2 frame buffer address high bit</p>
15:8	R/W	0x0	<p>CH1_HIGH_BIT</p> <p>Channel 1 frame buffer address high bit</p>
7:0	R/W	0x0	<p>CH0_HIGH_BIT</p> <p>Channel 0 frame buffer address high bit</p>

#### 7.3.4.92. DEFE 3D Channel Buffer Address High bits Register

Offset: 0xB04			Register Name: DEFE_3D_ADD_HIGH_REG
Bit	Read/Wri te	Default/H ex	Description

	te	ex	
31:24	/	/	/
23:16	R/W	0x0	3D_RIGHT_CH2_HIGH_BIT  3D Channel 2 frame buffer address high bit
15:8	R/W	0x0	3D_RIGHT_CH1_HIGH_BIT  3D Channel 1 frame buffer address high bit
7:0	R/W	0x0	3D_RIGHT_CH0_HIGH_BIT  3D Channel 0 frame buffer address high bit

### 7.3.4.93. DEFE De-interlacing Buffer Address High bits Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xB08			Register Name: DEFE_DI_ADD_HIGH_REG
Bit	Read/Wri te	Default/H ex	Description
31:24	R/W	0x0	TILE_FLAG1_HIGH_BIT  Current frame tile flag buffer address1 high bit.
23:16	R/W	0x0	TILE_FLAG0_HIGH_BIT  Current frame tile flag buffer address0 high bit.

15:8	R/W	0x0	PREFRM_CHROMA_HIGH_BIT  Pre-frame chroma buffer address high bit.
7:0	R/W	0x0	PREFRM_LUMA_HIGH_BIT  Pre-frame luma buffer address high bit.

#### 7.3.4.94. DEFE 3D De-interlacing Buffer Address High bits Register

Note: Only for DEFE0/1, invalid for DEFE2.

Offset: 0xB0c			Register Name: DEFE_DI_3D_ADD_HIGH_REG
Bit	Read/Wri te	Default/H ex	Description
31:16	/	/	/
15:8	R/W	0x0	3D_RIGHT_PREFRM_CHROMA_HIGH_BIT  Pre-frame buffer address high bit of chroma for 3D right image.
7:0	R/W	0x0	3D_RIGHT_PREFRM_LUMA_HIGH_BIT  Pre-frame buffer address high bit of luma for 3D right image.

#### 7.3.4.95. DEFE Write Back Address High bits Register

Offset: 0xB10	Register Name: DEFE_WB_ADD_HIGH_REG
---------------	-------------------------------------



Bit	Read/Wri te	Default/H ex	Description
31:24	/	/	/
23:16	R/W	0x0	CH5_HIGH_BIT  Write-back address channel 5 high bit of output data.
15:8	R/W	0x0	CH4_HIGH_BIT  Write-back address channel 4 high bit of output data.

## 7.4. DEBE

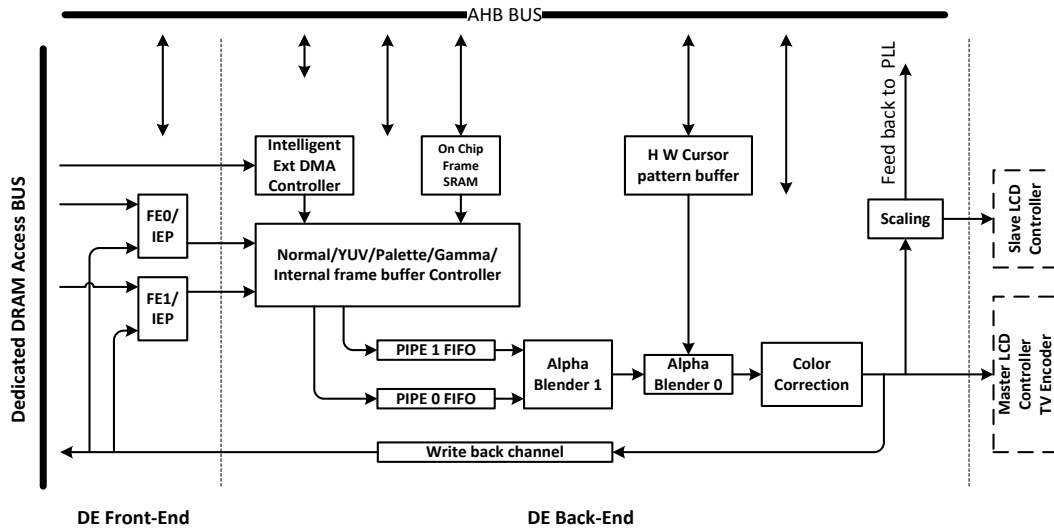
### 7.4.1. Overview

DEBE has two pipes data path, and it can receive data from DEFE or data from SDRAM to do overlay, and then doing alpha blending in the Alpha Blender1 part. After blending, the data can do image enhancement in Color Correction or bypass to the flowing part LCD etc.

It features:

- Support layer size up to 8192x8192 pixels
- Support four layers overlay in one pipe
- Support alpha blending
- Support hardware cursor
- Support color key
- Support pre-multiply alpha image data
- Support internal frame buffer scaling(1x/2x/4x)
- Support 3D format image data
- Support gamma correction
- Support output color correction
- Support input format: 1/2/4/8 bpp/RGB655/RGB565/RGB556/RGB888/ARGB1555/ARGB4444/ARGB8888/iYUV422/iYUV444/YUV422/YUV420/YUV411

7.4.2. Block Diagram



Display Engine Function Block

### 7.4.3. DEBE Register List

Module name	Base address
DEBE0	0x03200000
DEBE1	0x03240000
DEBE2	0x03280000

Register name	Offset	Description
DEBE_MODCTL_REG	0x800	DE back-end mode control register
DEBE_BACKCOLOR_REG	0x804	DE-back color control register
DEBE_DISSIZE_REG	0x808	DE-back display size setting register
DEBE_LAYSIZE_REG	0x810 – 0x81C	DE-layer size register
DEBE_LAYCOORD_REG	0x820 – 0x82C	DE-layer coordinate control register
DEBE_LAYLINEWIDTH_REG	0x840 – 0x84C	DE-layer frame buffer line width register
DEBE_LAYFB_L32ADD_REG	0x850 – 0x85C	DE-layer frame buffer low 32 bit address register
DEBE_LAYFB_H4ADD_REG	0x860	DE-layer frame buffer high 8 bit address register
DEBE_REGBUFFCTL_REG	0x870	DE-Register buffer control register
DEBE_CKMAX_REG	0x880	DE-color key MAX register
DEBE_CKMIN_REG	0x884	DE-color key MIN register
DEBE_CKCFG_REG	0x888	DE-color key configuration register
DEBE_ATTCTL_REG0	0x890 – 0x89C	DE-layer attribute control register0
DEBE_ATTCTL_REG1	0x8A0 – 0x8AC	DE-layer attribute control register1
DEBE_HWCCTL_REG	0x8D8	DE-HWC coordinate control register
DEBE_HWCFBCTL_REG	0x8E0	DE-HWC frame buffer format register

DEBE_IYUVCTL_REG	0x920	DE backend input YUV channel control register
DEBE_IYUVLADD_REG	0x930 – 0x938	DE backend YUV channel frame buffer low 32 bit address register
DEBE_IYUVHADD_REG	0x93C	DE backend YUV channel frame buffer high 8 bit address register
DEBE_IYUVLINEWIDTH_REG	0x940 – 0x948	DE backend YUV channel buffer line width register
DEBE_OCCTL_REG	0x9C0	DE backend output color control register
DEBE_OCRCOEF_REG	0x9D0-0x9D8	DE backend output color R coefficient register
DEBE_OCRCONS_REG	0x9DC	DE backend output color R constant register
DEBE_OCGCOEF_REG	0x9E0-0x9E8	DE backend output color G coefficient register
DEBE_OCGCONS_REG	0x9EC	DE backend output color G constant register
DEBE_OCBCOEF_REG	0x9F0-0x9F8	DE backend output color B coefficient register
DEBE_OCBCONS_REG	0x9FC	DE backend output color B constant register
Memories	0x4400-0x47FF	Gamma table
Memories	0x4800-0x4BFF	DE-HWC pattern memory block
Memories	0x4C00-0x4FFF	DE-HWC color palette table
Memories	0x5000-0x53FF	Pipe0 palette table
Memories	0x5400-0x57FF	Pipe1 palette table

## 7.4.4. DEBE Register Description

### 7.4.4.1. DE back-end mode control register

Offset: 0x800			Register Name: DEBE_MODCTL_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29	R/W	0	LINE_SEL  Start top/bottom line selection in interlace mode
28	R/W	0	ITLMOD_EN  Interlace mode enable  0:disable  1:enable
27:23	/	/	/
22:20	R/W	0	OUT_SEL  Output selection  000:LCD  110:FE0 only  111:FE1 only  Other: reserved
19:18	/	/	/

17	R/W	0	<p>OSCA_EN</p> <p>Output scaling function enable</p> <p>0:disable</p> <p>1:enable</p>
16	R/W	0	<p>HWC_EN</p> <p>Hardware cursor enabled/disabled control</p> <p>0: Disabled</p> <p>1: Enabled</p> <p>Hardware cursor has the highest priority, in the alpha blender0, the alpha value of cursor will be selected</p>
15:12	/	/	/
11	R/W	0	<p>LAY3_EN</p> <p>Layer3 Enable/Disable</p> <p>0: Disabled</p> <p>1: Enabled</p>
10	R/W	0	<p>LAY2_EN</p> <p>Layer2 Enable/Disable</p> <p>0: Disabled</p> <p>1: Enabled</p>
9	R/W	0	LAY1_EN

			Layer1 Enable/Disable  0: Disabled  1: Enabled
8	R/W	0	LAY0_EN  Layer0 Enable/Disable  0: Disabled  1: Enabled
7:2	/	/	/
1	R/W	0	START_CTL  Normal output channel Start & Reset control  0: reset  1: start
0	R/W	0	DEBE_EN  DE back-end enable/disable  0: disable  1: enable

#### 7.4.4.2. DE-back color control register

Offset: 0x804			Register Name: DEBE_BACKCOLOR_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/



23:16	R/W	UDF	BK_RED  Red  Red screen background color value
15:8	R/W	UDF	BK_GREEN  Green  Green screen background color value
7:0	R/W	UDF	BK_BLUE  Blue  Blue screen background color value

#### 7.4.4.3. DE-back display size setting register

Offset: 0x808			Register Name: DEBE_DISSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:16	R/W	UDF	DIS_HEIGHT  Display height  The real display height = The value of these bits add 1
15:0	R/W	UDF	DIS_WIDTH  Display width  The real display width = The value of these bits add 1

**7.4.4.4. DE-layer size register**

Offset:  Layer 0: 0x810  Layer 1: 0x814  Layer 2: 0x818  Layer 3: 0x81C			Register Name: DEBE_LAYSIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	UDF	LAY_HEIGHT  Layer Height  The Layer Height = The value of these bits add 1
15:13	/	/	/
12:0	R/W	UDF	LAY_WIDTH  Layer Width  The Layer Width = The value of these bits add 1

**7.4.4.5. DE-layer coordinate control register**

Offset:  Layer 0: 0x820  Layer 1: 0x824			Register Name: DEBE_LAYCOOR_REG
---	--	--	---------------------------------

Layer 2: 0x828			
Layer 3: 0x82C			
Bit	Read/Write	Default/Hex	Description
31:16	R/W	UDF	<p>LAY_YCOORD</p> <p>Y coordinate</p> <p>Y is the left-top y coordinate of layer on screen in pixels</p> <p>The Y represent the two's complement</p>
15:0	R/W	UDF	<p>LAY_XCOORD</p> <p>X coordinate</p> <p>X is left-top x coordinate of the layer on screen in pixels</p> <p>The X represent the two's complement</p>

Setting the layer0-layer3 the coordinate (left-top) on screen control information

**7.4.4.6. DE-layer frame buffer pitch register**

<p>Offset:</p> <p>Layer 0: 0x840</p> <p>Layer 1: 0x844</p> <p>Layer 2: 0x848</p>	<p>Register Name: DEBE_LAYPITCH_REG</p>
--	---

Layer 3: 0x84C			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	PITCH  Layer frame buffer pitch in bits

Note: If the layer is selected by video channel or YUV channel, the setting of this register will be ignored.

#### 7.4.4.7. DE-layer frame buffer low 32 bit address register

Offset:  Layer 0: 0x850  Layer 1: 0x854  Layer 2: 0x858  Layer 3: 0x85C			Register Name: DEBE_LAYFB_L32ADD_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	LAYFB_L32ADD  Buffer start Address  Layer Frame start Buffer Address in bit

Note: If the layer is selected by video channel or YUV channel, the setting of this register will be ignored.

#### 7.4.4.8. DE-layer frame buffer high 8 bit address register

Offset: 0x860			Register Name: DEBE_LAYFB_H4ADD_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	

31:24	R/W	UDF	LAY3FB_H4ADD  Layer3  Layer Frame Buffer Address in bit
23:16	R/W	UDF	LAY2FB_H4ADD  Layer2  Layer Frame Buffer Address in bit
15:8	R/W	UDF	LAY1FB_H4ADD  Layer1  Layer Frame Buffer Address in bit
7:0	R/W	UDF	LAY0FB_H4ADD  Layer0  Layer Frame Buffer Address in bit

Note: If the layer is selected by video channel or YUV channel, the setting of this register will be ignored.

#### 7.4.4.9. DE-Register buffer control register

Offset: 0x870			Register Name: DEBE_REGBUFFCTL_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W	0X00	REGAUTOLOAD_DIS  Module registers loading auto mode disable control  0: registers auto loading mode

			1: disable registers auto loading mode, the registers will be loaded by write 1 to bit0 of this register
0	R/W	0X00	<p>REGLOADCTL</p> <p>Register load control</p> <p>When the Module registers loading auto mode disable control bit is set, the registers will be loaded by write 1 to the bit, and the bit will self clean when the registers is loading done.</p>

#### 7.4.4.10. DE-color key MAX register

Offset: 0x880			Register Name: DEBE_CKMAX_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	UDF	<p>CKMAX_R</p> <p>Red</p> <p>Red color key max</p>
15:8	R/W	UDF	<p>CKMAX_G</p> <p>Green</p> <p>Green color key max</p>
7:0	R/W	UDF	<p>CKMAX_B</p> <p>Blue</p> <p>Blue color key max</p>

**7.4.4.11. DE-color key MIN register**

Offset: 0x884			Register Name: DEBE_CKMIN_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	UDF	CKMIN_R  Red  Red color key min
15:8	R/W	UDF	CKMIN_G  Green  Green color key min
7:0	R/W	UDF	CKMIN_B  Blue  Blue color key min

**7.4.4.12. DE-color key configuration register**

Offset: 0x888			Register Name: DEBE_CKCFG_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5:4	R/W	UDF	CKR_MATCH  Red Match Rule  00: always match

			01: always match  10: match if (Color Min=<Color<=Color Max)  11: match if (Color>Color Max or Color<Color Min)
3:2	R/W	UDF	CKG_MATCH  Green Match Rule  00: always match  01: always match  10: match if (Color Min=<Color<=Color Max)  11: match if (Color>Color Max or Color<Color Min)
1:0	R/W	UDF	CKB_MATCH  Blue Match Rule  00: always match  01: always match  10: match if (Color Min=<Color<=Color Max)  11: match if (Color>Color Max or Color<Color Min)

#### 7.4.4.13. DE-layer attribute control register0

Offset:  Layer0: 0x890  Layer1: 0x894	Register Name: DEBE_ATTCTL_REG0
---	---------------------------------



Layer2: 0x898			
Layer3: 0x89C			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	UDF	<p>LAY_GLBALPHA</p> <p>Alpha value</p> <p>Alpha value is used for this layer</p>
23:22	R/W	UDF	<p>LAY_WORKMOD</p> <p>Layer working mode selection</p> <p>00: normal mode (Non-Index mode)</p> <p>01: palette mode (Index mode)</p> <p>10: internal frame buffer mode</p> <p>11: gamma correction</p> <p>Except the normal mode, if the other working mode is selected, the on chip SRAM will be enabled.</p>
21:20	R/W	UDF	<p>PREMUL</p> <p>0: normal input layer</p> <p>1: pre-multiply input layer</p> <p>Other: reserved</p>
19:18	R/W	UDF	CKEN

			<p>Color key Mode</p> <p>00: disabled color key</p> <p>01: The layer color key match another channel pixel data in Alpha Blender1.</p> <p>1x: Reserved</p> <p>Only 2 channels pixel data can get to Alpha Blender1 at the same screen coordinate.</p>
17:16	R/W	UDF	<p>LAY_GLB_MIX_ALPHAEN</p> <p>00: normal input layer</p> <p>01: layer alpha value = pixels alpha value*globe alpha value</p> <p>Other: reserved</p>
15	R/W	UDF	<p>LAY_PIPESEL</p> <p>Pipe Select</p> <p>0: select Pipe 0</p> <p>1: select Pipe 1</p>
14:12	/	/	/
11:10	R/W	UDF	<p>LAY_PRISEL</p> <p>Priority</p> <p>The rule is: 11&gt;10&gt;01&gt;00</p>

			<p>When more than 2 layers are enabled, the priority value of each layer must be different, soft designer must keep the condition.</p> <p>If more than 1 layer selects the same pipe, in the overlapping area, only the pixel of highest priority layer can pass the pipe to blender1.</p> <p>If both 2 pipes are selected by layers, in the overlapping area, the alpha value will use the alpha value of higher priority layer in the blender1.</p>
9:6	/	/	/
5:4	R/W	UDF	<p>LAY_VDOSEL</p> <p>Video channel selection control</p> <p>0:select video channel 0 (FE0)</p> <p>1:select video channel 1 (FE1)</p> <p>2:select video channel 2 (FE2)</p> <p>Other: reserved</p> <p>The selection setting is only valid when Layer video channel selection is enabled.</p>
3	/	/	/
2	R/W	UDF	<p>LAY_YUVEN</p> <p>YUV channel selection</p> <p>0: disable</p> <p>1: enable</p>

			Setting 2 or more layers YUV channel mode is illegal, programmer should confirm it.
1	R/W	UDF	<p>LAY_VDOEN</p> <p>Layer video channel selection enable control</p> <p>0: disable</p> <p>1: enable</p> <p>Normally, one layer can not be set both video channel and YUV channel mode, if both 2 mode is set, the layer will work in video channel mode, YUV channel mode will be ignored, programmer should confirm it.</p> <p>Setting 2 or more layers video channel mode is illegal, programmer should confirm it.</p>
0	R/W	UDF	<p>LAY_GLBALPHAEN</p> <p>Alpha Enable</p> <p>0: Disabled the alpha value of this register</p> <p>1: Enabled the alpha value of this register for the layer</p>

#### 7.4.4.14. DE-layer attribute control register1

Offset:	Register Name: DEBE_ATTCTL_REG1
---------	---------------------------------

Layer0: 0x8A0			
Layer1: 0x8A4			
Layer2: 0x8A8			
Layer3: 0x8AC			
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:14	R/W	UDF	<p>LAY_HSCAFCT</p> <p>Setting the internal frame buffer scaling factor, only valid in internal frame buffer mode</p> <p>SH</p> <p>Height scale factor</p> <p>00: no scaling</p> <p>01: *2</p> <p>10: *4</p> <p>11: Reserved</p>
13:12	R/W	UDF	<p>LAY_WSCAFCT</p> <p>Setting the internal frame buffer scaling factor, only valid in internal frame buffer mode</p> <p>SW</p> <p>Width scale factor</p>

			00: no scaling  01: *2  10: *4  11: Reserved
11:8	R/W	UDF	LAY_FBFMT  Frame buffer format  Normal mode data format  0000: mono 1-bpp  0001: mono 2-bpp  0010: mono 4-bpp  0011: mono 8-bpp  0100: color 16-bpp (R:6/G:5/B:5)  0101: color 16-bpp (R:5/G:6/B:5)  0110: color 16-bpp (R:5/G:5/B:6)  0111: color 16-bpp (Alpha:1/R:5/G:5/B:5)  1000: color 16-bpp (R:5/G:5/B:5/Alpha:1)  1001: color 24-bpp (Padding:8/R:8/G:8/B:8)  1010: color 32-bpp (Alpha:8/R:8/G:8/B:8)

			<p>1011: color 24-bpp (R:8/G:8/B:8)</p> <p>1100: color 16-bpp (Alpha:4/R:4/G:4/B:4)</p> <p>1101: color 16-bpp (R:4/G:4/B:4/Alpha:4)</p> <p>Other: Reserved</p> <p>Palette Mode data format</p> <p>In palette mode, the data of external frame buffer is regarded as pattern.</p> <p>0000: 1-bpp</p> <p>0001: 2-bpp</p> <p>0010: 4-bpp</p> <p>0011: 8-bpp</p> <p>other: Reserved</p> <p>Internal Frame buffer mode data format</p> <p>0000: 1-bpp</p> <p>0001: 2-bpp</p> <p>0010: 4-bpp</p> <p>0011: 8-bpp</p>
--	--	--	---

			Other: Reserved
7:3	/	/	/
2	R/W	UDF	LAY_BRSWAPEN  B R channel swap  0: RGB. Follow the bit[11:8]----RGB  1: BGR. Swap the B R channel in the data format.
1:0	R/W	UDF	LAY_FBPS  PS  Pixels Sequence  See the follow table "Pixels Sequence"

**7.4.4.15. Pixels sequence table**

DE-layer attribute control register1 [11:08] = FBF (frame buffer format)

DE-layer attribute control register1 [01:00] = PS (pixels sequence)

**Mono or internal frame buffer 1-bpp or palette 1-bpp mode** : FBF = 0000

PS=00

Bit

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

P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16
P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P01	P00



15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=01

Bit

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

P24	P25	P26	P27	P28	P29	P30	P31	P16	P17	P18	P19	P20	P21	P22	P23
P08	P09	P10	P11	P12	P13	P14	P15	P00	P01	P02	P03	P04	P05	P06	P07

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=10

Bit

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

P07	P06	P05	P04	P03	P02	P01	P00	P15	P14	P13	P12	P11	P10	P09	P08
P23	P22	P21	P20	P19	P18	P17	P16	P31	P30	P29	P28	P27	P26	P25	P24

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=11

Bit

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

P00	P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	P15
P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

**Mono or internal frame buffer 2-bpp or palette 2-bpp mode** : FBF = 0001

PS=00

Bit

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

P15	P14	P13	P12	P11	P10	P09	P08
P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=01

Bit

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

P12	P13	P14	P15	P08	P09	P10	P11
P04	P05	P06	P07	P00	P01	P02	P03

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=10

Bit

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

P03	P02	P01	P00	P07	P06	P05	P04
P11	P10	P09	P08	P15	P14	P13	P12

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=11

Bit

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

P00	P01	P02	P03	P04	P05	P06	P07
P08	P09	P10	P11	P12	P13	P14	P15

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

**Mono 4-bpp or palette 4-bpp mode** : FBF = 0010

PS=00

Bit

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

P07					P06					P05					P04				
P03					P02					P01					P00				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=01

Bit

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

P06					P07					P04					P05				
P02					P03					P00					P01				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=10

Bit

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

P01					P00					P03					P02				
P05					P04					P07					P06				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=11

Bit

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

P00					P01					P02					P03				
P04					P05					P06					P07				

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

**Mono 8-bpp mode or palette 8-bpp mode** : FBF = 0011

PS=00/11

Bit

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

P3										P2					
P1										P0					

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=01/10

Bit

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

P0										P1					
P2										P3					

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

**Color 16-bpp mode** : FBF = 0100 or 0101 or 0110 or 0111 or 1000

PS=00

Bit

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

P1															
P0															

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=01

Bit

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



15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

PS=10/11

Invalid

**Color 24-bpp or 32-bpp mode** : FBF = 1001 or 1010

PS=00/01

Bit

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



15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

The bytes sequence is ARGB

PS=10/11

Bit

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



15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

The bytes sequence is BGRA

**7.4.4.16. DE-HWC coordinate control register**

Offset: 0x8D8		Register Name: DEBE_HWCCTL_REG	
Bit	Read/Writ	Default/H	Description

	e	ex	
31:16	R/W	UDF	HWC_YCOOR Hardware cursor Y coordinate
15:0	R/W	UDF	HWC_XCOOR Hardware cursor X coordinate

#### 7.4.4.17. DE-HWC frame buffer format register

Offset: 0x8E0			Register Name: DEBE_HWCFBCTL_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:24	R/W	UDF	HWC_YCOOROFF Y coordinate offset  The hardware cursor is 32*32 2-bpp pattern, this value represent the start position of the cursor in Y coordinate
23:16	R/W	UDF	HWC_XCOOROFF X coordinate offset  The hardware cursor is 32*32 2-bpp pattern, this value represent the start position of the cursor in X coordinate
15:6	/	/	/
5:4	R/W	UDF	HWC_YSIZE Y size control  00: 32pixels per line  01: 64pixels per line

			10:128pixels per line 11:256pixels per line
3:2	R/W	UDF	HWC_XSIZE  X size control  00: 32pixels per row  01: 64pixels per row  10:128pixels per row  11:256pixels per row
1:0	R/W	UDF	HWC_FBFMT  Pixels format control  00: 1bpp  01: 2bpp  10: 4bpp  11: 8bpp

#### 7.4.4.18. DE backend input YUV channel control register

Offset: 0x920			Register Name: DEBE_IYUVCTL_REG
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14:12	R/W	UDF	IYUV_FBFMT

			Input data format  000: planar YUV 411  001: planar YUV 422  010: planar YUV 444  011: interleaved YUV 422  100: interleaved YUV 444  Other: illegal
11:10	/	/	/
9:8	R/W	UDF	IYUV_FBPS  Pixel sequence  In planar data format mode:  00: Y3Y2Y1Y0  01: Y0Y1Y2Y3 (the other 2 components are same)  Other: illegal  In interleaved YUV 422 data format mode:  00: UYVY  01: YUYV  10: VYUY  11: YVYU



			In interleaved YUV 444 data format mode:  00: AYUV  01: VUYA  Other: illegal
7:5	/	/	/
4	R/W	UDF	IYUV_LINNEREN  0:  linner  1:
3:1	/	/	/
0	R/W	UDF	IYUV_EN  YUV channel enable control  0: disable  1: enable

**Source data input data ports:**

Input buffer channel	Planar YUV	Interleaved YUV
Channel0	Y	YUV
Channel1	U	-
Channel2	V	-

**7.4.4.19. DE backend YUV channel frame buffer low 32 bit address register**

Offset:			Register Name: DEBE_IYUVLADD_REG
Channel 0 : 0x930			
Channel 1 : 0x934			
Channel 2 : 0x938			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	IYUV_LADD  Buffer Address  Frame buffer address in BYTE

**7.4.4.20. DE backend YUV channel frame buffer high 8 bit address register**

Offset: 0x93C			Register Name: DEBE_IYUVHADD_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	UDF	V_HADD  V channel buffer address  Frame buffer address in BYTE
15:8	R/W	UDF	U_HADD  U channel buffer address  Frame buffer address in BYTE

7:0	R/W	UDF	Y_HADD  Y channel buffer address  Frame buffer address in BYTE
-----	-----	-----	--

#### 7.4.4.21. DE backend YUV channel buffer line width register

Offset:			Register Name: DEBE_IYUVLINEWIDTH_REG
Channel 0 : 0x940			
Channel 1 : 0x944			
Channel 2 : 0x948			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	IYUV_LINEWIDTH  Line width  The width is the distance from the start of one line to the start of the next line.  Description in bits

#### 7.4.4.22. E backend output color control register

Offset: 0x9C0			Register Name: DEBE_OCCTL_REG
Bit	Read/Write	Default/Hex	Description
	e	ex	
31:1	/	/	/
0	R/W	UDF	OC_EN

			Color control module enable control  0: disable  1: enable
--	--	--	--

**Color correction conversion algorithm formula:**

<p>R =</p> <p>(R R component coefficient * R) +</p> <p>(R G component coefficient * G) +</p> <p>(R B component coefficient * B) +</p> <p>R constant</p> <p>G =</p> <p>(G R component coefficient * R) +</p> <p>(G G component coefficient * G) +</p> <p>(G B component coefficient * B) +</p> <p>G constant</p>
--

B =  
  
 (B R component coefficient \* R) +  
  
 (B G component coefficient \* G) +  
  
 (B B component coefficient \* B) +  
  
 B constant

#### 7.4.4.23. DE backend output color R coefficient register

Offset:			Register Name: DEBE_OCRCOEF_REG
R component: 0x9D0			
G component: 0x9D4			
B component: 0x9D8			
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	UDF	OC_RCOEF  the R coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.4.4.24. DE backend output color R constant register

Offset: 0x9DC	Register Name: DEBE_OCRCONS_REG
---------------	---------------------------------

Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14:0	R/W	UDF	OC_RCONS  the R constant  the value equals to coefficient*2 <sup>4</sup>

#### 7.4.4.25. DE backend output color G coefficient register

Offset:		Register Name: DEBE_OCGCOEF_REG	
R component: 0x9E0			
G component: 0x9E4			
B component: 0x9E8			
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	UDF	OC_GCOEF  the G coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.4.4.26. DE backend output color G constant register

Offset: 0x9EC		Register Name: DEBE_OCGCONS_REG	
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/

14:0	R/W	UDF	OC_GCONS  the G constant  the value equals to coefficient*2 <sup>4</sup>
------	-----	-----	--

#### 7.4.4.27. DE backend output color B coefficient register

Offset:			Register Name: DEBE_OCBCOEF_REG
G/Y component: 0x9F0			
R/U component: 0x9F4			
B/V component: 0x9F8			
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	UDF	OC_BCOEF  the B coefficient  the value equals to coefficient*2 <sup>10</sup>

#### 7.4.4.28. DE backend output color B constant register

Offset: 0x9FC			Register Name: DEBE_OCBCONS_REG
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14:0	R/W	UDF	OC_BCONS

			the B constant
			the value equals to coefficient*2 <sup>4</sup>

**7.4.4.29. DE-HWC pattern memory block**

**Function:**

1bpp:

Bit

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

P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16
P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

2bpp:

Bit

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

P15	P14	P13	P12	P11	P10	P09	P08
P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

4bpp:

Bit

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



P07	P06	P05	P04
P03	P02	P01	P00
15 14 13 12 11	10 09 08 07 06	05 04 03 02 01	00

8bpp:

Bit

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

P3	P2
P1	P0
15 14 13 12 11 10 09 08 07 06	05 04 03 02 01 00

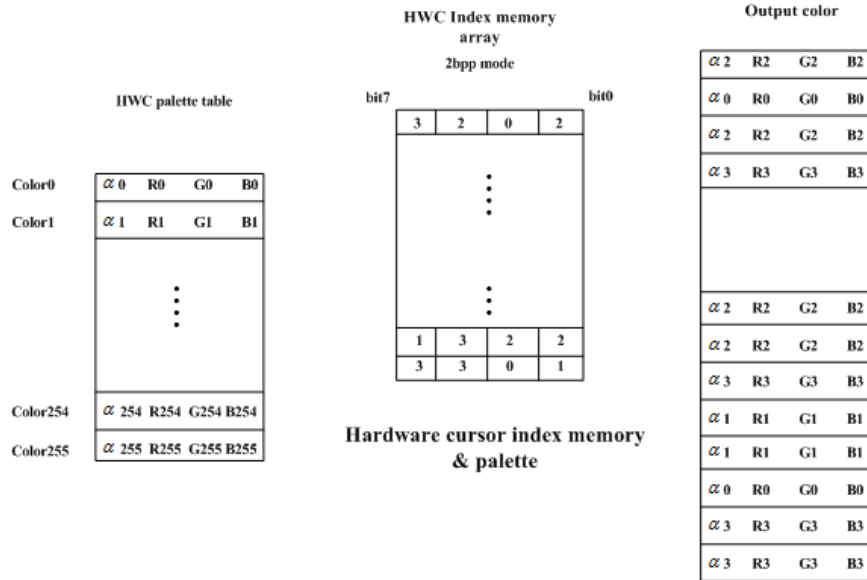
Offset:			DE-HW cursor pattern memory block
0x4800-0x4BFF			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	<p>Hardware cursor pixel pattern</p> <p>Specify the color displayed for each of the hardware cursor pixels.</p>

#### 7.4.4.30. DE-HWC palette table

Offset:			DE-HW palette table
0x4C00-0x4FFF			
Bit	Read/Write	Default/Hex	Description

	e	ex	
31:24	R/W	UDF	Alpha value
23:16	R/W	UDF	Red value
15:8	R/W	UDF	Green value
7:0	R/W	UDF	Blue value

The follow figure (only with 2bpp mode) shows the RAM array used for hardware cursor palette lookup and the corresponding colors output.



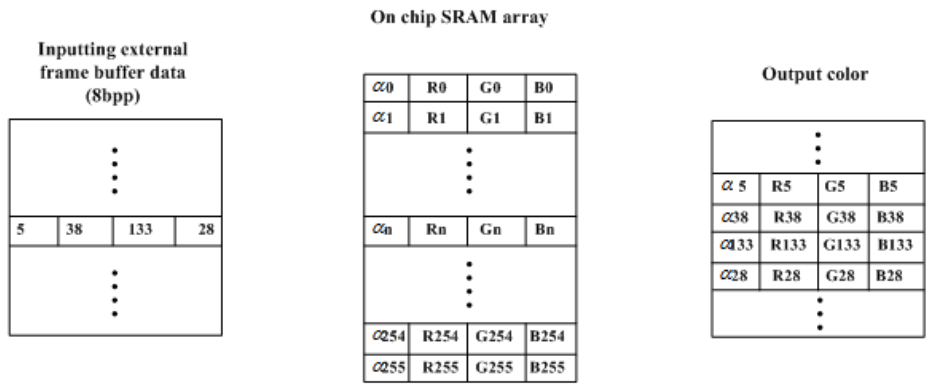
**7.4.4.31. Palette mode**

Offset:	Pipe palette color table SRAM block
---------	-------------------------------------

Pipe0:0x5000-0x53FF			
Pipe1:0x5400-0x57FF			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	UDF	Alpha value
23:16	R/W	UDF	Red value
15:8	R/W	UDF	Green value
7:0	R/W	UDF	Blue value

In this mode, RAM array is used for palette lookup table, each pixel in the layer frame buffer is treated as an index into the RAM array to select the actual color.

The follow figure shows the RAM array used for palette lookup and the corresponding colors output.



**On chip SRAM for palette lookup**

**7.4.4.32. Internal frame buffer mode:**

In internal frame buffer mode, the RAM array is used as an on-chip frame buffer, each pixel in the RAM array is used to select one of the palette 32-bit colors.

1bpp:

Bit

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

P31	P30	P29	P28	P27	P26	P25	P24	P23	P22	P21	P20	P19	P18	P17	P16
P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

2bpp:

Bit

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

P15	P14	P13	P12	P11	P10	P09	P08
P07	P06	P05	P04	P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

4bpp:

Bit

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

P07	P06	P05	P04
P03	P02	P01	P00

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

8bpp:

Bit

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

P3	P2
P1	P0

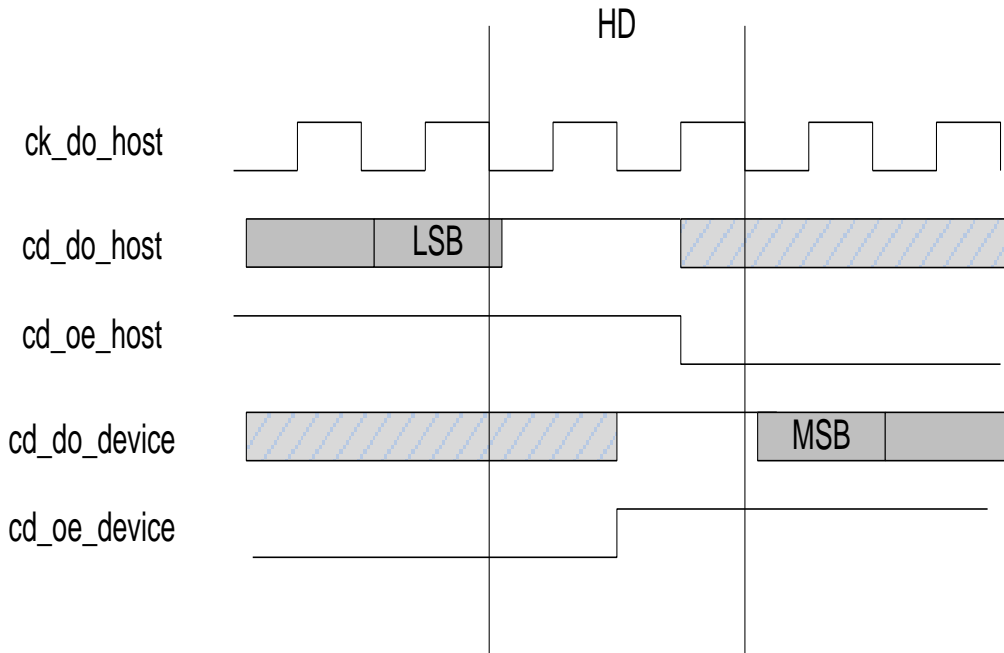
15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00

Offset:			DE-on chip SRAM block
0x4000-0x57FF			
Bit	Read/Write	Default/Hex	Description
31:0	R/W	UDF	Internal frame buffer pixel pattern  Specify the color displayed for each of the internal frame buffer pixels.

#### 7.4.4.33. Internal frame buffer mode palette table

Address:			Pipe palette table
Pipe0:0x5000-0x53FF			
Pipe1:0x5400-0x57FF			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	UDF	Alpha value
23:16	R/W	UDF	Red value
15:8	R/W	UDF	Green value
7:0	R/W	UDF	Blue value

The follow figure shows the RAM array used for internal frame buffer mode and the corresponding colors output.



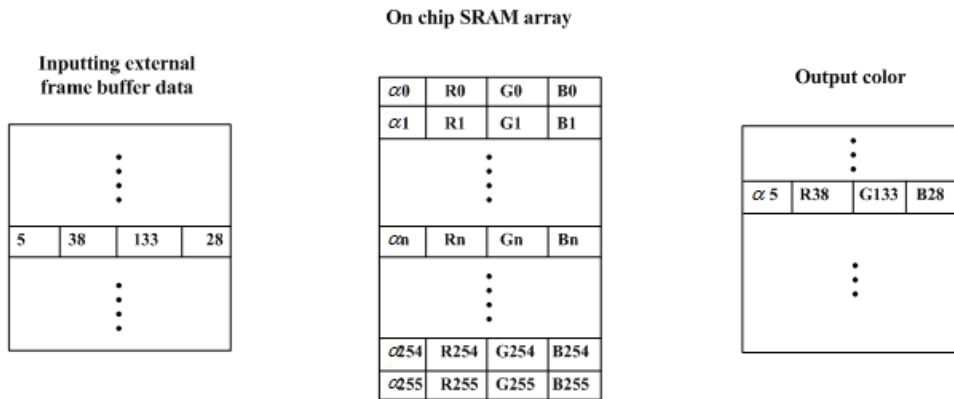
**7.4.4.34. Gamma correction mode**

Offset:			DE-on chip SRAM block
0x4400-0x47FF			
Bit	Read/Write	Default/Hex	Description
31:24	R/W	UDF	Alpha channel intensity
23:16	R/W	UDF	Red channel intensity
15:8	R/W	UDF	Green channel intensity
7:0	R/W	UDF	Blue channel intensity

In gamma correction mode, the RAM array is used for gamma correction, each pixel's alpha, red, green, and blue color

component is treated as an index into the SRAM array. The corresponding alpha, red, green, or blue channel intensity value at that index is used in the actual color.

The follow figure shows the RAM array used for gamma correction and the corresponding colors output.



**On chip SRAM for gamma correction**

### 7.4.5. Display engine memory mapping

Base Address:

Offset:

0x0000 0x07FF 0x0800	<b>Reserved</b>
0x0DFF 0x0E00	<b>Registers</b>
0x3FFF 0x4000	<b>Reserved</b>
0x43FF 0x4400	<b>Reserved</b>
0x47FF 0x4800	<b>Gamma Table</b>
0x4BFF 0x4C00	<b>HWC Memory Block</b>
0x4FFF 0x5000	<b>HWC Palette Table</b>
0x53FF 0x5400	<b>PIPE0 Palette Table</b>
0x57FF 0x5800	<b>PIPE1 Palette Table</b>
0xFFFF	<b>Reserved</b>



## 7.5. CMU

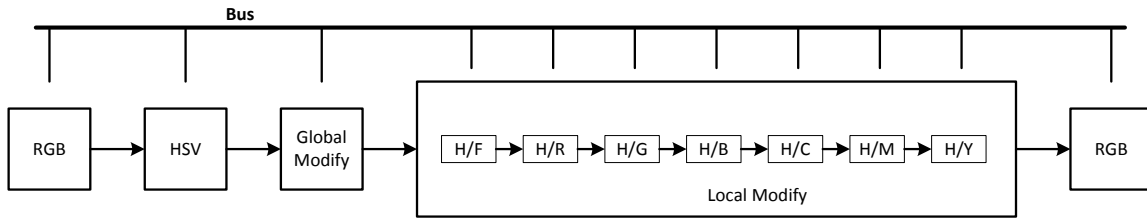
### 7.5.1. Overview

Color Management Unit (CMU) is capable of color adjustment to provide more vivid vision effects, and it is also capable of skin tones enhancement.

The CMU features:

- Support RGB888 input and output format
- Support window clipping up to 8192x8192 pixels
- Support global and local adjustment for hue/ saturation/ brightness in HSV space
- Support red/green/blue/cyan/magenta/yellow/flesh areas modifying in local adjustment mode

### 7.5.2. Block Diagram



Color Management Unit General Diagram

## 7.6. DEU

### 7.6.1. Overview

- Require planar YUV444 color space input
- Support planar YUV444/RGB888 color space output
- Support input/output size up to 4096×4096
- 2D Luma peaking for luminance channel for maximum 2048 pixel/line input
- 1D Luma peaking for luminance channel for input 2048 pixel/line above
- 1D dynamic color transient improvement for two chrominance channels
- White level expansion/ Black level expansion for luminance channel

## 7.7. DRC

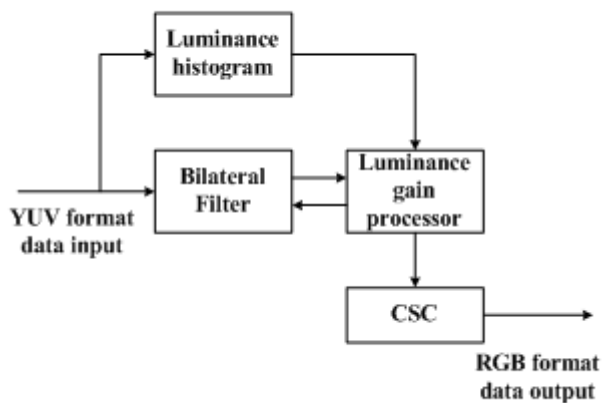
### 7.7.1. Overview

DRC (Dynamic Range Controller) adjusts the image mapping curve according to the histogram frame by frame. The control function can be defined by software drivers accordingly. A typical application is content-based backlight control.

It features:

- Support 4096x4096 input/output
- Support HISTOGRAM and DRC in YUV or HSV color space
- De-flicker function for low-end TV device

### 7.7.2. Block Diagram



## Chapter 8 Interfaces

This chapter describes the A80 interfaces, including:

- SD3.0
- TWI
- SPI
- UART
- RSB
- One wire interface
- CIR receiver
- USB
- Digital audio interface
- Transport stream
- Ethernet MAC

## 8.1. SD3.0

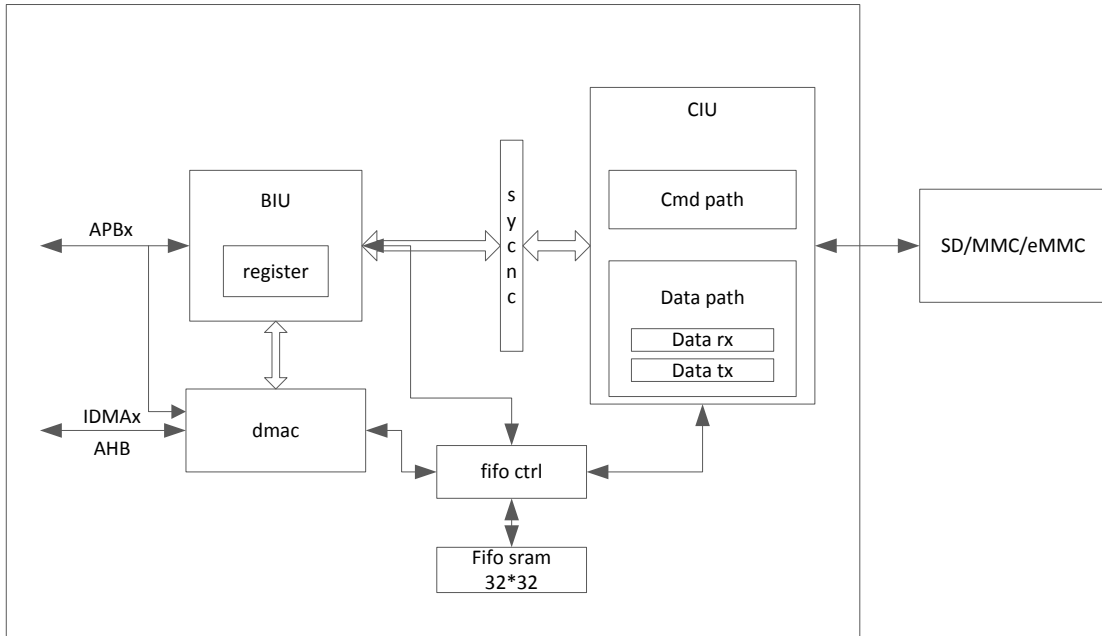
### 8.1.1. Overview

The SD3.0 controller can be configured either as a Secure Digital Multimedia Card controller, which simultaneously supports Secure Digital I/O (SDIO), Multimedia Cards (MMC), eMMC Card.

The SD3.0 controller includes the following features:

- Supports Secure Digital memory protocol commands (up to SD3.0)
- Supports Secure Digital I/O protocol commands (up to SDIO 2.0)
- Supports Multimedia Card protocol commands (up to eMMC4.5)
- Supports eMMC boot operation
- Supports one SD (Version 1.0 to 3.0) or MMC (Version 3.3 to eMMC4.5)
- Supports hardware CRC generation and error detection
- Supports host pull-up control
- Supports SDIO interrupts in 1-bit and 4-bit modes
- Supports SDIO suspend and resume operation
- Supports SDIO read wait
- Supports block size of 1 to 65535 bytes
- Supports descriptor-based internal DMA controller
- Internal 32x32-bit (128 bytes total) FIFO for data transfer
- Supports 3.3 V and 1.8V IO pad

### 8.1.2. Block Diagram



(BIU: Bus Interface Unit; CIU: Card Interface Unit)

SD3.0 Controller Block Diagram

### 8.1.3. SD3.0 Timing Diagram

Please refer to relative specifications:

- Physical Layer Specification Ver3.00 Final, 2009.04.16
- SDIO Specification Ver2.00
- Multimedia Cards (MMC – version 4.2)
- JEDEC Standard – JESD84-B45, EMBEDDED MULTI-MEDIA CARD (e•MMC), ELECTRICAL STANDARD (4.5 Device)



#### 8.1.4. SD3.0 Register List

Module Name	Base Address
SDC0	0x01C0F000
SDC1	0x01C10000
SDC2	0x01C11000
SDC3	0x01C12000
SD/MMC-COMM	0x01C13000

Register Name	Offset	Description
SD_CTRL	0x00	Control register
SD_CLKDIV	0x04	Clock Control register
SD_TMOUT	0x08	Time out register
SD_CTYPE	0x0C	Bus Width register
SD_BLKSIZE	0x10	Block size register
SD_BYTCNT	0x14	Byte count register
SD_CMD	0x18	Command register
SD_CMDARG	0x1c	Command argument register
SD_RESP0	0x20	Response 0 register
SD_RESP1	0x24	Response 1 register
SD_RESP2	0x28	Response 2 register
SD_RESP3	0x2C	Response 3 register
SD_INTMASK	0x30	Interrupt mask register
SD_MINTSTS	0x34	Masked interrupt status register
SD_RINTSTS	0x38	Raw interrupt status register
SD_STATUS	0x3C	Status register
SD_FIFOTH	0x40	FIFO Water Level register
SD_FUNS	0x44	FIFO Function Select register
SD_A12A	0x58	Auto command 12 argument
SD_HWRST	0x78	Hardware Reset Register
SD_DMAC	0x80	BUS Mode Control
SD_DLBA	0x84	Descriptor List Base Address
SD_IDST	0x88	DMAC Status
SD_IDIE	0x8c	DMAC Interrupt Enable
SD_CHDA	0x90	Current host descriptor address
SD_CBDA	0x94	Current buffer descriptor address
SD_THLD	0x100	Card Threshold Control register
EMMC_DDR_SBIT_DET	0x10c	eMMC4.5 DDR Start Bit Detection Control
SD_FIFO	0x200	Read/ Write FIFO

### 8.1.5. SD3.0 Register Description

#### 8.1.5.1. SD Global Control Register

Offset: 0x0000			Register Name: SD_CTRL
			Default Value: 0x0000_0300
Bit	Read/Write	Default	Description
31	R/W	0	FIFO_AC_MOD FIFO Access Mode 1-AHB bus 0-DMA bus
30:11	-	-	/
10	R/W	0	DDR_MOD_SEL DDR Mode Select 0 – SDR mode 1 – DDR mode
9	-	-	reserved
8	R/W	1	CD_DBC_ENB Card Detect (Data[3] status) De-bounce Enable 0 - disable de-bounce 1 – enable de-bounce
7:6	-	-	/
5	R/W	0	DMA_ENB DMA Global Enable 0 – Disable DMA to transfer data, using AHB bus 1 – Enable DMA to transfer data
4	R/W	0	INT_ENB Global Interrupt Enable 0 – Disable interrupts

			1 – Enable interrupts
3	-	-	/
2	R/W	0	DMA_RST DMA Reset
1	R/W	0	FIFO_RST FIFO Reset 0 – No change 1 – Reset FIFO <i>This bit is auto-cleared after completion of reset operation.</i>
0	R/W	0	SOFT_RST Software Reset 0 – No change 1 – Reset SD/MMC controller <i>This bit is auto-cleared after completion of reset operation.</i>

### 8.1.5.2. SD Clock Control Register

Offset: 0x0004			Register Name: SD_CLKDIV Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:18	/	/	/
17	R/W	0	CCLK_CTRL Card Clock Output Control 0 – Card clock always on 1 – Turn off card clock when FSM in IDLE state
16	R/W	0	CCLK_ENB Card Clock Enable

			0 – Card Clock off 1 – Card Clock on
15:8	/	/	/
7:0	R/W	0	CCLK_DIV Card clock divider n – Source clock is divided by 2*n.(n=0~255)

### 8.1.5.3. SD Timeout Register

<b>Offset: 0x0008</b>			<b>Register Name: SD_TMOUT</b> <b>Default Value: 0xFFFF_FF40</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:8	R/W	0xffffffff	DTO_LMT Data Timeout Limit
7:0	R/W	0x40	RTO_LMT Response Timeout Limit

### 8.1.5.4. SD Bus Width Register

<b>Offset: 0x000c</b>			<b>Register Name: SD_CTYPE</b> <b>Default Value: 0x0000_0000</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:2	/	/	/
1:0	R/W	0	CARD_WID Card width 2'b00 – 1-bit width 2'b01 – 4-bit width 2'b1x – 8-bit width

**8.1.5.5. SD Block Size Register**

Offset: 0x0010			Register Name: SD_BLKSIZE Default Value: 0x0000_0200
Bit	Read/Write	Default	Description
31:16	/	/	/
15:0	R/W	0x200	BLK_SZ Block size

**8.1.5.6. SD Byte Count Register**

Offset: 0x0014			Register Name: SD_BYTCNT Default Value: 0x0000_0200
Bit	Read/Write	Default	Description
31:0	R/W	0x200	BYTE_CNT Byte counter Number of bytes to be transferred; should be integer multiple of Block Size for block transfers.

**8.1.5.7. SD Command Register**

Offset: 0x0018			Register Name: SD_CMD Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31	R/W	0	CMD_LOAD Start Command. This bit is auto cleared when current command is sent. If there is no any response error happened, a command complete interrupt bit (CMD_OVER) will be set in interrupt register. You should not write any other command before this bit is cleared, or a command busy interrupt bit (CMD_BUSY) will be set in interrupt register.

30	/	/	/
29	R/W	0	Use Hold Register 0 - CMD and DATA sent to card bypassing HOLD Register 1 - CMD and DATA sent to card through the HOLD Register
28	R/W	0	VOL_SW Voltage Switch 0 – normal command 1 – Voltage switch command, set for CMD11 only
27	R/W	0	BOOT_ABT Boot Abort Setting this bit will terminate the boot operation.
26	R/W	0	EXP_BOOT_ACK Expect Boot Acknowledge. When Software sets this bit along in mandatory boot operation, controller expects a boot acknowledge start pattern of 0-1-0 from the selected card.
25:24	R/W	0	BOOT_MOD Boot Mode 2'b00 – normal command 2'b01 - Mandatory Boot operation 2'b10 - Alternate Boot operation 2'b11 - reserved
23	R/W	0	CCS_EXP ccs_expected 0 – Interrupts are not enabled in CE-ATA device (nIEN = 1 in ATA control register), or command does not expect CCS from device 1 – Interrupts are enabled in CE-ATA device (nIEN = 0), and RW_BLK command expects command completion signal from CE-ATA device If the command expects Command Completion Signal (CCS) from the CE-ATA device, the software should set this control bit. SD/MMC sets Data Transfer Over bit in RINTSTS register and generates interrupt to

			host if Data Transfer Over interrupt is not masked.
22	R/W	0	<p>RD_CEATA_DEV</p> <p>read_ceata_device</p> <p>0 – Host is not performing read access (RW_REG or RW_BLK) towards CE-ATA device</p> <p>1 – Host is performing read access (RW_REG or RW_BLK) towards CE-ATA device</p> <p>Software should set this bit to indicate that CE-ATA device is being accessed for read transfer. This bit is used to disable read data timeout indication while performing CE-ATA read transfers. Maximum value of I/O transmission delay can be no less than 10 seconds. SD/MMC should not indicate read data timeout while waiting for data from CE-ATA device.</p>
21	R/W	0	<p>PRG_CLK</p> <p>Change Clock</p> <p>0 – Normal command</p> <p>1 – Change Card Clock; when this bit is set, controller will change clock domain and clock output. No command will be sent.</p>
20:16	-	-	/
15	R/W	0	<p>SEND_INIT_SEQ</p> <p>Send Initialization</p> <p>0 – normal command sending</p> <p>1 – Send initialization sequence before sending this command.</p>
14	R/W	0	<p>STOP_ABT_CMD</p> <p>Stop Abort Command</p> <p>0 – normal command sending</p> <p>1 – send Stop or abort command to stop current data transfer in progress.(CMD12, CMD52 for writing “I/O Abort” in SDIO CCCR)</p>
13	R/W	0	<p>WAIT_PRE_OVER</p> <p>Wait Data Transfer Over</p> <p>0 – Send command at once, do not care of data transferring</p> <p>1 – Wait for data transfer completion before sending current command</p>

12	R/W	0	<p>STOP_CMD_FLAG</p> <p>Send Stop CMD Automatically (CMD12)</p> <p>0 – Do not send stop command at end of data transfer</p> <p>1 – Send stop command automatically at end of data transfer</p>
11	R/W	0	<p>TRANS_MODE</p> <p>Transfer Mode</p> <p>0 – Block data transfer command</p> <p>1 – Stream data transfer command</p>
10	R/W	0	<p>TRANS_DIR</p> <p>Transfer Direction</p> <p>0 – Read operation</p> <p>1 – Write operation</p>
9	R/W	0	<p>DATA_TRANS</p> <p>Data Transfer</p> <p>0 – without data transfer</p> <p>1 – with data transfer</p>
8	R/W	0	<p>CHK_RESP_CRC</p> <p>Check Response CRC</p> <p>0 – Do not check response CRC</p> <p>1 – Check response CRC</p>
7	R/W	0	<p>LONG_RESP</p> <p>Response Type</p> <p>0 –Short Response (48 bits)</p> <p>1 –Long Response (136 bits)</p>
6	R/W	0	<p>RESP_RCV</p> <p>Response Receive</p> <p>0 – Command without Response</p> <p>1 – Command with Response</p>
5:0	R/W	0	<p>CMD_IDX</p>



			CMD Index Command index value
--	--	--	----------------------------------

#### 8.1.5.8. SD Command Argument Register

<b>Offset: 0x001c</b>			<b>Register Name: SD_CMDARG</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	CMD_ARG Command argument

#### 8.1.5.9. SD Response 0 Register

<b>Offset: 0x0020</b>			<b>Register Name: SD_RESP0</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R	0	CMD_RESP0 response 0 Bit[31:0] of response

#### 8.1.5.10. SD Response 1 Register

<b>Offset: 0x0024</b>			<b>Register Name: SD_RESP1</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R	0	CMD_RESP1 response 1 Bit[63:31] of response

**8.1.5.11. SD Response 2 Register**

Offset: 0x0028			Register Name: SD_RESP2 Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	R	0	CMD_RESP2 response 2 Bit[95:64] of response

**8.1.5.12. SD Response 3 Register**

Offset: 0x002C			Register Name: SD_RESP3 Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	R	0	CMD_RESP3 response 3 Bit[127:96] of response

**8.1.5.13. SD Interrupt Mask Register**

Offset: 0x0030			Register Name: SD_INTMASK Default Value: 0xc000_0000
Bit	R/W	Default	Description
31:0	R/W	0	INT_MASK 0 – interrupt masked 1 – interrupt enabled  Bit field defined as following:  bit 31– card removed

		bit 30 – card inserted bit 17~29 - reserved bit 16 – SDIO interrupt bit 15 – Data End-bit error bit 14 – Auto Stop Command done bit 13 – Data Start Error bit 12 – Command Busy and illegal write bit 11 – FIFO under run/overflow bit 10 – Data starvation timeout /V1.8 Switch Done bit 9 – Data timeout/Boot data start bit 8 – Response timeout/Boot ACK received bit 7 – Data CRC error bit 6 – Response CRC error bit 5 – Data Receive Request bit 4 –Data Transmit Request bit 3 – Data Transfer Complete bit 2 – Command Complete bit 1 – Response Error (no response or response CRC error) bit 0 – Reserved
--	--	---

#### 8.1.5.14. SD Masked Interrupt Status Register

Offset: 0x0034			Register Name: SD_MINTSTS
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	R	0	MSKD_ISTA Interrupt status. Enabled only if corresponding bit in mask register is set.

			Bit field defined as following:  bit 31 – card removed  bit 30 – card inserted  bit 17~29 - reserved  bit 16 – SDIO interrupt  bit 15 – Data End-bit error  bit 14 – Auto command done  bit 13 – Data Start Error  bit 12 – Command Busy and illegal write  bit 11 – FIFO under run/overflow  bit 10 – Data starvation timeout (HTO)/V1.8 Switch Done  bit 9 – Data timeout/Boot data start  bit 8 – Response timeout/Boot ACK received  bit 7 – Data CRC error  bit 6 – Response CRC error  bit 5 – Data Receive Request  bit 4 –Data Transmit Request  bit 3 – Data Transfer Complete  bit 2 – Command Complete  bit 1 – Response Error (no response or response CRC error)  bit 0 – Reserved
--	--	--	---

#### 8.1.5.15. SD Raw Interrupt Status Register

Offset: 0x0038			Register Name: SD_RINTSTS
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	R/W	0	RAW_ISTA Raw Interrupt Status.

		<p><i>This is write-1-to-clear bits.</i></p> <p>Bit field defined as following:</p> <ul style="list-style-type: none"> <li>bit 31 – card removed</li> <li>bit 30 – card inserted</li> <li>bit 17~29 - reserved</li> <li>bit 16 – SDIO interrupt</li> <li>bit 15 – Data End-bit error</li> <li>bit 14 – Auto command done</li> <li>bit 13 – Data Start Error</li> <li>bit 12 – Command Busy and illegal write</li> <li>bit 11 – FIFO under run/overflow</li> <li>bit 10 – Data starvation timeout (HTO)/V1.8 Switch Done</li> <li>bit 9 – Data timeout/Boot data start</li> <li>bit 8 – Response timeout/Boot ACK received</li> <li>bit 7 – Data CRC error</li> <li>bit 6 – Response CRC error</li> <li>bit 5 – Data Receive Request</li> <li>bit 4 –Data Transmit Request</li> <li>bit 3 – Data Transfer Complete</li> <li>bit 2 – Command Complete</li> <li>bit 1 – Response Error (no response or response CRC error)</li> <li>bit 0 – Reserved</li> </ul>
--	--	--

#### 8.1.5.16. SD Status Register

<b>Offset: 0x003c</b>			<b>Register Name: SD_STATUS</b>
			<b>Default Value: 0x0000_0006</b>
Bit	Read/Write	Default	Description

31	R	0	DMA_REQ dma_req DMA request signal state
30:23	/	/	/
22:17	R	0	FIFO_LEVEL FIFO Level Number of filled locations in FIFO
16:11	R	0	RESP_IDX Response Index Index of previous response, including any auto-stop sent by controller
10	R	0	FSM_BUSY Data FSM Busy Data transmit or receive state-machine is busy
9	R	0	CARD_BUSY Card data busy Inverted version of DATA[0] 0 – card data not busy 1 – card data busy
8	R	0	CARD_PRESENT Data[3] status level of DATA[3]; checks whether card is present 0 – card not present 1 – card present
7:4	R	0	FSM_STA Command FSM states: 0 – Idle 1 – Send init sequence 2 – Tx cmd start bit 3 – Tx cmd tx bit

			4 – Tx cmd index + arg 5 – Tx cmd crc7 6 – Tx cmd end bit 7 – Rx resp start bit 8 – Rx resp IRQ response 9 – Rx resp tx bit 10 – Rx resp cmd idx 11 – Rx resp data 12 – Rx resp crc7 13 – Rx resp end bit 14 – Cmd path wait NCC 15 – Wait; CMD-to-response turnaround
3	R	0	FIFO_FULL FIFO full 1 – FIFO full 0 – FIFO not full
2	R	1	FIFO_EMPTY FIFO Empty 1 - FIFO Empty 0 - FIFO not Empty
1	R	1	FIFO_TX_LEVEL FIFO TX Water Level flag 0 – FIFO didn't reach transmit trigger level 1 - FIFO reached transmit trigger level
0	R	0	FIFO_RX_LEVEL FIFO RX Water Level flag 0 – FIFO didn't reach receive trigger level 1 - FIFO reached receive trigger level

**8.1.5.17. SD FIFO Water Level Register**

Offset: 0x0040			Register Name: SD_FIFOTH
			Default Value: 0x000F_0000
Bit	Read/Write	Default	Description
31	/	/	/
30:28	R/W	0	<p>BSIZE_OF_TRANS</p> <p>Burst size of multiple transaction</p> <p>000 – 1 transfers</p> <p>001 – 4</p> <p>010 – 8</p> <p>011 – 16</p> <p>100 – 32</p> <p>101 – 64</p> <p>110 – 128</p> <p>111 – 256</p> <p>Should be programmed same as DMA controller multiple transaction size. The units for transfers are the DWORD. A single transfer would be signaled based on this value. Value should be sub-multiple of (RX_TL + 1) and (FIFO_DEPTH - TX_TL)</p> <p>Recommended:</p> <p>MSize = 8, TX_TL = 16, RX_TL = 15</p>
27:21	R	0	/
20:16	R/W	0xF	<p>RX_TL</p> <p>Rx Trigger Level</p> <p>0x0~0x1e – RX Trigger Level is 0~30</p> <p>0x1f – reserved</p> <p>FIFO threshold when FIFO request host to receive data from FIFO. When FIFO data level is greater than this value, DMA is request is raised if DMA enabled, or RX interrupt bit is set if interrupt enabled. At the end of packet, if the last transfer is less than this level, the value is ignored</p>



			and relative request will be raised as usual. Recommended: 15 (means greater than 15)
15:5	R	0	/
4:0	R/W	0	<p>TX_TL TX Trigger Level 0x1~0xf – TX Trigger Level is 1~31 0x0 – no trigger</p> <p>FIFO threshold when FIFO requests host to transmit data to FIFO. When FIFO data level is less than or equal to this value, DMA TX request is raised if DMA enabled, or TX request interrupt bit is set if interrupt enabled. At the end of packet, if the last transfer is less than this level, the value is ignored and relative request will be raised as usual.</p> <p>Recommended: 16 (means less than or equal to 16)</p>

#### 8.1.5.18. SD Function Select Register

Offset: 0x0044			Register Name: SD_CTRL Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:16	R/W	0	<p>CEATA_EN CEATA Support ON/OFF 0xceaa – CEATA support on. All hidden CEATA relative bits are accessible normally and these 16 bits return value of 0x1 when be read. Other value – CEATA support off. All hidden CEATA relative bits cannot be access and these 16 bits return value of 0 when be read.</p>
15:11	-	0	/
10	R/W	0	<p>CEATA_ISTA ceata device interrupt status 0 – Interrupts not enabled in CE-ATA device (nIEN = 1 in ATA control register)</p>

			<p>1 – Interrupts are enabled in CE-ATA device (nIEN = 0 in ATA control register)</p> <p>Software should appropriately write to this bit after power-on reset or any other reset to CE-ATA device. After reset, usually CE-ATA device interrupt is disabled (nIEN = 1). If the host enables CE-ATA device interrupt, then software should set this bit.</p>
9	R/W	0	<p>SEND_AUTO_STOP_CCSD</p> <p>send auto stop ccscd</p> <p>0 – Clear bit if SD/MMC does not reset the bit.</p> <p>1 – Send internally generated STOP after sending CCSD to CEATA device.</p> <p>When set, SD/MMC automatically sends internally generated STOP command (CMD12) to CE-ATA device. After sending internally-generated STOP command, Auto Command Done (ACD) bit in RINTSTS is set and generates interrupt to host if Auto Command Done interrupt is not masked. After sending the CCSD, SD/MMC automatically clears send_auto_stop_ccscd bit.</p>
8	R/W	0	<p>SEND_CCSD</p> <p>send ccscd</p> <p>0 – Clear bit if SD/MMC does not reset the bit.</p> <p>1 – Send Command Completion Signal Disable (CCSD) to CE-ATA device</p> <p>When set, SD/MMC sends CCSD to CE-ATA device. Software sets this bit only if current command is expecting CCS (that is, RW_BLK) and interrupts are enabled in CE-ATA device. Once the CCSD pattern is sent to device, SD/MMC automatically clears send_ccscd bit. It also sets Command Done (CD) bit in RINTSTS register and generates interrupt to host if Command Done interrupt is not masked.</p>
7:3	-	-	/
2	R/W	0	<p>ABT_RDATA</p> <p>Abort Read Data</p> <p>0 – Ignored</p> <p>1 –After suspend command is issued during read-transfer, software polls card to find when suspend happened. Once suspend occurs, software sets bit to reset data state-machine, which is waiting for</p>

			next block of data.  Used in SDIO card suspends sequence.  <i>This bit is auto-cleared once controller reset to idle state.</i>
1	R/W	0	READ_WAIT  Read Wait  0 – Clear SDIO read wait  1 – Assert SDIO read wait
0	R/W	0	HOST_SEND_MMC_IRQRESQ  Host Send MMC IRQ Response  0 – Ignored  1 – Send auto IRQ response  When host is waiting MMC card interrupt response, setting this bit will make controller cancel wait state and return to idle state, at which time, controller will receive IRQ response sent by itself.  <i>This bit is auto-cleared after response is sent.</i>

#### 8.1.5.19. SD Auto Command 12 Argument Register

<b>Offset: 0x58</b>			<b>Register Name: SD_A12A</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	/	/	/
0:15	R/W	0xffff	SD_A12A.  SD_A12A set the argument of command 12 automatically send by controller

#### 8.1.5.20. SD Hardware Reset Register

<b>Offset: 0x78</b>	<b>Register Name: SD_HWRST</b>
---------------------	--------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:1	/	/	/
0	R/W	1	<p>HW_RESET.</p> <p>1 – Active mode</p> <p>0 – Reset</p> <p>These bits cause the cards to enter pre-idle state, which requires them to be re-initialized.</p>

#### 8.1.5.21. SD DMAC Control Register

Offset: 0x0080			Register Name: SD_BUS_MODE
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31	W	0	<p>DES_LOAD_CTRL</p> <p>When DMAC fetches a descriptor, if the valid bit of a descriptor is not set, DMAC FSM will go to the suspend state. Setting this bit will make DMAC re-fetch descriptor again and do the transfer normally.</p>
10:8	R	0	<p>PRG_BURST_LEN</p> <p>Programmable Burst Length.</p> <p>These bits indicate the maximum number of beats to be performed in one IDMAC transaction. The IDMAC will always attempt to burst as specified in PBL each time it starts a Burst transfer on the host bus. The permissible values are 1, 4, 8, 16, 32, 64, 128 and 256. This value is the mirror of MSIZE of FIFOTH register. In order to change this value, write the required value to FIFOTH register. This is an encode value as follows.</p> <p>000 – 1 transfers</p> <p>001 – 4 transfers</p> <p>010 – 8 transfers</p> <p>011 – 16 transfers</p> <p>100 – 32 transfers</p>

			<p>101 – 64 transfers</p> <p>110 – 128 transfers</p> <p>111 – 256 transfers</p> <p>Transfer unit is either 16, 32, or 64 bits, based on HDATA_WIDTH. PBL is a read-only value.</p>
7	R/W	0	<p>IDMAC_ENB</p> <p>IDMAC Enable.</p> <p>When set, the IDMAC is enabled. DE is read/write.</p>
6:2	R/W	0	<p>DES_SKIP_LEN</p> <p>Descriptor Skip Length.</p> <p>Specifies the number of Word to skip between two unchained descriptors. This is applicable only for dual buffer structure.</p> <p>Default value is set to 4 DWORD.</p>
1	R/W	0	<p>FIX_BUST_CTRL</p> <p>Fixed Burst.</p> <p>Controls whether the AHB Master interface performs fixed burst transfers or not. When set, the AHB will use only SINGLE, INCR4, INCR8 during start of normal burst transfers. When reset, the AHB will use SINGLE and INCR burst transfer operations.</p>
0	R/W	0	<p>IDMAC_RST</p> <p>DMA Reset.</p> <p>When set, the DMA Controller resets all its internal registers. SWR is read/write. It is automatically cleared after 1 clock cycle.</p>

#### 8.1.5.22. SD Descriptor List Base Address Register

<b>Offset: 0x0084</b>			<p><b>Register Name: SD_DLBA</b></p> <p><b>Default Value: 0x0000_0000</b></p>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:0	R/W	0	<p>DES_BASE_ADDR</p> <p>Start of Descriptor List.</p>

		Contains the base address of the First Descriptor. The LSB bits [1:0] are ignored and taken as all-zero by the IDMAC internally. Hence these LSB bits are read-only.
--	--	--

### 8.1.5.23. SD DMAC Status Register

Offset: 0x0088			Register Name: SD_IDST
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:17	/	/	/
16:13	R	0	DMAC_FSM_STA DMAC FSM present state. 0 – DMA_IDLE 1 – DMA_SUSPEND 2 – DESC_RD 3 – DESC_CHK 4 – DMA_RD_REQ_WAIT 5 – DMA_WR_REQ_WAIT 6 – DMA_RD 7 – DMA_WR 8 – DESC_CLOSE This bit is read-only.
12:10	R	0	DMAC_ERR_STA Error Bits. Indicates the type of error that caused a Bus Error. Valid only with Fatal Bus Error bit (IDSTS[2]) set. This field does not generate an interrupt. 3'b001 – Host Abort received during transmission 3'b010 – Host Abort received during reception Others: Reserved EB is read-only.
9	R/W	0	ABN_INT_SUM

			<p>Abnormal Interrupt Summary.</p> <p>Logical OR of the following:</p> <p>IDSTS[2] – Fatal Bus Interrupt</p> <p>IDSTS[4] – DU bit Interrupt</p> <p>IDSTS[5] – Card Error Summary Interrupt</p> <p>Only unmasked bits affect this bit.</p> <p>This is a sticky bit and must be cleared each time a corresponding bit that causes AIS to be set is cleared. Writing a 1 clears this bit.</p>
8	R/W	0	<p>NOR_INT_SUM</p> <p>Normal Interrupt Summary.</p> <p>Logical OR of the following:</p> <p>IDSTS[0] – Transmit Interrupt</p> <p>IDSTS[1] – Receive Interrupt</p> <p>Only unmasked bits affect this bit.</p> <p>This is a sticky bit and must be cleared each time a corresponding bit that causes NIS to be set is cleared. Writing a 1 clears this bit.</p>
7:6	/	/	/
5	R/W	0	<p>ERR_FLAG_SUM</p> <p>Card Error Summary.</p> <p>Indicates the status of the transaction to/from the card; also present in RINTSTS. Indicates the logical OR of the following bits:</p> <p>EBE – End Bit Error</p> <p>RTO – Response Timeout/Boot Ack Timeout</p> <p>RCRC – Response CRC</p> <p>SBE – Start Bit Error</p> <p>DRTO – Data Read Timeout/BDS timeout</p> <p>DCRC – Data CRC for Receive</p> <p>RE – Response Error</p> <p>Writing a 1 clears this bit.</p>
4	R/W	0	DES_UNAVL_INT

			Descriptor Unavailable Interrupt. This bit is set when the descriptor is unavailable due to OWN bit = 0 (DES0[31] =0). Writing a 1 clears this bit.
3	/	/	/
2	R/W	0	FATAL_BERR_INT Fatal Bus Error Interrupt. Indicates that a Bus Error occurred (IDSTS[12:10]). When this bit is set, the DMA disables all its bus accesses. Writing a 1 clears this bit.
1	R/W	0	RX_INT Receive Interrupt. Indicates the completion of data reception for a descriptor. Writing a 1 clears this bit.
0	R/W	0	TX_INT Transmit Interrupt. Indicates that data transmission is finished for a descriptor. Writing a '1' clears this bit.

#### 8.1.5.24. SD DMAC Interrupt Enable Register

Offset: 0x008c			Register Name: SD_IDIE Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:10	/	/	/
9	R/W	0	ABN_INT_ENB Abnormal Interrupt Summary Enable. When set, an abnormal interrupt is enabled. This bit enables the following bits: IDINTEN[2] – Fatal Bus Error Interrupt IDINTEN[4] – DU Interrupt IDINTEN[5] – Card Error Summary Interrupt
8	R/W	0	NOR_INT_ENB



			<p>Normal Interrupt Summary Enable.</p> <p>When set, a normal interrupt is enabled. When reset, a normal interrupt is disabled. This bit enables the following bits:</p> <p>IDINTEN[0] – Transmit Interrupt</p> <p>IDINTEN[1] – Receive Interrupt</p>
7:6	/	/	/
5	R/W	0	<p>ERR_SUM_INT_ENB</p> <p>Card Error summary Interrupt Enable.</p> <p>When set, it enables the Card Interrupt summary.</p>
4	R/W	0	<p>DES_UNAVL_INT_ENB</p> <p>Descriptor Unavailable Interrupt.</p> <p>When set along with Abnormal Interrupt Summary Enable, the DU interrupt is enabled.</p>
3	/	/	/
2	R/W	0	<p>FERR_INT_ENB</p> <p>Fatal Bus Error Enable.</p> <p>When set with Abnormal Interrupt Summary Enable, the Fatal Bus Error Interrupt is enabled. When reset, Fatal Bus Error Enable Interrupt is disabled.</p>
1	R/W	0	<p>RX_INT_ENB</p> <p>Receive Interrupt Enable.</p> <p>When set with Normal Interrupt Summary Enable, Receive Interrupt is enabled. When reset, Receive Interrupt is disabled.</p>
0	R/W	0	<p>TX_INT_ENB</p> <p>Transmit Interrupt Enable.</p> <p>When set with Normal Interrupt Summary Enable, Transmit Interrupt is enabled. When reset, Transmit Interrupt is disabled.</p>

#### 8.1.5.25. Card Threshold Control Register

<b>Offset: 0x0100</b>	<b>Register Name: SD_THLD</b>
-----------------------	-------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:28	/	/	/
27:16	R/W	0	CARD_RD_THLD Card Read Threshold Size
15:1	/	/	/
0	R/W	0	CARD_RD_THLD_ENB Card Read Threshold Enable 0 – Card Read Threshold Disable 1 - Card Read Threshold Enable  Host controller initiates Read Transfer only if CARD_RD_THLD amount of space is available in receive FIFO

#### 8.1.5.26. eMMC4.5 DDR Start Bit Detection Control Register

Offset: 0x010c			Register Name: EMMC_DDR_SBIT_DET
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:1	/	/	/
0	R/W	0	HALF_START_BIT Control for start bit detection mechanism inside mstorage based on duration of start bit.  For eMMC 4.5, start bit can be: 0 - Full cycle 1 - Less than one full cycle  Set HALF_START_BIT=1 for eMMC 4.5 and above; set to 0 for SD applications.

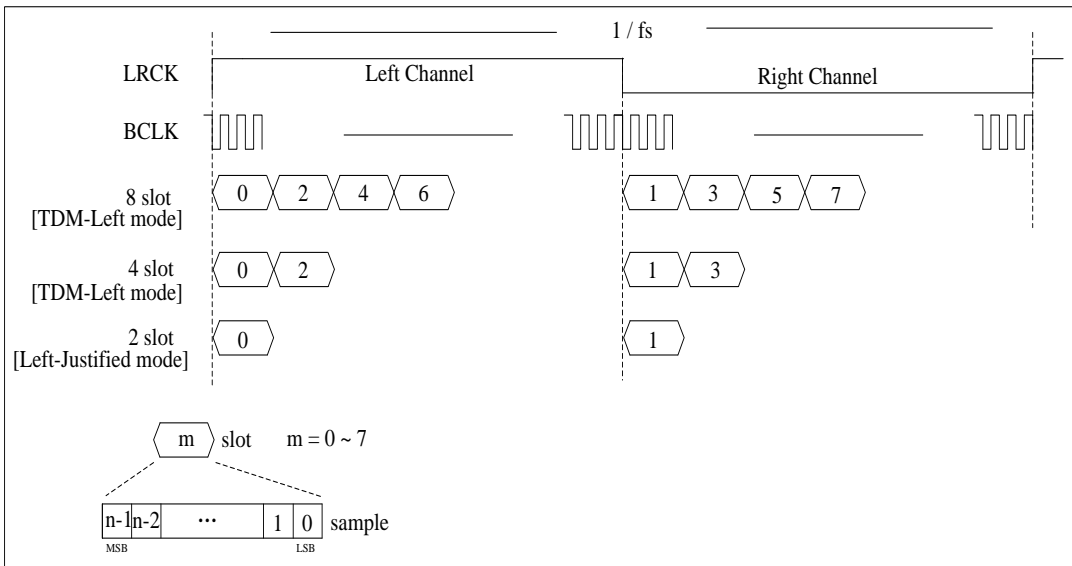
**8.1.5.27. SD FIFO Register**

Offset: 0x0200			Register Name: SD_FIFO Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:0	R/W	0	TX/RX_FIFO Data FIFO

### 8.1.6. SD/MMC Special Requirement

#### 8.1.6.1. SD/MMC Pin List

Port Name	Width	Direction	Description
SD_CCLK	1	OUT	Clock output for SD/SDIO/MMC card
SD_CCMD	1	IN/OUT	CMD line
SD_CDATA	4/8	IN/OUT	Data line



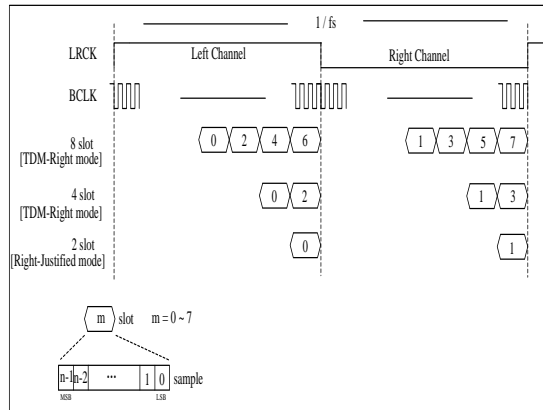
SD/MMC Pin Diagram

### 8.1.7. SD/MMC DMA Controller Description

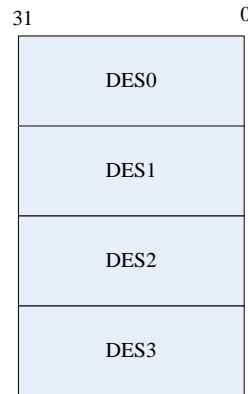
SD3.0 controller has an internal DMA controller (IDMAC) to transfer data between host memory and SDMMC port. With a descriptor, IDMAC can efficiently move data from source to destination by automatically loading next DMA transfer arguments, which need less CPU intervention. Before transfer data in IDMAC, host driver should construct a descriptor list, configure arguments of every DMA transfer, then launch the descriptor and start the DMA. IDMAC has an interrupt controller, when enabled, it can interrupt the HOST CPU in situations such as data transmission completed or some errors happened.

#### 8.1.7.1. IDMAC Descriptor Structure

The IDMAC uses a descriptor with a chain structure, and each descriptor points to a unique buffer and the next descriptor.



This figure illustrates the internal formats of a descriptor. The descriptor addresses must be aligned to the bus width used for 32-bit buses. Each descriptor contains 16 bytes of control and status information.



DES0 is a notation used to denote the [31:0] bits, DES1 to denote [63:32] bits, DES2 to denote [95:64]bits, and DES3 to denote [127:96]bits in a descriptor.

**8.1.7.2. DES0 definition**

Bits	Name	Descriptor
31	HOLD	DES_OWN_FLAG When set, this bit indicates that the descriptor is owned by the IDMAC. When this bit is reset, it indicates that the descriptor is owned by the host. This bit is cleared when transfer is over.
30	ERROR	ERR_FLAG When some error happened in transfer, this bit will be set.
29:6	/	/
5	/	Not used
4	Chain Flag	CHAIM_MOD

		When set, this bit indicates that the second address in descriptor is the next descriptor address. Must be set 1.
3	First DES Flag	FIRST_FLAG When set, this bit indicates that this descriptor contains the first buffer of data. Must be set to 1 in first DES.
2	Last DES Flag	LAST_FLAG When set, this bit indicates that the buffers pointed to by this descriptor are the last data buffer
1	Disable Interrupt on completion	CUR_TXRX_OVER_INT_DIS When set, this bit will prevent the setting of the TX/RX interrupt bit of the IDMAC status register for data that ends in the buffer pointed to by this descriptor
0	/	/

#### 8.1.7.3. DES1 definition

Bits	Name	Descriptor
31:13	/	/
15:0	Buffer size	BUFF_SIZE These bits indicate the data buffer byte size, which must be a multiple of 4 bytes. If this field is 0, the DMA ignores this buffer and proceeds to the next descriptor.

#### 8.1.7.4. DES2 definition

Bits	Name	Descriptor
31:0	Buffer address pointer	BUFF_ADDR these bits indicate the physical address of data buffer. The IDMAC ignores DES2[1:0], corresponding to the bus width of 32.

#### 8.1.7.5. DES3 definition

Bits	Name	Descriptor
31:0	Next descriptor address	NEXT_DESP_ADDR These bits indicate the pointer to the physical memory where the next

	descriptor is present.
--	------------------------

### 8.1.8. SD/MMC-COMM

Module Name	Base Address	
SD/MMC-COMM	0x01C13000	Set SD/MMC internal reset and clock gating

Offset: 0x0000			Register Name: SDC Common Resister_0
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:19	/	/	/
18	R/W	0x0	SDC0 reset switch 0x1: SDC0 reset switch on 0x0: SDC0 reset switch off
17	/	/	/
16	R/W	0x0	SDC0 config clock switch 0x1: SDC0 switch on 0x0: SDC0 switch off

Offset: 0x0004			Register Name: SDC Common Resister_1
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:19	/	/	/
18	R/W	0x0	SDC1 reset switch 0x1: SDC1 reset switch on 0x0: SDC1 reset switch off
17	/	/	/
16	R/W	0x0	SDC1 config clock switch 0x1: SDC1 switch on 0x0: SDC1 switch off

Offset: 0x0008			Register Name: SDC Common Resister_2
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:19	/	/	/
18	R/W	0x0	SDC2 reset switch 0x1: SDC2 reset switch on 0x0: SDC2 reset switch off
17	/	/	/
16	R/W	0x0	SDC2 config clock switch 0x1: SDC2 switch on 0x0: SDC2 switch off

Offset: 0x000c			Register Name: SDC Common Resister_3
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:19	/	/	/
18	R/W	0x0	SDC3 reset switch 0x1: SDC3 reset switch on 0x0: SDC3 reset switch off
17	/	/	/
16	R/W	0x0	SDC3 config clock switch 0x1: SDC3 switch on 0x0: SDC3 switch off



## 8.2. TWI

### 8.2.1. Overview

This TWI Controller is designed to be used as an interface between CPU host and the serial TWI bus. It can supports all the standard TWI transfer, including Slave and Master. The communication to the TWI bus is carried out on a byte-wise basis using interrupt or polled handshaking. This TWI Controller can be operated in standard mode (100K bps) or fast-mode, supporting data rate up to 400K bps. Multiple Masters and 10-bit addressing Mode are supported for this specified application. General Call Addressing is also supported in Slave mode.

The TWI Controller includes the following features:

- Software-programmable for Slave or Master
- Support Repeated START signal
- Multi-master systems supported
- Allow 10-bit addressing with TWI bus
- Performs arbitration and clock synchronization
- Own address and General Call address detection
- Interrupt on address detection
- Support speeds up to 400Kbits/s ('fast mode')
- Allow operation from a wide range of input clock frequencies

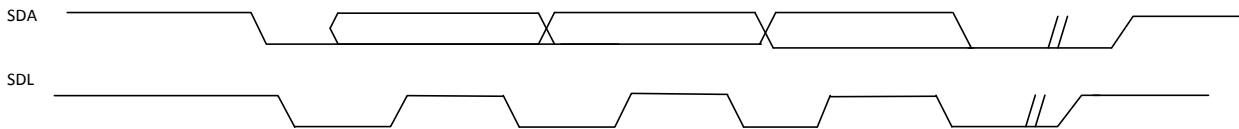
### 8.2.2. Timing Diagram

Data transferred are always in a unit of 8-bit (byte), followed by an acknowledge bit. The number of bytes that can be transmitted per transfer is unrestricted. Data is transferred in serial with the MSB first. Between each byte of data transfer, a receiver device will hold the clock line SCL low to force the transmitter into a wait state while waiting the response from microprocessor.

Data transfer with acknowledge is obligatory. The clock line is driven by the master all the time, including the acknowledge-related clock cycle, except for the SCL holding between each bytes. After sending each byte, the transmitter releases the SDA line to allow the receiver to pull down the SDA line and send an acknowledge signal (or leave it high to send a "not acknowledge") to the transmitter.

When a slave receiver doesn't acknowledge the slave address (unable to receive because of no resource available), the data line must be left high by the slave so that the master can then generate a STOP condition to abort the transfer. Slave receiver can also indicate not to want to send more data during a transfer by leave the acknowledge signal high. And the master should generate the STOP condition to abort the transfer.

Below diagram provides an illustration the relation of SDA signal line and SCL signal line on the TWI serial bus.



TWI Timing Diagram

### 8.2.3. TWI Controller Register List

Module Name	Base Address
R_TWI0	0x08002400
R_TWI1	0x08003800
TWI0	0x07002800
TWI1	0x07002C00
TWI2	0x07003000
TWI3	0x07003400
TWI4	0x07003800

Register Name	Offset	Description
TWI_ADDR	0x0000	TWI Slave address
TWI_XADDR	0x0004	TWI Extended slave address
TWI_DATA	0x0008	TWI Data byte
TWI_CNTR	0x000C	TWI Control register
TWI_STAT	0x0010	TWI Status register
TWI_CCR	0x0014	TWI Clock control register
TWI_SRST	0x0018	TWI Software reset
TWI_EFR	0x001C	TWI Enhance Feature register
TWI_LCR	0x0020	TWI Line Control register

## 8.2.4. TWI Controller Register Description

### 8.2.4.1. TWI Slave Address Register

Offset: 0x00			Register Name: TWI_ADDR Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:8	/	/	/
7:1	R/W	0	SLA Slave address <ul style="list-style-type: none"> <li>• 7-bit addressing SLA6, SLA5, SLA4, SLA3, SLA2, SLA1, SLA0</li> <li>• 10-bit addressing 1, 1, 1, 1, 0, SLAX[9:8]</li> </ul>
0	R/W	0	GCE General call address enable 0: Disable 1: Enable

Notes:

For 7-bit addressing:

SLA6 – SLA0 is the 7-bit address of the TWI when in slave mode. When the TWI receives this address after a START condition, it will generate an interrupt and enter slave mode. (SLA6 corresponds to the first bit received from the TWI bus.) If GCE is set to '1', the TWI will also recognize the general call address (00h).

For 10-bit addressing:

When the address received starts with 11110b, the TWI recognizes this as the first part of a 10-bit address and if the next two bits match ADDR[2:1] (i.e. SLAX9 and SLAX8 of the device's extended address), it sends an ACK. (The device does not generate an interrupt at this point.) If the next byte of the address matches the XADDR register (SLAX7 – SLAX0), the TWI generates an interrupt and goes into slave mode.

#### 8.2.4.2. TWI Extend Address Register

<b>Offset: 0x04</b>			<b>Register Name: TWI_XADDR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R/W	0	SLAX Extend Slave Address SLAX[7:0]

#### 8.2.4.3. TWI Data Register

<b>Offset: 0x08</b>			<b>Register Name: TWI_DATA</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R/W	0	TWI_DATA Data byte for transmitting or received

#### 8.2.4.4. TWI Control Register

<b>Offset: 0x0C</b>			<b>Register Name: TWI_CNTR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7	R/W	0	INT_EN Interrupt Enable 1'b0: The interrupt line always low 1'b1: The interrupt line will go high when INT_FLAG is set.

6	R/W	0	<p><b>BUS_EN</b></p> <p>TWI Bus Enable</p> <p>1'b0: The TWI bus inputs ISDA/ISCL are ignored and the TWI Controller will not respond to any address on the bus</p> <p>1'b1: The TWI will respond to calls to its slave address – and to the general call address if the GCE bit in the ADDR register is set.</p> <p>Notes: In master operation mode, this bit should be set to '1'</p>
5	R/W	0	<p><b>M_STA</b></p> <p>Master Mode Start</p> <p>When M_STA is set to '1', TWI Controller enters master mode and will transmit a START condition on the bus when the bus is free. If the M_STA bit is set to '1' when the TWI Controller is already in master mode and one or more bytes have been transmitted, then a repeated START condition will be sent. If the M_STA bit is set to '1' when the TWI is being accessed in slave mode, the TWI will complete the data transfer in slave mode then enter master mode when the bus has been released.</p> <p>The M_STA bit is cleared automatically after a START condition has been sent: writing a '0' to this bit has no effect.</p>
4	R/W	0	<p><b>M_STP</b></p> <p>Master Mode Stop</p> <p>If M_STP is set to '1' in master mode, a STOP condition is transmitted on the TWI bus. If the M_STP bit is set to '1' in slave mode, the TWI will behave as if a STOP condition has been received, but no STOP condition will be transmitted on the TWI bus. If both M_STA and M_STP bits are set, the TWI will first transmit the STOP condition (if in master mode) then transmit the START condition.</p> <p>The M_STP bit is cleared automatically: writing a '0' to this bit has no effect.</p>
3	R/W	0	<p><b>INT_FLAG</b></p> <p>Interrupt Flag</p> <p>INT_FLAG is automatically set to '1' when any of 28 (out of the possible 29) states is entered (see 'STAT Register' below). The only state that does not set INT_FLAG is state F8h. If the INT_EN bit is set, the interrupt</p>

			line goes high when IFLG is set to '1'. If the TWI is operating in slave mode, data transfer is suspended when INT_FLAG is set and the low period of the TWI bus clock line (SCL) is stretched until '1' is written to INT_FLAG. The TWI clock line is then released and the interrupt line goes low.
			<p>A_ACK</p> <p>Assert Acknowledge</p> <p>When A_ACK is set to '1', an Acknowledge (low level on SDA) will be sent during the acknowledge clock pulse on the TWI bus if:</p> <ol style="list-style-type: none"> <li>1. Either the whole of a matching 7-bit slave address or the first or the second byte of a matching 10-bit slave address has been received.</li> <li>2. The general call address has been received and the GCE bit in the ADDR register is set to '1'.</li> <li>3. A data byte has been received in master or slave mode.</li> </ol> <p>When A_ACK is '0', a Not Acknowledge (high level on SDA) will be sent when a data byte is received in master or slave mode.</p> <p>If A_ACK is cleared to '0' in slave transmitter mode, the byte in the DATA register is assumed to be the 'last byte'. After this byte has been transmitted, the TWI will enter state C8h then return to the idle state (status code F8h) when INT_FLAG is cleared.</p>
2	R/W	0	The TWI will not respond as a slave unless A_ACK is set.
1:0	R/W	0	/

#### 8.2.4.5. TWI Status Register

<b>Offset: 0x10</b>			<b>Register Name: TWI_STAT</b>
			<b>Default Value: 0x0000_00F8</b>
Bit	Read/Write	Default	Description
31:8	/	/	/

7:0	R	<p>STA</p> <p>Status Information Byte</p> <p><b>Code Status</b></p> <p>0x00: Bus error</p> <p>0x08: START condition transmitted</p> <p>0x10: Repeated START condition transmitted</p> <p>0x18: Address + Write bit transmitted, ACK received</p> <p>0x20: Address + Write bit transmitted, ACK not received</p> <p>0x28: Data byte transmitted in master mode, ACK received</p> <p>0x30: Data byte transmitted in master mode, ACK not received</p> <p>0x38: Arbitration lost in address or data byte</p> <p>0x40: Address + Read bit transmitted, ACK received</p> <p>0x48: Address + Read bit transmitted, ACK not received</p> <p>0x50: Data byte received in master mode, ACK transmitted</p> <p>0x58: Data byte received in master mode, not ACK transmitted</p> <p>0x60: Slave address + Write bit received, ACK transmitted</p> <p>0x68: Arbitration lost in address as master, slave address + Write bit received, ACK transmitted</p> <p>0x70: General Call address received, ACK transmitted</p> <p>0x78: Arbitration lost in address as master, General Call address received, ACK transmitted</p> <p>0x80: Data byte received after slave address received, ACK transmitted</p> <p>0x88: Data byte received after slave address received, not ACK transmitted</p> <p>0x90: Data byte received after General Call received, ACK transmitted</p> <p>0x98: Data byte received after General Call received, not ACK transmitted</p> <p>0xA0: STOP or repeated START condition received in slave mode</p> <p>0xA8: Slave address + Read bit received, ACK transmitted</p> <p>0xB0: Arbitration lost in address as master, slave address + Read bit received, ACK transmitted</p>
-----	---	---



		0xB8: Data byte transmitted in slave mode, ACK received 0xC0: Data byte transmitted in slave mode, ACK not received 0xC8: Last byte transmitted in slave mode, ACK received 0xD0: Second Address byte + Write bit transmitted, ACK received 0xD8: Second Address byte + Write bit transmitted, ACK not received 0xF8: No relevant status information, INT_FLAG=0 Others: Reserved
--	--	---

#### 8.2.4.6. TWI Clock Register

Offset: 0x14			Register Name: TWI_CCR
			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:7	/	/	/
6:3	R/W	0	CLK_M
			CLK_N The TWI bus is sampled by the TWI at the frequency defined by F0: $F_{\text{samp}} = F_0 = F_{\text{in}} / 2^{\text{CLK\_N}}$  The TWI OSCL output frequency, in master mode, is F1 / 10: $F_1 = F_0 / (\text{CLK\_M} + 1)$ $F_{\text{oscl}} = F_1 / 10 = F_{\text{in}} / (2^{\text{CLK\_N}} * (\text{CLK\_M} + 1) * 10)$ For Example: Fin = 48Mhz (APB clock input) For 400kHz full speed 2Wire, CLK_N = 2, CLK_M=2 $F_0 = 48\text{M} / 2^2 = 12\text{Mhz}$ , $F_1 = F_0 / (10 * (2+1)) = 0.4\text{Mhz}$  For 100Khz standard speed 2Wire, CLK_N=2, CLK_M=11 $F_0 = 48\text{M} / 2^2 = 12\text{Mhz}$ , $F_1 = F_0 / (10 * (11+1)) = 0.1\text{Mhz}$
2:0	R/W	0	

**8.2.4.7. TWI Soft Reset Register**

<b>Offset: 0x18</b>			<b>Register Name: TWI_SRST</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:1	/	/	/
0	R/W	0	SOFT_RST Soft Reset Write '1' to this bit to reset the TWI and clear to '0' when completing Soft Reset operation.

**8.2.4.8. TWI Enhance Feature Register**

<b>Offset: 0x1C</b>			<b>Register Name: TWI_EFR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:2	/	/	/
0:1	R/W	0	DBN Data Byte number follow Read Command Control 0— No Data Byte to be written after read command 1— Only 1 byte data to be written after read command 2— 2 bytes data can be written after read command 3— 3 bytes data can be written after read command

**8.2.4.9. TWI Line Control Register**

<b>Offset: 0x20</b>			<b>Register Name: TWI_LCR</b> <b>Default Value: 0x0000_003a</b>
Bit	Read/Write	Default	Description
31:6	/	/	/

5	R	1	<p>SCL_STATE</p> <p>Current state of TWI_SCL</p> <p>0 – low</p> <p>1 - high</p>
4	R	1	<p>SDA_STATE</p> <p>Current state of TWI_SDA</p> <p>0 – low</p> <p>1 - high</p>
3	R/W	1	<p>SCL_CTL</p> <p>TWI_SCL line state control bit</p> <p>When line control mode is enabled (bit[2] set), value of this bit decide the output level of TWI_SCL</p> <p>0 – output low level</p> <p>1 – output high level</p>
2	R/W	0	<p>SCL_CTL_EN</p> <p>TWI_SCL line state control enable</p> <p>When this bit is set, the state of TWI_SCL is control by the value of bit[3].</p> <p>0-disable TWI_SCL line control mode</p> <p>1-enable TWI_SCL line control mode</p>
1	R/W	1	<p>SDA_CTL</p> <p>TWI_SDA line state control bit</p> <p>When line control mode is enabled (bit[0] set), value of this bit decide the output level of TWI_SDA</p> <p>0 – output low level</p> <p>1 – output high level</p>
0	R/W	0	<p>SDA_CTL_EN</p> <p>TWI_SDA line state control enable</p> <p>When this bit is set, the state of TWI_SDA is control by the value of bit[1].</p>

			0-disable TWI_SDA line control mode 1-enable TWI_SDA line control mode
--	--	--	---

#### 8.2.4.10. TWI DVFS Control Register

Offset: 0x24			Register Name: TWI_DVFSCR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:2	/	/	/
2	R/W	0	MS_PRIORITY CPU and DVFS BUSY set priority select 0: CPU has higher priority 1: DVFS has higher priority
1	R/W	0	CPU_BUSY_SET CPU Busy set
0	R/W	0	DVFC_BUSY_SET DVFS Busy set

Notes: This register is only implemented in TWI0.

## 8.2.5. TWI Controller Special Requirement

### 8.2.5.1. TWI Pin List

Port Name	Width	Direction	Description
TWI_SCL	1	IN/OUT	TWI Clock line
TWI_SDA	1	IN/OUT	TWI Serial Data line

### 8.2.5.2. TWI Controller Operation

There are four operation modes on the TWI bus which dictates the communications method. They are Master Transmit, Master Receive, Slave Transmit and Slave Receive. In general, CPU host controls TWI by writing commands and data to its registers. The TWI interrupts the CPU host for the attention each time a byte transfer is done or a START/STOP conditions is detected. The CPU host can also poll the status register for current status if the interrupt mechanism is not disabled by the CPU host.

When the CPU host wants to start a bus transfer, it initiates a bus START to enter the master mode by setting IM\_STA bit in the 2WIRE\_CNTR register to high (before it must be low). The TWI will assert INT line and INT\_FLAG to indicate a completion for the START condition and each consequent byte transfer. At each interrupt, the micro-processor needs to check the 2WIRE\_STAT register for current status. A transfer has to be concluded with STOP condition by setting M\_STP bit high.

In Slave Mode, the TWI also constantly samples the bus and look for its own slave address during addressing cycles. Once a match is found, it is addressed and interrupt the CPU host with the corresponding status. Upon request, the CPU host should read the status, read/write 2WIRE\_DATA data register, and set the 2WIRE\_CNTR control register. After each byte transfer, a slave device always halt the operation of remote master by holding the next low pulse on SCL line until the microprocessor responds to the status of previous byte transfer or START condition.

## 8.3. SPI

### 8.3.1. Overview

The SPI is the Serial Peripheral Interface which allows rapid data communication with fewer software interrupts. The SPI module contains one 128x8 receiver buffer (RXFIFO) and one 128x8 transmit buffer (TXFIFO). It can work at two modes: Master mode and Slave mode.

It includes the following features:

- Full-duplex synchronous serial interface
- Master/Slave configurable
- Four chip selects to support multiple peripherals for SPI0, SPI1 and SPI2 has one chip select
- 8-bit wide by 128-entry FIFO for both transmit and receive data
- TXFIFO/RXFIFO access size by byte or word
- Support single and dual mode
- Polarity and phase of the Chip Select (SPI\_SS) and SPI Clock (SPI\_SCLK) are configurable
- Support DMA wait and handshake mode

### 8.3.2. SPI Timing Diagram

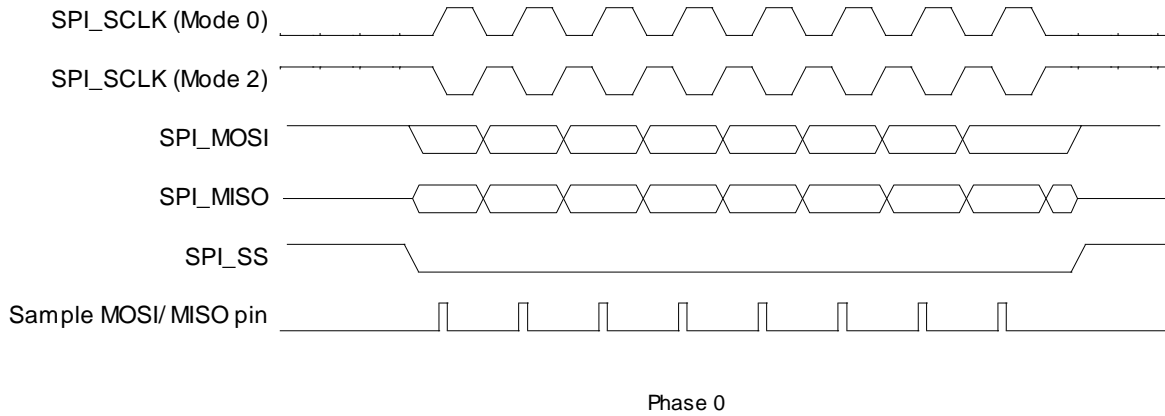
The serial peripheral interface master uses the SPI\_SCLK signal to transfer data in and out of the shift register. Data is clocked using any one of four programmable clock phase and polarity combinations.

During Phase 0, Polarity 0 and Phase 1, Polarity 1 operations, output data changes on the falling clock edge and input data is shifted in on the rising edge.

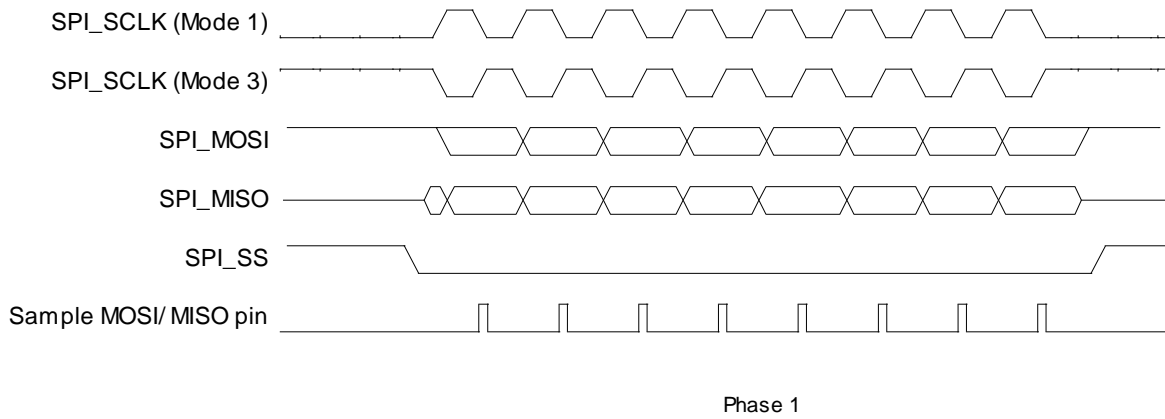
During Phase 1, Polarity 0 and Phase 0, Polarity 1 operations, output data changes on the rising edges of the clock and is shifted in on falling edges.

The POL defines the signal polarity when SPI\_SCLK is in idle state. The SPI\_SCLK is high level when POL is '1' and it is low level when POL is '0'. The PHA decides whether the leading edge of SPI\_SCLK is used for setup or sample data. The leading edge is used for setup data when PHA is '1' and for sample data when PHA is '0'. The four kind of modes are listed below:

SPI Mode	POL	PHA	Leading Edge	Trailing Edge
0	0	0	Rising, Sample	Falling, Setup
1	0	1	Rising, Setup	Falling, Sample
2	1	0	Falling, Sample	Rising, Setup
3	1	1	Falling, Setup	Rising, Sample



**SPI Phase 0 Timing Diagram**



**SPI Phase 1 Timing Diagram**

### 8.3.3. SPI Register List

Module Name	Base Address
SPI0	0x01C1A000
SPI1	0x01C1B000
SPI2	0x01C1C000
SPI3	0x01C1D000

Register Name	Offset	Description
SPI_VER	0x00	SPI Version Number Register
SPI_GCR	0x04	SPI Global Control Register
SPI_TCR	0x08	SPI Transfer Control register
/	0x0c	reserved
SPI_IER	0x10	SPI Interrupt Control register
SPI_ISR	0x14	SPI Interrupt Status register
SPI_FCR	0x18	SPI FIFO Control register
SPI_FSR	0x1C	SPI FIFO Status register
SPI_WCR	0x20	SPI Wait Clock Counter register
SPI_CCR	0x24	SPI Clock Rate Control register
/	0x28	reserved
/	0x2c	reserved
SPI_MBC	0x30	SPI Burst Counter register
SPI_MTC	0x34	SPI Transmit Counter Register
SPI_BCC	0x38	SPI Burst Control register
SPI_DMA	0x88	SPI DMA Control Register
SPI_TXD	0x200	SPI TX Data register
SPI_RXD	0x300	SPI RX Data register



### 8.3.4. SPI Register Description

#### 8.3.4.1. SPI Version Number Register

<b>Offset: 0x00</b>			<b>Register Name: SPI_VER</b>
			<b>Default Value: 0x0000_0080</b>
Bit	Read/Write	Default	Description
31:16	R	0	VER_H
15:0	R	0	VER_L

#### 8.3.4.2. SPI Global Control Register

<b>Offset: 0x04</b>			<b>Register Name: SPI_CTL</b>
			<b>Default Value: 0x0000_0080</b>
Bit	Read/Write	Default	Description
31	R/W	0	SRST  Soft reset  Write '1' to this bit will clear the SPI controller, and auto clear to '0' when reset operation completes  Write '0' has no effect.
30:8	/	/	/
7	R/W	1	TP_EN  Transmit Pause Enable  In master mode, it is used to control transmit state machine to stop smart burst sending when RX FIFO is full.  1 – stop transmit data when RXFIFO full  0 – normal operation, ignore RXFIFO status  Note: Can't be written when XCH=1
6:2	/	/	/
1	R/W	0	MODE

			SPI Function Mode Select  0: Slave Mode  1: Master Mode  Note: Can't be written when XCH=1
0	R/W	0	EN  SPI Module Enable Control  0: Disable  1: Enable

### 8.3.4.3. SPI Transfer Control Register

Offset: 0x08			Register Name: SPI_INTCTL
Default Value: 0x0000_0087			
Bit	Read/Write	Default	Description
31	R/W	0x0	XCH  Exchange Burst  In master mode it is used to start SPI burst  0: Idle  1: Initiates exchange.  Write "1" to this bit will start the SPI burst, and will auto clear after finishing the bursts transfer specified by BC. Write "1" to SRST will also clear this bit. Write '0' to this bit has no effect.  Note: Can't be written when XCH=1.
30:14	/	/	/
13	R/W	0x0	SDM  Master Sample Data Mode  0 - Delay Sample Mode

			<p>1 - Normal Sample Mode</p> <p>In Normal Sample Mode, SPI master samples the data at the correct edge for each SPI mode;</p> <p>In Delay Sample Mode, SPI master samples data at the edge that is half cycle delayed by the correct edge defined in respective SPI mode.</p>
12	R/W	0x0	<p>FBS</p> <p>First Transmit Bit Select</p> <p>0: MSB first</p> <p>1: LSB first</p> <p>Note: Can't be written when XCH=1.</p>
11	R/W	0x0	<p>SDC</p> <p>Master Sample Data Control</p> <p>Set this bit to '1' to make the internal read sample point with a delay of half cycle of SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK propagating between master and slave.</p> <p>0 – normal operation, do not delay internal read sample point</p> <p>1 – delay internal read sample point</p> <p>Note: Can't be written when XCH=1.</p>
10	R/W	0x0	<p>RPSM</p> <p>Rapids mode select</p> <p>Select Rapids mode for high speed write.</p> <p>0: normal write mode</p> <p>1: rapids write mode</p> <p>Note: Can't be written when XCH=1.</p>
9	R/W	0x0	<p>DDB</p> <p>Dummy Burst Type</p>

			<p>0: The bit value of dummy SPI burst is zero</p> <p>1: The bit value of dummy SPI burst is one</p> <p>Note: Can't be written when XCH=1.</p>
8	R/W	0x0	<p>DHB</p> <p>Discard Hash Burst</p> <p>In master mode it controls whether discarding unused SPI bursts</p> <p>0: Receiving all SPI bursts in BC period</p> <p>1: Discard unused SPI bursts, only fetching the SPI bursts during dummy burst period. The bursts number is specified by TC.</p> <p>Note: Can't be written when XCH=1.</p>
7	R/W	0x1	<p>SS_LEVEL</p> <p>When control SS signal manually (SPI_CTRL_REG.SS_CTRL==1), set this bit to '1' or '0' to control the level of SS signal.</p> <p>0: set SS to low</p> <p>1: set SS to high</p> <p>Note: Can't be written when XCH=1.</p>
6	R/W	0x0	<p>SS_OWNER</p> <p>SS Output Owner Select</p> <p>Usually, controller sends SS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of SS signal.</p> <p>0: SPI controller</p> <p>1: Software</p> <p>Note: Can't be written when XCH=1.</p>
5:4	R/W	0x0	<p>SS_SEL</p> <p>SPI Chip Select</p> <p>Select one of four external SPI Master/Slave Devices</p>

			<p>00: SPI_SS0 will be asserted</p> <p>01: SPI_SS1 will be asserted</p> <p>10: SPI_SS2 will be asserted</p> <p>11: SPI_SS3 will be asserted</p> <p>Note: Can't be written when XCH=1.</p>
3	R/W	0x0	<p>SSCTL</p> <p>In master mode, this bit selects the output wave form for the SPI_SSx signal. Only valid when SS_OWNER = 0.</p> <p>0: SPI_SSx remains asserted between SPI bursts</p> <p>1: Negate SPI_SSx between SPI bursts</p> <p>Note: Can't be written when XCH=1.</p>
2	R/W	0x1	<p>SPOL</p> <p>SPI Chip Select Signal Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>Note: Can't be written when XCH=1.</p>
1	R/W	0x1	<p>CPOL</p> <p>SPI Clock Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>Note: Can't be written when XCH=1.</p>
0	R/W	0x1	<p>CPHA</p> <p>SPI Clock/Data Phase Control</p> <p>0: Phase 0 (Leading edge for sample data)</p> <p>1: Phase 1 (Leading edge for setup data)</p> <p>Note: Can't be written when XCH=1.</p>

#### 8.3.4.4. SPI Interrupt Control Register

Offset: 0x10			Register Name: SPI_IER
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:14	R	0x0	Reserved.
13	R/W	0x0	SS_INT_EN SSI Interrupt Enable Chip Select Signal (SSx) from valid state to invalid state 0: Disable 1: Enable
12	R/W	0x0	TC_INT_EN Transfer Completed Interrupt Enable 0: Disable 1: Enable
11	R/W	0x0	TF_UDR_INT_EN TXFIFO under run Interrupt Enable 0: Disable 1: Enable
10	R/W	0x0	TF_OVF_INT_EN TX FIFO Overflow Interrupt Enable 0: Disable 1: Enable
9	R/W	0x0	RF_UDR_INT_EN RXFIFO under run Interrupt Enable 0: Disable

			1: Enable
8	R/W	0x0	RF_OVF_INT_EN RX FIFO Overflow Interrupt Enable 0: Disable 1: Enable
7	R	0x0	Reserved.
6	R/W	0x0	TF_FUL_INT_EN TX FIFO Full Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TX_EMP_INT_EN TX FIFO Empty Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TX_ERQ_INT_EN TX FIFO Empty Request Interrupt Enable 0: Disable 1: Enable
3	R	0x0	Reserved
2	R/W	0x0	RF_FUL_INT_EN RX FIFO Full Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RX_EMP_INT_EN RX FIFO Empty Interrupt Enable 0: Disable 1: Enable

0	R/W	0x0	<p>RF_RDY_INT_EN</p> <p>RX FIFO Ready Request Interrupt Enable</p> <p>0: Disable</p> <p>1: Enable</p>
---	-----	-----	---

#### 8.3.4.5. SPI Interrupt Status Register

Offset: 0x14			Register Name: SPI_INT_STA
Default Value: 0x0000_0022			
Bit	Read/Write	Default	Description
31:14	/	0	/
13	R/W	0	<p>SSI</p> <p>SS Invalid Interrupt</p> <p>When SSI is 1, it indicates that SS has changed from valid state to invalid state. Writing 1 to this bit clears it.</p>
12	R/W	0	<p>TC</p> <p>Transfer Completed</p> <p>In master mode, it indicates that all bursts specified by BC has been exchanged. In other condition, When set, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it.</p> <p>0: Busy</p> <p>1: Transfer Completed</p>
11	R/W	0	<p>TF_UDF</p> <p>TXFIFO Underrun</p> <p>This bit is set when if the TXFIFO is underrun. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not underrun</p> <p>1: TXFIFO is underrun</p>



10	R/W	0	<p>TF_OVF</p> <p>TXFIFO Overflow</p> <p>This bit is set when if the TXFIFO is overflow. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not overflow</p> <p>1: TXFIFO is overflowed</p>
9	R/W	0	<p>RX_UDF</p> <p>RXFIFO Underrun</p> <p>When set, this bit indicates that RXFIFO has underrun. Writing 1 to this bit clears it.</p>
8	R/W	0	<p>RX_OVF</p> <p>RXFIFO Overflow</p> <p>When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it.</p> <p>0: RXFIFO is available.</p> <p>1: RXFIFO has overflowed.</p>
7	/	/	/
6	R/W	0	<p>TX_FULL</p> <p>TXFIFO Full</p> <p>This bit is set when if the TXFIFO is full . Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not Full</p> <p>1: TXFIFO is Full</p>
5	R/W	1	<p>TX_EMP</p> <p>TXFIFO Empty</p> <p>This bit is set if the TXFIFO is empty. Writing 1 to this bit clears it.</p> <p>0: TXFIFO contains one or more words.</p> <p>1: TXFIFO is empty</p>

4	R/W	0	<p><b>TX_READY</b></p> <p>TXFIFO Ready</p> <p>0: TX_WL &gt; TX_TRIG_LEVEL</p> <p>1: TX_WL &lt;= TX_TRIG_LEVEL</p> <p>This bit is set any time if TX_WL &lt;= TX_TRIG_LEVEL. Writing "1" to this bit clears it. Where TX_WL is the water level of RXFIFO</p>
3	/	/	reserved
2	R/W	0	<p><b>RX_FULL</b></p> <p>RXFIFO Full</p> <p>This bit is set when the RXFIFO is full . Writing 1 to this bit clears it.</p> <p>0: Not Full</p> <p>1: Full</p>
1	R/W	1	<p><b>RX_EMP</b></p> <p>RXFIFO Empty</p> <p>This bit is set when the RXFIFO is empty . Writing 1 to this bit clears it.</p> <p>0: Not empty</p> <p>1: empty</p>
0	R/W	0	<p><b>RX_RDY</b></p> <p>RXFIFO Ready</p> <p>0: RX_WL &lt; RX_TRIG_LEVEL</p> <p>1: RX_WL &gt;= RX_TRIG_LEVEL</p> <p>This bit is set any time if RX_WL &gt;= RX_TRIG_LEVEL. Writing "1" to this bit clears it. Where RX_WL is the water level of RXFIFO.</p>

#### 8.3.4.6. SPI FIFO Control Register

<b>Offset: 0x18</b>	<b>Register Name: SPI_DMACTL</b>
---------------------	----------------------------------

			Default Value: 0x0040_0001
Bit	Read/Write	Default	Description
31	R/W	0	<p>TX_FIFO_RST</p> <p>TX FIFO Reset</p> <p>Write '1' to this bit will reset the control portion of the TX FIFO and auto clear to '0' when completing reset operation, write to '0' has no effect.</p>
30	R/W	0	<p>TF_TEST_ENB</p> <p>TX Test Mode Enable</p> <p>0: disable</p> <p>1: enable</p> <p>Note: In normal mode, TX FIFO can only be read by SPI controller, write '1' to this bit will switch TX FIFO read and write function to AHB bus. This bit is used to test the TX FIFO, don't set in normal operation and don't set RF_TEST and TF_TEST at the same time.</p>
29:28	/	/	/
27:26	R/W	0x0	<p>TX_FIFO_ACCESS_SIZE</p> <p>00: TX FIFO access in byte</p> <p>01: TX FIFO access in word (4bytes)</p> <p>10: Reserved</p> <p>11: TX FIFO access size controlled by bus</p>
25	/	/	/
24	R/W	0x0	<p>TF_DRQ_EN</p> <p>TX FIFO DMA Request Enable</p> <p>0: Disable</p> <p>1: Enable</p>
23:16	R/W	0x40	<p>TX_TRIG_LEVEL</p> <p>TX FIFO Empty Request Trigger Level</p>
15	R/W	0x0	<p>RF_RST</p>

			<p>RXFIFO Reset</p> <p>Write '1' to this bit will reset the control portion of the receiver FIFO, and auto clear to '0' when completing reset operation, write '0' to this bit has no effect.</p>
14	R/W	0x0	<p>RF_TEST</p> <p>RX Test Mode Enable</p> <p>0: Disable</p> <p>1: Enable</p> <p>Note: In normal mode, RX FIFO can only be written by SPI controller, write '1' to this bit will switch RX FIFO read and write function to AHB bus. This bit is used to test the RX FIFO, don't set in normal operation and don't set RF_TEST and TF_TEST at the same time.</p>
13:12	R	0x0	Reserved
11:10	R/W	0x0	<p>RX_FIFO_ACCESS_SIZE</p> <p>00: RX FIFO access in byte</p> <p>01: RX FIFO access in word (4bytes)</p> <p>10: Reserved</p> <p>11: RX FIFO access size controlled by bus</p>
9	R/W	0x0	<p>RX_DMA_MODE</p> <p>SPI RX DMA Mode Control</p> <p>0: Normal DMA mode</p> <p>1: Dedicate DMA mode</p>
8	R/W	0x0	<p>RF_DRQ_EN</p> <p>RX FIFO DMA Request Enable</p> <p>0: Disable</p> <p>1: Enable</p>
7:0	R/W	0x1	<p>RX_TRIG_LEVEL</p> <p>RX FIFO Ready Request Trigger Level</p>

**8.3.4.7. SPI FIFO Status Register**

Offset: 0x1c			Register Name: SPI_FSR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31	R	0x0	TB_WR TX FIFO Write Buffer Write Enable
30:28	R	0x0	TB_CNT TX FIFO Write Buffer Counter These bits indicate the number of words in TX FIFO Write Buffer
27:24	R	0x0	Reserved
23:16	R	0x0	TF_CNT TX FIFO Counter These bits indicate the number of words in TX FIFO 0: 0 byte in TX FIFO 1: 1 byte in TX FIFO ... 128: 128 bytes in TX FIFO
15	R	0x0	RB_WR RX FIFO Read Buffer Write Enable
14:12	R	0x0	RB_CNT RX FIFO Read Buffer Counter These bits indicate the number of words in RX FIFO Read Buffer
11:8	R	0x0	Reserved
7:0	R	0x0	RF_CNT RX FIFO Counter These bits indicate the number of words in RX FIFO 0: 0 byte in RX FIFO

			1: 1 byte in RX FIFO
			...
			128: 128 bytes in RX FIFO

#### 8.3.4.8. SPI Wait Clock Register

Offset: 0x20			Register Name: SPI_WAIT
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:20	/	/	/
19:16	R/W	0x0	SWC Dual mode direction switch wait clock counter (for master mode only). 0: No wait states inserted n: n SPI_SCLK wait states inserted Note: These bits control the number of wait states to be inserted before start dual data transfer in dual SPI mode. The SPI module counts SPI_SCLK by SWC for delaying next word data transfer. Note: Can't be written when XCH=1.
15:0	R/W	0	WCC Wait Clock Counter (In Master mode) These bits control the number of wait states to be inserted in data transfers. The SPI module counts SPI_SCLK by WCC for delaying next word data transfer. 0: No wait states inserted N: N SPI_SCLK wait states inserted

#### 8.3.4.9. SPI Clock Control Register

Offset: 0x24	Register Name: SPI_CCTL
--------------	-------------------------

			Default Value: 0x0000_0002
Bit	Read/Write	Default	Description
31:13	/	/	/
12	R/W	0	DRS Divide Rate Select (Master Mode Only) 0: Select Clock Divide Rate 1 1: Select Clock Divide Rate 2
11:8	R/W	0	CDR1 Clock Divide Rate 1 (Master Mode Only) The SPI_SCLK is determined according to the following equation: $SPI\_CLK = Source\_CLK / 2^n$ .
7:0	R/W	0x2	CDR2 Clock Divide Rate 2 (Master Mode Only) The SPI_SCLK is determined according to the following equation: $SPI\_CLK = Source\_CLK / (2 * (n + 1))$ .

#### 8.3.4.10. SPI Master Burst Counter Register

			Register Name: SPI_BC
Offset: 0x30			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:24	/	/	/
23:0	R/W	0	MBC Master Burst Counter In master mode, this field specifies the total burst number. 0: 0 burst 1: 1 burst ... N: N bursts

### 8.3.4.11. SPI Master Transmit Counter Register

Offset: 0x34			Register Name: SPI_TC
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:24	/	/	/
23:0	R/W	0	<p>MWTC</p> <p>Master Write Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent to TXFIFO before automatically sending dummy burst. For saving bus bandwidth, the dummy burst (all zero bits or all one bits) is sent by SPI Controller automatically.</p> <p>0: 0 burst</p> <p>1: 1 burst</p> <p>...</p> <p>N: N bursts</p>

### 8.3.4.12. SPI Master Burst Control Counter Register

Offset: 0x38			Register Name: SPI_BCC
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:29	R	0x0	Reserved
28	R/W	0x0	<p>DRM</p> <p>Master Dual Mode RX Enable</p> <p>0: RX use single-bit mode</p> <p>1: RX use dual mode</p> <p>Note: Can't be written when XCH=1.</p>
27:24	R/W	0x0	DBC



			<p>Master Dummy Burst Counter</p> <p>In master mode, this field specifies the burst number that should be sent before receive in dual SPI mode. The data is don't care by the device.</p> <p>0: 0 burst</p> <p>1: 1 burst</p> <p>...</p> <p>N: N bursts</p> <p>Note: Can't be written when XCH=1.</p>
23:0	R/W	0x0	<p>STC</p> <p>Master Single Mode Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent in single mode before automatically sending dummy burst. This is the first transmit counter in all bursts.</p> <p>0: 0 burst</p> <p>1: 1 burst</p> <p>...</p> <p>N: N bursts</p> <p>Note: Can't be written when XCH=1.</p>

### 8.3.4.13. SPI DMA Control Register

<b>Offset: 0x88</b>			<b>Register Name: SPI_DMA_CTL</b>  <b>Default Value: 0x0000_00A5</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R/W	0xA5	DMA Handshake configuration

			0xA5: DMA wait cycle mode
			0xEA: DMA handshake mode

#### 8.3.4.14. SPI TX Data Register

Offset: 0x200			Register Name: SPI_TXD
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:0	W/R	0x0	TDATA Transmit Data This register can be accessed in byte, half-word or word unit by AHB. In byte accessing method, if there are rooms in RXFIFO, one burst data is written to RXFIFO and the depth is increased by 1. In half-word accessing method, two SPI burst data are written and the TXFIFO depth is increase by 2. In word accessing method, four SPI burst data are written and the TXFIFO depth is increased by 4. Note: This address is writing-only if TF_TEST is '0', and if TF_TEST is set to '1', this address is readable and writable to test the TX FIFO through the AHB bus.

#### 8.3.4.15. SPI RX Data Register

Offset: 0x300			Register Name: SPI_RXD
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:0	R	0	RDATA Receive Data This register can be accessed in byte, half-word or word unit by AHB. In byte accessing method, if there are data in RXFIFO, the top word is returned and the RXFIFO depth is decreased by 1. In half-word accessing method, two SPI bursts are returned and the RXFIFO depth is decrease by 2. In word accessing method, the four SPI bursts are returned and

the RXFIFO depth is decreased by 4.

Note: This address is read-only if RF\_TEST is '0', and if RF\_TEST is set to '1', this address is readable and writable to test the RX FIFO through the AHB bus.

#### 8.3.4.16. SPI Special Requirement

#### 8.3.5. SPI Pin List

The direction of SPI pin is different in two work modes: Master Mode and Slave Mode.

Port Name	Width	Direction(M)	Direction(S)	Description
SPI_SCLK	1	OUT	IN	SPI Clock
SPI_MOSI	1	OUT	IN	SPI Master Output Slave Input Data Signal
SPI_MISO	1	IN	OUT	SPI Master Input Slave Output Data Signal
SPI_SS[3:0]	4	OUT	IN	SPI Chip Select Signal

#### 8.3.6. SPI Module Clock Source and Frequency

The SPI module uses two clock source: AHB\_CLK and SPI\_CLK. The SPI\_SCLK can in the range from 3Khz to 100 MHZ and  $AHB\_CLK \geq 2 \times SPI\_SCLK$ .

Clock Name	Description	Requirement
AHB_CLK	AHB bus clock, as the clock source of SPI module	$AHB\_CLK \geq 2 \times SPI\_SCLK$
SPI_CLK	SPI serial input clock	

## 8.4. UART

### 8.4.1. Overview

The UART is used for serial communication with a peripheral, modem (data carrier equipment, DCE) or data set. Data is written from a master (CPU) over the APB bus to the UART and it is converted to serial form and transmitted to the destination device. Serial data is also received by the UART and stored for the master (CPU) to read back.

The UART contains registers to control the character length, baud rate, parity generation/checking, and interrupt generation. Although there is only one interrupt output signal from the UART, there are several prioritized interrupt types that can be responsible for its assertion. Each of the interrupt types can be separately enabled/disabled with the control registers.

The UART has 16450 and 16550 modes of operation, which are compatible with a range of standard software drivers. In 16550 mode, transmit and receive operations are both buffered by FIFOs. In 16450 mode, these FIFOs are disabled.

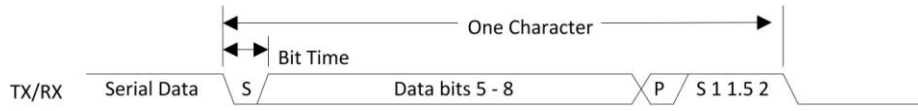
The UART supports data lengths from five to eight bits, an optional parity bit and 1, 1 ½ or 2 stop bits, and is fully programmable by an AMBA APB CPU interface. A 16-bit programmable baud rate generator and an 8-bit scratch register are included, together with separate transmit and receive FIFOs. Eight modem control lines and a diagnostic loop-back mode are provided.

Interrupts can be generated for a range of TX Buffer/FIFO, RX Buffer/FIFO, Modem Status and Line Status conditions.

The UART includes the following features:

- Compatible with industry-standard 16550 UARTs
- 64-Bytes Transmit and receive data FIFOs
- DMA controller interface
- Software/ Hardware Flow Control
- Programmable Transmit Holding Register Empty interrupt
- Interrupt support for FIFOs, Status Change

### 8.4.2. UART Timing Diagram



UART Serial Data Format

### 8.4.3. UART Register List

There are 7 UART controllers. UART1 has full modem control signals, including RTS, CTS, DTR, DSR, DCD and RING signal.

Module Name	Base Address
UART0	0x07000000
UART1	0x07000400
UART2	0x07000800
UART3	0x07000C00
UART4	0x07001000
UART5	0x07001400
R-UART	0x08002800

Register Name	Offset	Description
UART_RBR	0x00	UART Receive Buffer Register
UART_THR	0x00	UART Transmit Holding Register
UART_DLL	0x00	UART Divisor Latch Low Register
UART_DLH	0x04	UART Divisor Latch High Register
UART_IER	0x04	UART Interrupt Enable Register
UART_IIR	0x08	UART Interrupt Identity Register
UART_FCR	0x08	UART FIFO Control Register
UART_LCR	0x0C	UART Line Control Register
UART_MCR	0x10	UART Modem Control Register
UART_LSR	0x14	UART Line Status Register
UART_MSR	0x18	UART Modem Status Register
UART_SCH	0x1C	UART Scratch Register
UART_USR	0x7C	UART Status Register
UART_TFL	0x80	UART Transmit FIFO Level
UART_RFL	0x84	UART_RFL
UART_HSK	0x88	UART DMA Handshake Config Register
UART_HALT	0xA4	UART Halt TX Register

## 8.4.4. UART Register Description

### 8.4.4.1. UART Receiver Buffer Register

Offset: 0x00			Register Name: UART_RBR
Offset: 0x00			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R	0	<p>RBR</p> <p>Receiver Buffer Register</p> <p>Data byte received on the serial input port . The data in this register is valid only if the Data Ready (DR) bit in the Line Status Register (LCR) is set.</p> <p>If in FIFO mode and FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO is preserved, but any incoming data are lost and an overrun error occurs.</p>

### 8.4.4.2. UART Transmit Holding Register

Offset: 0x00			Register Name: UART_THR
Offset: 0x00			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	W	0	<p>THR</p> <p>Transmit Holding Register</p> <p>Data to be transmitted on the serial output port . Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set.</p> <p>If in FIFO mode and FIFOs are enabled (FCR[0] = 1) and THRE is set, 16 number of characters of data may be written to the THR before the FIFO is full. Any attempt to write data when the FIFO is full results in the write data being lost.</p>

#### 8.4.4.3. UART Divisor Latch Low Register

<b>Offset: 0x00</b>			<b>Register Name: UART_DLL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
			<p>DLL</p> <p>Divisor Latch Low</p> <p>Lower 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero).</p> <p>The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor).</p> <p>Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.</p>
7:0	R/W	0	

#### 8.4.4.4. UART Divisor Latch High Register

<b>Offset: 0x04</b>			<b>Register Name: UART_DLH</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
			<p>DLH</p> <p>Divisor Latch High</p> <p>Upper 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero).</p>
7:0	R/W	0	



			<p>The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor).</p> <p>Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLH is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.</p>
--	--	--	--

#### 8.4.4.5. UART Interrupt Enable Register

Offset: 0x04			Register Name: UART_IER
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:8	/	/	/ PTIME Programmable THRE Interrupt Mode Enable This is used to enable/disable the generation of THRE Interrupt. 0: Disable 1: Enable
7	R/W		
6:4	/	/	/
			EDSSI Enable Modem Status Interrupt This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0: Disable 1: Enable
3	R/W	0	
			ELSI Enable Receiver Line Status Interrupt This is used to enable/disable the generation of Receiver Line Status
2	R/W	0	

			Interrupt. This is the highest priority interrupt.  0: Disable  1: Enable
1	R/W	0	ETBEI  Enable Transmit Holding Register Empty Interrupt  This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third highest priority interrupt.  0: Disable  1: Enable
0	R/W	0	ERBFI  Enable Received Data Available Interrupt  This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFOs enabled). These are the second highest priority interrupts.  0: Disable  1: Enable

#### 8.4.4.6. UART Interrupt Identity Register

<b>Offset: 0x08</b>			<b>Register Name: UART_IIR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/ FEFLAG  FIFOs Enable Flag  This is used to indicate whether the FIFOs are enabled or disabled.  00: Disable  11: Enable
7:6	R	0	
5:4	/	/	/

3:0	R	0x1	<p>IID</p> <p>Interrupt ID</p> <p>This indicates the highest priority pending interrupt which can be one of the following types:</p> <p>0000: modem status</p> <p>0001: no interrupt pending</p> <p>0010: THR empty</p> <p>0100: received data available</p> <p>0110: receiver line status</p> <p>0111: busy detect</p> <p>1100: character timeout</p> <p>Bit 3 indicates an interrupt can only occur when the FIFOs are enabled and used to distinguish a Character Timeout condition interrupt.</p>
-----	---	-----	---

Interrupt ID	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset
0001	-	None	None	-
0110	Highest	Receiver line status	Overrun/parity/ framing errors or break interrupt	Reading the line status register
0100	Second	Received data available	Receiver data available (non-FIFO mode or FIFOs disabled) or RCVR FIFO trigger level reached (FIFO mode and FIFOs enabled)	Reading the receiver buffer register (non-FIFO mode or FIFOs disabled) or the FIFO drops below the trigger level (FIFO mode and FIFOs enabled)
1100	Second	Character timeout indication	No characters in or out of the RCVR FIFO during the last 4 character times and there is at least 1 character in it during This time	Reading the receiver buffer register
0010	Third	Transmit holding register empty	Transmitter holding register empty (Program THRE Mode disabled) or XMIT FIFO at or below threshold (Program THRE	Reading the IIR register (if source of interrupt); or, writing into THR (FIFOs or THRE Mode not selected or disabled) or XMIT FIFO above

			Mode enabled)	threshold (FIFOs and THRE Mode selected and enabled).
0000	Fourth	Modem status	Clear to send or data set ready or ring indicator or data carrier detect. Note that if auto flow control mode is enabled, a change in CTS (that is, DCTS set) does not cause an interrupt.	Reading the Modem status Register
0111	Fifth	Busy detect indication	UART_16550_COMPATIBLE = NO and master has tried to write to the Line Control Register while the UART is busy (USR[0] is set to one).	Reading the UART status register

#### 8.4.4.7. UART FIFO Control Register

<b>Offset: 0x08</b>			<b>Register Name: UART_FCR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
			RT  RCVR Trigger  This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. In auto flow control mode it is used to determine when the rts_n signal is de-asserted. It also determines when the dma_rx_req_n signal is asserted in certain modes of operation.  00: 1 character in the FIFO  01: FIFO ¼ full  10: FIFO ½ full
7:6	W	0	11: FIFO-2 less than full
			TFT  TX Empty Trigger
5:4	W	0	Writes have no effect when THRE_MODE_USER = Disabled. This is used

			<p>to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. It also determines when the dma_tx_req_n signal is asserted when in certain modes of operation.</p> <p>00: FIFO empty</p> <p>01: 2 characters in the FIFO</p> <p>10: FIFO ¼ full</p> <p>11: FIFO ½ full</p>
3	W	0	<p>DMAM</p> <p>DMA Mode</p> <p>0: Mode 0</p> <p>1: Mode 1</p>
2	W	0	<p>XFIFOR</p> <p>XMIT FIFO Reset</p> <p>This resets the control portion of the transmit FIFO and treats the FIFO as empty. This also de-asserts the DMA TX request.</p> <p>It is 'self-clearing'. It is not necessary to clear this bit.</p>
1	W	0	<p>RFIFOR</p> <p>RCVR FIFO Reset</p> <p>This resets the control portion of the receive FIFO and treats the FIFO as empty. This also de-asserts the DMA RX request.</p> <p>It is 'self-clearing'. It is not necessary to clear this bit.</p>
0	W	0	<p>FIFOE</p> <p>Enable FIFOs</p> <p>This enables/disables the transmit (XMIT) and receive (RCVR) FIFOs. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFOs is reset.</p>

#### 8.4.4.8. UART Line Control Register

Offset: 0x0C			Register Name: UART_LCR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:8	/	/	/
7	R/W	0	<p>DLAB</p> <p>Divisor Latch Access Bit</p> <p>It is writeable only when UART is not busy (USR[0] is zero) and always readable. This bit is used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.</p> <p>0: Select RX Buffer Register (RBR) / TX Holding Register(THR) and Interrupt Enable Register (IER)</p> <p>1: Select Divisor Latch LS Register (DLL) and Divisor Latch MS Register (DLM)</p>
6	R/W	0	<p>BC</p> <p>Break Control Bit</p> <p>This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If SIR_MODE = Enabled and active (MCR[6] set to one) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver and the sir_out_n line is forced low.</p>
5:4	R/W	0	<p>EPS</p> <p>Even Parity Select</p> <p>It is writeable only when UART is not busy (USR[0] is zero) and always writable readable. This is used to select between even and odd parity, when parity is enabled (PEN set to one). Setting the LCR[5] is to reverse the LCR[4].</p>

			00: Odd Parity  01: Even Parity  1X: Reverse LCR[4]
3	R/W	0	PEN  Parity Enable  It is writeable only when UART is not busy (USR[0] is zero) and always readable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively.  0: parity disabled  1: parity enabled
2	R/W	0	STOP  Number of stop bits  It is writeable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits is transmitted. Otherwise, two stop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit.  0: 1 stop bit  1: 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit
1:0	R/W	0	DLS  Data Length Select  It is writeable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected areas follows:  00: 5 bits  01: 6 bits

			10: 7 bits
			11: 8 bits

#### 8.4.4.9. UART Modem Control Register

Offset: 0x10			Register Name: UART_MCR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:6	/	/	/
5	R/W	0	AFCE Auto Flow Control Enable When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, Auto Flow Control features are enabled. 0: Auto Flow Control Mode disabled 1: Auto Flow Control Mode enabled
4	R/W	0	LOOP Loop Back Mode 0: Normal Mode 1: Loop Back Mode This is used to put the UART into a diagnostic mode for test purposes. If operating in UART mode (SIR_MODE != Enabled or not active, MCR[6] set to zero), data on the sout line is held high, while serial data output is looped back to the sin line, internally. In this mode all the interrupts are fully functional. Also, in loopback mode, the modem control inputs (dsr_n, cts_n, ri_n, dcd_n) are disconnected and the modem control outputs (dtr_n, rts_n, out1_n, out2_n) are looped back to the inputs, internally. If operating in infrared mode (SIR_MODE == Enabled AND active, MCR[6] set to one), data on the sir_out_n line is held low, while serial data output is inverted and looped back to the sir_in line.
3: 2	/	/	/
1	R/W	0	RTS



			<p>Request to Send</p> <p>This is used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto RTS Flow Control is not enabled (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. In Auto Flow Control, AFCE_MODE == Enabled and active (MCR[5] set to one) and FIFOs enable (FCR[0] set to one), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal is de-asserted when MCR[1] is set low.</p> <p>0: rts_n de-asserted (logic 1)</p> <p>1: rts_n asserted (logic 0)</p> <p>Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive high while the value of this location is internally looped back to an input.</p>
0	R/W	0	<p>DTR</p> <p>Data Terminal Ready</p> <p>This is used to directly control the Data Terminal Ready (dtr_n) output. The value written to this location is inverted and driven out on dtr_n.</p> <p>0: dtr_n de-asserted (logic 1)</p> <p>1: dtr_n asserted (logic 0)</p> <p>The Data Terminal Ready output is used to inform the modem or data set that the UART is ready to establish communications.</p> <p>Note that in Loopback mode (MCR[4] set to one), the dtr_n output is held inactive high while the value of this location is internally looped back to an input.</p>

#### 8.4.4.10. UART Line Status Register

<b>Offset: 0x14</b>	<p><b>Register Name: UART_LSR</b></p> <p><b>Default Value: 0x0000_0060</b></p>
---------------------	--

Bit	Read/Write	Default	Description
31:8	/	/	/
7	R	0	<p>FIFOERR</p> <p>RX Data Error in FIFO</p> <p>When FIFOs are disabled, this bit is always 0. When FIFOs are enabled, this bit is set to 1 when there is at least one PE, FE, or BI in the RX FIFO. It is cleared by a read from the LSR register provided there are no subsequent errors in the FIFO.</p>
6	R	1	<p>TEMT</p> <p>Transmitter Empty</p> <p>If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register and the TX Shift Register are empty. If the FIFOs are enabled, this bit is set whenever the TX FIFO and the TX Shift Register are empty. In both cases, this bit is cleared when a byte is written to the TX data channel.</p>
5	R	1	<p>THRE</p> <p>TX Holding Register Empty</p> <p>If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register is empty and ready to accept new data and it is cleared when the CPU writes to the TX Holding Register.</p> <p>If the FIFOs are enabled, this bit is set to "1" whenever the TX FIFO is empty and it is cleared when at least one byte is written to the TX FIFO.</p>
4	R	0	<p>BI</p> <p>Break Interrupt</p> <p>This is used to indicate the detection of a break sequence on the serial input data.</p> <p>It is set whenever the serial input, <i>sin</i>, is held in a logic '0' state for longer than the sum of <i>start time + data bits + parity + stop bits</i>.</p> <p>In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.</p>

3	R	0	<p>FE</p> <p>Framing Error</p> <p>This is used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data.</p> <p>In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs, the UART tries to resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]).</p> <p>0: no framing error</p> <p>1: framing error</p> <p>Reading the LSR clears the FE bit.</p>
2	R	0	<p>PE</p> <p>Parity Error</p> <p>This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]).</p> <p>0: no parity error</p> <p>1: parity error</p> <p>Reading the LSR clears the PE bit.</p>
1	R	0	<p>OE</p> <p>Overrun Error</p> <p>This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new</p>

			<p>character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost.</p> <p>0: no overrun error</p> <p>1: overrun error</p> <p>Reading the LSR clears the OE bit.</p>
0	R	0	<p>DR</p> <p>Data Ready</p> <p>This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO.</p> <p>0: no data ready</p> <p>1: data ready</p> <p>This bit is cleared when the RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.</p>

#### 8.4.4.11. UART Modem Status Register

<b>Offset: 0x18</b>			<b>Register Name: UART_MSR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7	R	0	<p>DCD</p> <p>Line State of Data Carrier Detect</p> <p>This is used to indicate the current state of the modem control line dcd_n. This bit is the complement of dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set.</p> <p>0: dcd_n input is de-asserted (logic 1)</p> <p>1: dcd_n input is asserted (logic 0)</p>

6	R	0	<p>RI</p> <p>Line State of Ring Indicator</p> <p>This is used to indicate the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set.</p> <p>0: ri_n input is de-asserted (logic 1)</p> <p>1: ri_n input is asserted (logic 0)</p>
5	R	0	<p>DSR</p> <p>Line State of Data Set Ready</p> <p>This is used to indicate the current state of the modem control line dsr_n. This bit is the complement of dsr_n. When the Data Set Ready input (dsr_n) is asserted it is an indication that the modem or data set is ready to establish communications with UART.</p> <p>0: dsr_n input is de-asserted (logic 1)</p> <p>1: dsr_n input is asserted (logic 0)</p> <p>In Loopback Mode (MCR[4] set to one), DSR is the same as MCR[0] (DTR).</p>
4	R	0	<p>CTS</p> <p>Line State of Clear To Send</p> <p>This is used to indicate the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with UART.</p> <p>0: cts_n input is de-asserted (logic 1)</p> <p>1: cts_n input is asserted (logic 0)</p> <p>In Loopback Mode (MCR[4] = 1), CTS is the same as MCR[1] (RTS).</p>
3	R	0	<p>DDCD</p> <p>Delta Data Carrier Detect</p>

			<p>This is used to indicate that the modem control line dcd_n has changed since the last time the MSR was read.</p> <p>0: no change on dcd_n since last read of MSR</p> <p>1: change on dcd_n since last read of MSR</p> <p>Reading the MSR clears the DDCD bit.</p> <p>Note: If the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.</p>
2	R	0	<p>TERI</p> <p>Trailing Edge Ring Indicator</p> <p>This is used to indicate that a change on the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the MSR was read.</p> <p>0: no change on ri_n since last read of MSR</p> <p>1: change on ri_n since last read of MSR</p> <p>Reading the MSR clears the TERI bit.</p>
1	R	0	<p>DDSR</p> <p>Delta Data Set Ready</p> <p>This is used to indicate that the modem control line dsr_n has changed since the last time the MSR was read.</p> <p>0: no change on dsr_n since last read of MSR</p> <p>1: change on dsr_n since last read of MSR</p> <p>Reading the MSR clears the DDSR bit. In Loopback Mode (MCR[4] = 1), DDSR reflects changes on MCR[0] (DTR).</p> <p>Note: If the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDSR bit is set when the reset is removed if the dsr_n signal remains asserted.</p>
0	R	0	<p>DCTS</p>

			<p>Delta Clear to Send</p> <p>This is used to indicate that the modem control line cts_n has changed since the last time the MSR was read.</p> <p>0: no change on ctsdsr_n since last read of MSR</p> <p>1: change on ctsdsr_n since last read of MSR</p> <p>Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] = 1), DCTS reflects changes on MCR[1] (RTS).</p> <p>Note: If the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.</p>
--	--	--	---

#### 8.4.4.12. UART Scratch Register

<b>Offset: 0x1C</b>			<b>Register Name: UART_SCH</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R/W	0	SCRATCH_REG Scratch Register This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART.

#### 8.4.4.13. UART Status Register

<b>Offset: 0x7C</b>			<b>Register Name: UART_USR</b> <b>Default Value: 0x0000_0006</b>
Bit	Read/Write	Default	Description
31:5	/	/	/
4	R	0	RFF Receive FIFO Full

			<p>This is used to indicate that the receive FIFO is completely full.</p> <p>0: Receive FIFO not full</p> <p>1: Receive FIFO Full</p> <p>This bit is cleared when the RX FIFO is no longer full.</p>
3	R	0	<p>RFNE</p> <p>Receive FIFO Not Empty</p> <p>This is used to indicate that the receive FIFO contains one or more entries.</p> <p>0: Receive FIFO is empty</p> <p>1: Receive FIFO is not empty</p> <p>This bit is cleared when the RX FIFO is empty.</p>
2	R	1	<p>TFE</p> <p>Transmit FIFO Empty</p> <p>This is used to indicate that the transmit FIFO is completely empty.</p> <p>0: Transmit FIFO is not empty</p> <p>1: Transmit FIFO is empty</p> <p>This bit is cleared when the TX FIFO is no longer empty.</p>
1	R	1	<p>TFNF</p> <p>Transmit FIFO Not Full</p> <p>This is used to indicate that the transmit FIFO is not full.</p> <p>0: Transmit FIFO is full</p> <p>1: Transmit FIFO is not full</p> <p>This bit is cleared when the TX FIFO is full.</p>
0	R	0	<p>BUSY</p> <p>UART Busy Bit</p>



			0: Idle or inactive
			1: Busy

#### 8.4.4.14. UART Transmit FIFO Level Register

<b>Offset: 0x80</b>			<b>Register Name: UART_TFL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:0	R	0	TFL Transmit FIFO Level This is indicates the number of data entries in the transmit FIFO.

#### 8.4.4.15. UART Receive FIFO Level Register

<b>Offset: 0x84</b>			<b>Register Name: UART_RFL</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:7	/	/	/
6:0	R	0	RFL Receive FIFO Level This is indicates the number of data entries in the receive FIFO.

#### 8.4.4.16. UART DMA Handshake Config Register

<b>Offset: 0x88</b>			<b>Register Name: UART_HSK</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R/W	0xA5	Handshake configuration

			<b>0xA5: DMA wait cycle mode</b>  <b>0xE5: DMA handshake mode</b>
--	--	--	---

#### 8.4.4.17. UART Halt TX Register

Offset: 0xA4			Register Name: UART_HALT
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:4	/	/	/
3	/	/	/
			CHANGE_UPDATE
			After the user using HALT[1] to change the baudrate or LCR configuration, write 1 to update the configuration and waiting this bit self clear to 0 to finish update process. Write 0 to this bit has no effect.
2	R/W	0	1: Update trigger, Self clear to 0 when finish update.
			CHCFG_AT_BUSY
			This is an enable bit for the user to change LCR register configuration (except for the DLAB bit) and baudrate register (DLH and DLL) when the UART is busy (USB[0] is 1).
1	R/W	0	1: Enable change when busy
			HALT_TX
			Halt TX
			This register is use to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled.
			0 : Halt TX disabled
			1 : Halt TX enabled
			Note: If FIFOs are not enabled, the setting of the halt TX register has no effect on operation.
0	R/W	0	

#### 8.4.4.18. UART DBG DLL Register

<b>Offset: 0xB0</b>			<b>Register Name: UART_DBG_DLL</b>
			<b>Default Value: 0x0000_0000</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:8	/	/	/
7:0	R/W	0	DEBUG DLL

#### 8.4.4.19. UART DBG DLH Register

<b>Offset: 0xB4</b>			<b>Register Name: UART_DBG_DLH</b>
			<b>Default Value: 0x0000_0000</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:8	/	/	/
7:0	R/W	0	DEBUG DLH

### 8.4.5. UART Pin List

Port Name	Width	Direction	Description
UART0_TX	1	OUT	UART Serial Bit output
UART0_RX	1	IN	UART Serial Bit input
UART1_TX	1	OUT	UART Serial Bit output
UART1_RX	1	IN	UART Serial Bit input
UART1_RTS	1	OUT	UART Request To Send  This active low output signal informs Modem that the UART is ready to send data
UART1_CTS	1	IN	UART Clear To End  This active low signal is an input showing when Modem is ready to accept data
UART1_DTR	1	OUT	UART Data Terminal Ready  This active low output signal informs Modem that the UART is ready to establish a communication link
UART1_DSR	1	IN	UART Data Set Ready  This active low signal is an input indicating when Modem is ready to set up a link with the UART0
UART1_DCD	1	IN	UART Data Carrier Detect  This active low signal is an input indicating when Modem has detected a carrier
UART1_RING	1	IN	UART Ring Indicator  This active low signal is an input showing when Modem has sensed a ring signal on the telephone line
UART2_TX	1	OUT	UART Serial Bit output
UART2_RX	1	IN	UART Serial Bit input
UART2_RTS	1	OUT	UART Request To Send  This active low output signal informs Modem that the UART is ready to send data
UART2_CTS	1	IN	UART Clear To End  This active low signal is an input showing when Modem is ready to accept data

UART3_TX	1	OUT	UART Serial Bit output
UART3_RX	1	IN	UART Serial Bit input
UART4_TX	1	OUT	UART Serial Bit output
UART4_RX	1	IN	UART Serial Bit input
UART4_RTS	1	OUT	UART Request To Send  This active low output signal informs Modem that the UART is ready to send data
UART4_CTS	1	IN	UART Clear To End  This active low signal is an input showing when Modem is ready to accept data
UART5_TX	1	OUT	UART Serial Bit output
UART5_RX	1	IN	UART Serial Bit input
UART5_RTS	1	OUT	UART Request To Send  This active low output signal informs Modem that the UART is ready to send data
UART5_CTS	1	IN	UART Clear To End  This active low signal is an input showing when Modem is ready to accept data
R_UART_TX	1	OUT	UART Serial Bit output
R_UART_RX	1	IN	UART Serial Bit input

## 8.5. RSB

### 8.5.1. Overview

The RSB (reduced serial bus) Host Controller is designed to communicate with RSB Device using two push-pull wires. It supports a simplified two wire protocol (RSB) on a push-pull bus. The transfer speed can be up to 20MHz and the performance will be improved much.

The RSB features:

- Supports industry-standard AMBA Peripheral Bus (APB) and it is fully compliant with the AMBA Specification, Revision 2.0.
- Supports speed up to 20MHz with ultra low power
- Supports Push-Pull bus
- Supports Host mode
- Supports programmable output delay of CD signal
- Supports parity check for address and data transmission
- Supports multi-devices

### 8.5.2. Terminology Definition

TERM	Description
CK	A line that is used to transmit clock from Host to Device
CD	A line that is used to transmit Command and Data between Host and Device
DA	Device Address is a 16bits address that is the ID of each type device.
RTA	Run-Time Address is an 8bits address that is used to address device during Read or Write transmission. The valid RTA is 0x17 0x2D 0x3A 0x4E 0x59 0x63 0x74 0x8B 0x9C 0xA6 0xB1 0xC5 0xD2 0xE8 and 0xFF.
HD	Host to Device Handshake is used to change the ownership of CD from Host to Device.
DH	Device to Host Handshake is used to change the ownership of CD from Device to Host.
SB	Start Bit: a HIGH to LOW transition on the CD while CK is high.

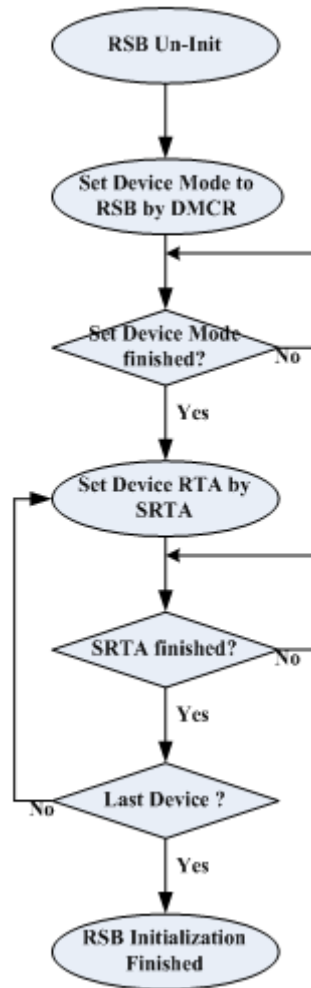
### 8.5.3. RSB Command Set

Command	Value	Description
SRTA	0xE8	Set Run-Time-Address
RD8	0x8B	Read one byte from Device
RD16	0x9C	Read two bytes from Device
RD32	0xA6	Read four bytes from Device
WR8	0x4E	Write one byte to Device
WR16	0x59	Write two bytes to Device

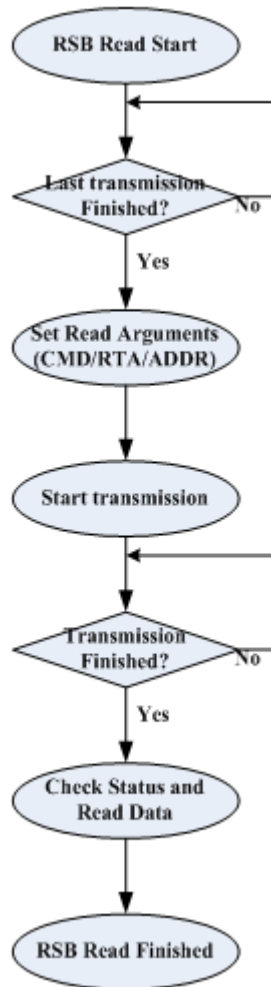
WR32	0x63	Write four bytes to Device
------	------	----------------------------

**8.5.4. Software Operation Flow**

RSB System Initialization

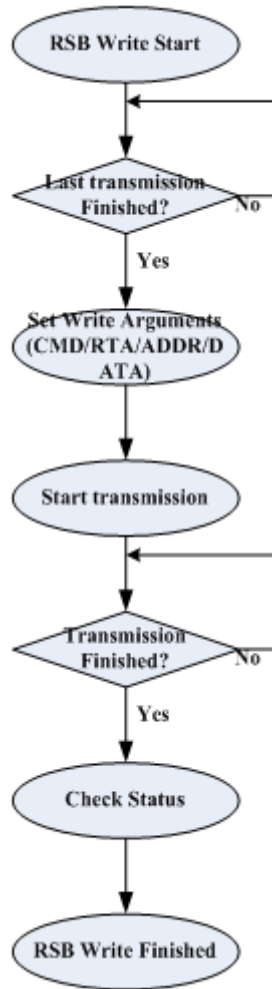


RSB Read from Device





RSB Write to Device



### 8.5.5. RSB Controller Register List

Module Name	Base Address
RSB	0x08003400

Register Name	Offset	Description
RSB_CTRL	0x0000	RSB Control Register
RSB_CCR	0x0004	RSB Clock Control Register
RSB_INTE	0x0008	RSB Interrupt Enable Register
RSB_STAT	0x000c	RSB Status Register
RSB_AR	0x0010	RSB Address Register
RSB_DATA	0x001c	RSB Data Buffer Register
RSB_LCR	0x0024	RSB Line Control register
RSB_DMCR	0x0028	RSB Device Mode Control register
RSB_CMD	0x002C	RSB Command Register
RSB_DAR	0x0030	RSB Device address Register

## 8.5.6. RBS Register Description

### 8.5.6.1. RSB Control Register

Offset: 0x00			Register Name: RSB_CTRL
Default Value: 0x0000_0000			
Bit	R/W	Default	Description
7	R/W	0	START_TRANS  Write '1' to this bit will start a new transmission with the configuration of other registers. It is cleared to '0' automatically when the transaction completes or an error happens in the transmission.
6	R/W	0	ABORT_TRANS  Write '1' to this bit will abort the current transmission. It is cleared to '0' automatically when the transmission has been aborted.
5:2	/	/	/
1	R/W	0	GLOBAL_INT_ENB  Global interrupt enable bit  1 – enable interrupt  0 – disable interrupt
0	R/W	0	Soft Reset  Write '1' to this bit will reset the controller into default state. All of the status of controller will be cleared. And this bit will be cleared to '0' automatically when reset operation completes.

### 8.5.6.2. RSB Clock Control Register

Offset: 0x04			Register Name: RSB_CCR
Default Value: 0x0000_0000			
Bit	R/W	Default	Description
31:11	/	/	/
10:8	R/W	0	CD_ODLY

			CD output delay Delay time of n source clock cycles before output CD signal.
7:0	R/W	0	CK_DIV $F_{ck} = F_{source} / 2*(divider+1)$

### 8.5.6.3. RSB Interrupt Enable Register

<b>Offset: 0x08</b>			<b>Register Name: RSB_INTE</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:3	/	/	/
2	R/W	0	LOAD_BSY_ENB Loading Busy Interrupt Enable 1 – enable 0 – disable
1	R/W	0	TRANS_ERR_ENB Transfer Error Interrupt Enable 1 – enable 0 – disable
0	R/W	0	TRANS_OVER_ENB Transfer complete Interrupt Enable 1 – enable 0 – disable

### 8.5.6.4. RSB Status Register

<b>Offset: 0x0c</b>			<b>Register Name: RSB_INTS</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:17	/	/	/

			<p>TRANS_ERR_ACK.</p> <p>If a negative ACK is received from Device, then this bit is set to '1' by hardware.</p>
16	R	0	<p>This bit is cleared when a new transmission is started.</p>
15:12	/	/	/
			<p>TRANS_ERR_DATA</p> <p>If the parity check of 1<sup>st</sup> byte is negative, then bit8 is set to '1' by hardware.</p> <p>If the parity check of 2<sup>nd</sup> byte is negative, then bit9 is set to '1' by hardware; and so on.</p>
11:8	R	0	<p>These bits are cleared when a new transmission is started.</p>
7:3	/	/	/
			<p>LOAD_BSY</p> <p>Loading Busy Flag</p> <p>If software writes any control registers during transmission, this bit will be set to '1'.</p> <p>If LOAD_BSY_ENB=1, an interrupt will be generated.</p>
2	R/W	0	<p>Software can clear this flag by writing '1' to this bit.</p>
			<p>TRANS_ERR</p> <p>Transfer Error Flag</p> <p>If an error happened during transmission, This bit will be set to '1'.</p> <p>If TRANS_ERR_ENB=1, an interrupt will be generated.</p>
1	R/W	0	<p>Software can clear this flag by writing '1' to this bit.</p>
			<p>TRANS_OVER</p> <p>Transfer Over Flag</p> <p>If the transmission has transfer over, this bit is set to '1'.</p> <p>If TRANS_OVER_ENB=1, an interrupt will be generated.</p>
0	R/W	0	<p>Software can clear this flag by writing '1' to this bit.</p>

#### 8.5.6.5. RSB Address Register

<b>Offset: 0x10</b>			<b>Register Name: RSB_AR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:8	/	/	/
7:0	R/W	0	ADDR The ADDR is send to device during Read and Write command.

#### 8.5.6.6. RSB Data Buffer Register

<b>Offset: 0x1c</b>			<b>Register Name: RSB_DATA</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:0	R/W	0	RSB DATA  If the command is configured as read, Software can read this register to get the data from device; If the command is configured as write, Software can write this register to send the data to device.  If the command is rd8 or wr8, then the low byte is active;  If the command is rd16 or wr16, then the low two byte is active;  If the command is rd32 or wr32, then the whole word is active

#### 8.5.6.7. RSB Line Control Register

<b>Offset: 0x24</b>			<b>Register Name: RSB_LCR</b>
			<b>Default Value: 0x0000_003A</b>
Bit	R/W	Default	Description
31:6	/	/	/
5	R	1	CK_STATE Current state of CK pin

			<p>0 – low</p> <p>1 - high</p>
4	R	1	<p>CD_STATE</p> <p>Current state of CD pin</p> <p>0 – low</p> <p>1 – high</p>
3	R/W	1	<p>CK_CTL</p> <p>CK line state control bit</p> <p>When line control mode is enabled (bit[2] set), value of this bit decide the output level of CK</p> <p>0 – output low level</p> <p>1 – output high level</p>
2	R/W	0	<p>CK_CTL_EN</p> <p>CK line state control enable</p> <p>When this bit is set, the state of CK is control by the value of bit[3].</p> <p>0-disable CK line control mode</p> <p>1-enable CK line control mode</p>
1	R/W	1	<p>CD_CTL</p> <p>CD line state control bit</p> <p>When line control mode is enabled (bit[0] set), value of this bit decide the output level of CD</p> <p>0 – output low level</p> <p>1 – output high level</p>
0	R/W	0	<p>CD_CTL_EN</p> <p>CD line state control enable</p> <p>When this bit is set, the state of CD is control by the value of bit [1].</p>

		0-disable CD line control mode
		1-enable CD line control mode

#### 8.5.6.8. RSB Device Mode Control Register

<b>Offset: 0x28</b>			<b>Register Name: RSB_DMCR</b>
			<b>Default Value: 0x003e3e00</b>
Bit	R/W	Default	Description
			DEVICE_MODE_START
			When set to '1', host will send DEVICE_MODE to device to switch the device's bus mode from NTWI to RSB.
31	R/W	0	This bit will be self-cleared when DEVICE_MODE is sent onto the RSB bus.
30:24	/	/	/
			DEVICE_MODE Data
23:0	R/W	0x3e3e00	The data send to device during DEVICE_MODE

#### 8.5.6.9. RSB Command Register

<b>Offset: 0x2C</b>			<b>Register Name: RSB_CMD</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:8	/	/	/
			CMD_IDX
7:0	R/W	0	command index

#### 8.5.6.10. RSB Device Address Register

<b>Offset: 0x30</b>			<b>Register Name: RSB_DAR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	R/W	Default	Description
31:24	/	/	/

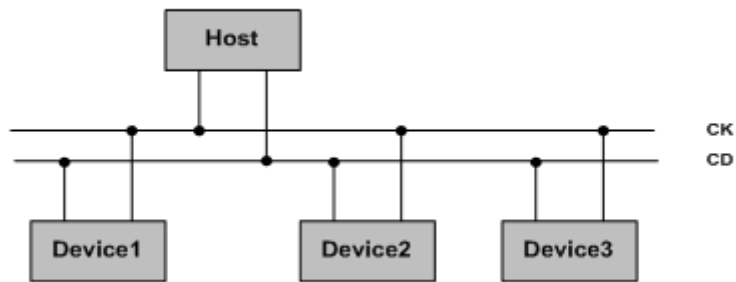


23:16	R/W	0	RTA Run-Time Address
15:0	R/W	0	DA Device Address

### 8.5.7. RSB General Specification

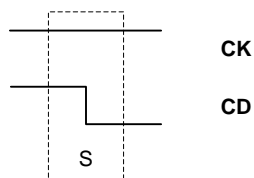
RSB uses push-pull bus, and supports multi-devices. It uses CK as clock and uses CD to transmit command and data. The Bus Topology is showed below:

RSB Bus Topology



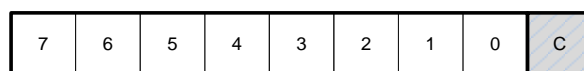
The start bit marks the beginning of a transaction. The Start bit is defined as a HIGH to LOW transition on the CD while CK is high.

Start signal



RSB protocol uses parity bit to check the correction of address and data.

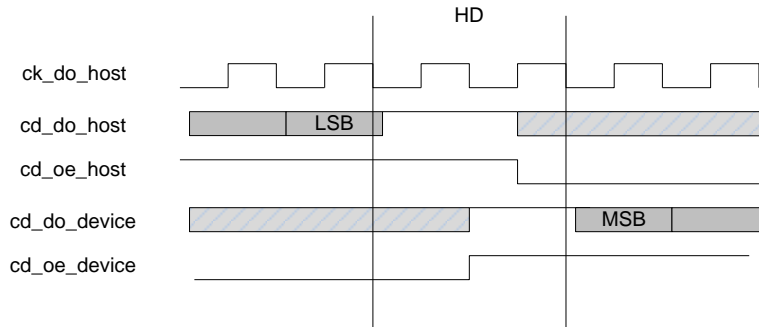
Parity bit



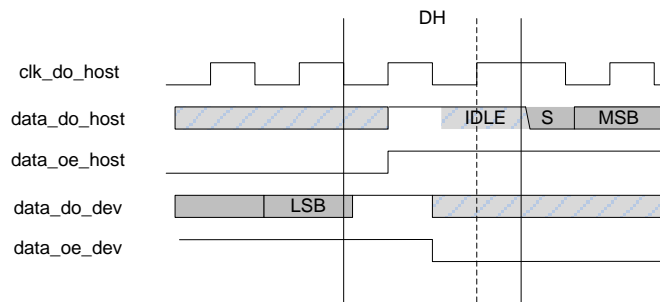
ACK bit is the acknowledgement from device to host, The ACK is low active. When device finds the parity bit is error, it will not send ACK to host, so host can know that an error happens in the transaction.

Both Host and Device can drive the CD, so there are two handshakes, HD (host to device) and DH (device to host), for Host and device to convert the direction of data transmission.

**HD Handshake**



**DH Handshake**



To improve transaction efficiency and to be flexible in device address assignment, RSB use Device Address (DA) and Run-Time Address (RTA). RTA is assigned dynamically by host. Host software shall ensure that different device has different RTA in the same system. Device’s default RTA is 0 and 0 is the reserved address. If RTA is 0 when setting RTA, the setting is invalid.

There are three command types in RSB:

- 1) Set run-time address (RTA): It is used to set run time address (RTA) for different devices in the same system. There are 15 devices in a system at most. The RTA can be selected from the RTA code set and a device's RTA can be modified many times by using set run-time address command.

**SRTA Timing**



- 2) Read command: It is used to read data from device. It has byte, half word and word operation. When the device receives the command, they shall check if the command's RTA matches their own RTA.

Read Timing



- 3) Write command: It is used to write data to the devices. It has byte, half word and word operation. When the device receives the command, they shall check if the command's RTA matches their own RTA.

Write Timing



## 8.6. One Wire Interface

### 8.6.1. Overview

The One Wire Interface implements the hardware protocol of the Master function of the 1-Wire protocol, which uses a single wire for communication between the Master (1-Wire controller) and the Slaves (1-Wire external compliant devices).

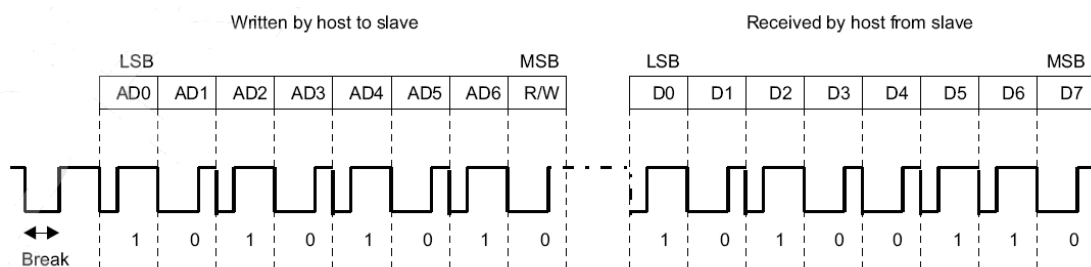
The One Wire interface is implemented as an open-drain output at the device level. Therefore, an external pullup is required and protocol use the return-to-1 mechanism (that is , after any command by any of the connected devices, the line is pulled to a logical high level).

The One Wire Interface can work at Simple mode or Standard mode at one time.

#### Simple Mode:

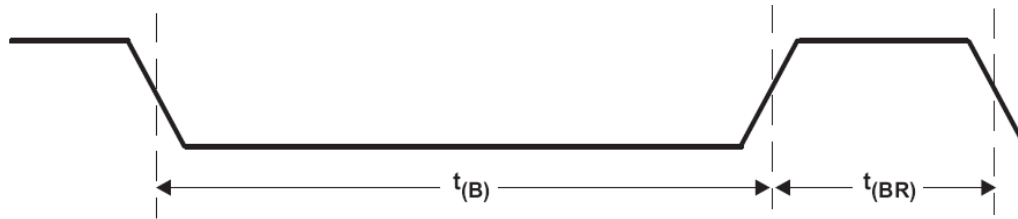
The bus of Simple Mode is a master-slave bus system using a simple one-wire, asynchronous, bi-directional, serial interface with a maximum bit-rate of about 5-Kbit/s.

It is a command-based protocol in which the host sends a command byte to the slave. The command directs the slave either to store the next eight bits of data received to a register specified by the command byte (Write command), or to output the eight bits of data from a register specified by the command byte (Read command). Command and data bytes consist of a stream of bits where the least-significant bit of a command or data byte is transmitted first. The first 7 bits of the command word are the register address and the last command bit transmitted is the read/write (R/W) bit. The following figure illustrates a typical read cycle.



In the figure, the 1 of the R/W bit indicates a write command where the 0 indicates the read command.

In Simple mode, the slave can be reset by using the break pulse. If the host does not get an expected response from the slave or if the host needs to restart a communication before it is complete, the host can hold the line low and generate a break to reset the communication engine. The Break timing is illustrated as follow.

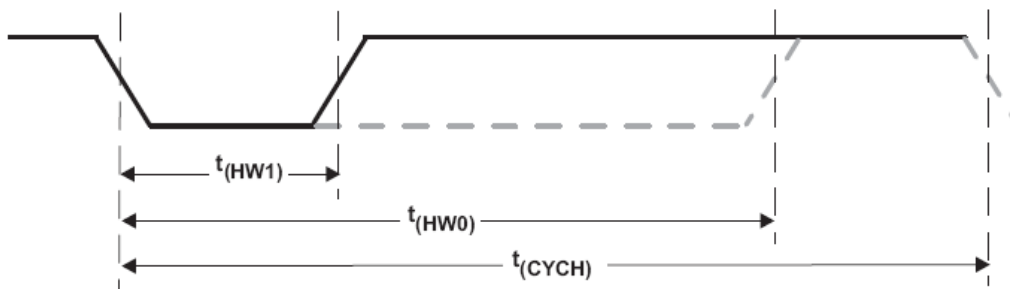


Timing Parameter	For Device	Minimum	Maximum
$t(B)$	All	190us	
$t(BR)$	All	40us	

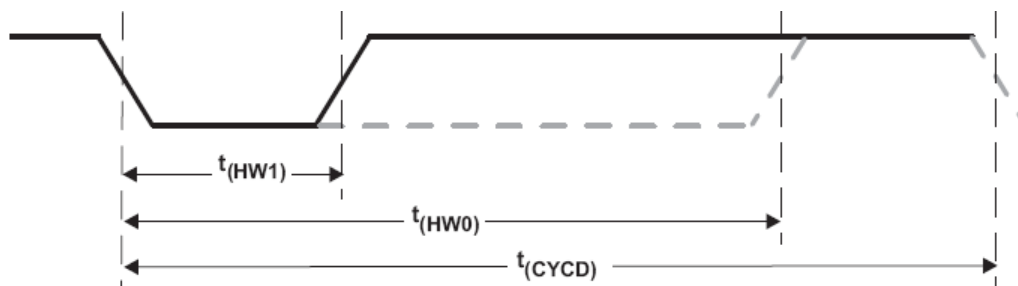
It is not required, but it is recommended to precede each communication with a break for the reliable communication.

After a successful break pulse (if have), the host and slave are ready for bit transmission. Each bit to transmit (either from the host to the slave or from the slave to the host) is preceded by a low-going edge on the line.

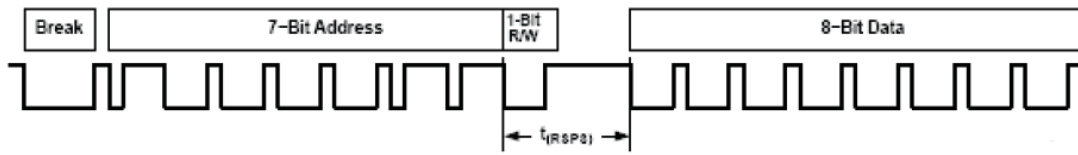
The host transmitted bit timing is showed as follow.



And the slave transmitted bit timing is showed as follow.



After the last bit of address is sent on a read cycle, the slave starts outputting the data after the specified response time,  $t(RSPS)$ . The response time is measured from the fall time of the command R/W bit to the fall time of the first data bit returned by the slave and therefore includes the entire bit time for the R/W bit. Because the minimum response time equal to the minimum bit cycle time, this means that the first data bit may begin as soon as the command R/W bit time ends. The timing is show as follow.



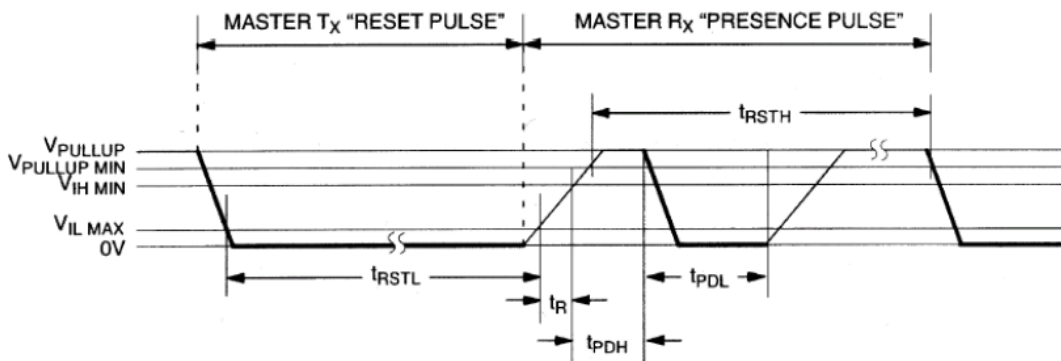
Timing Parameter	For Device	Minimum	Maximum
t(RSP3)	All	190us	320us

Also, to avoid short noise spike coupled onto the HDQ line, some filtering may be prudent.

**Standard Mode**

The Standard Mode consists of 4 types of signaling on the data line, which are Initialization Sequence, Write Zero, Write One and Read Data.

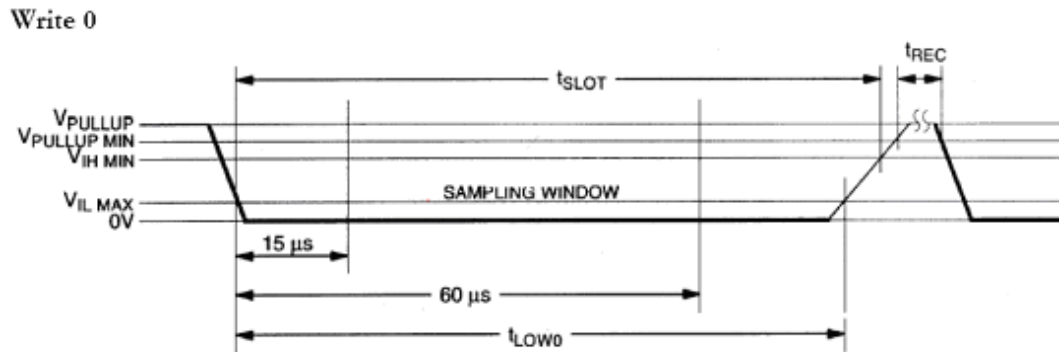
The host first sends an initialization pulse and then waits for the slave to respond with a presence pulse before enabling any communication sequence. The initialization pulse and presence pulse are showed as follow.



Timing Parameter	Minimum	Maximum
t(RSTL)	480us	
t(RSTH)	480us	
t(PDH)	15us	60us
t(PDL)	60us	240us

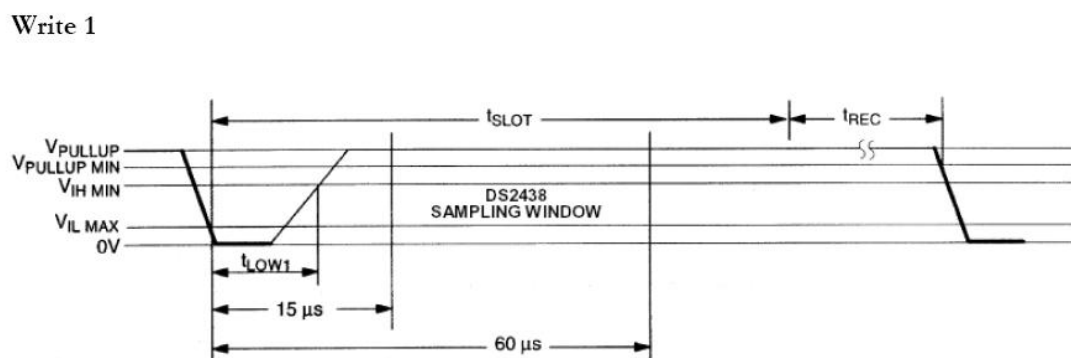
The other two types of signaling are Writing Zero and Writing One. The both write time slots must be a minimum of 60us in duration with a minimum of a 1us recovery time between individual write cycles. The slave device sample the data line in a window of 15us to 60us after the data line falls. If the line is high, a Write One occurs. If the line is low, a Write Zero occurs.

The Write Zero time slot is showed as follow.



Timing Parameter	Minimum	Maximum
T(LOW0)	60us	t(SLOT)
t(SLOT)	T(LOW0)	120 us
t(REC)	1us	

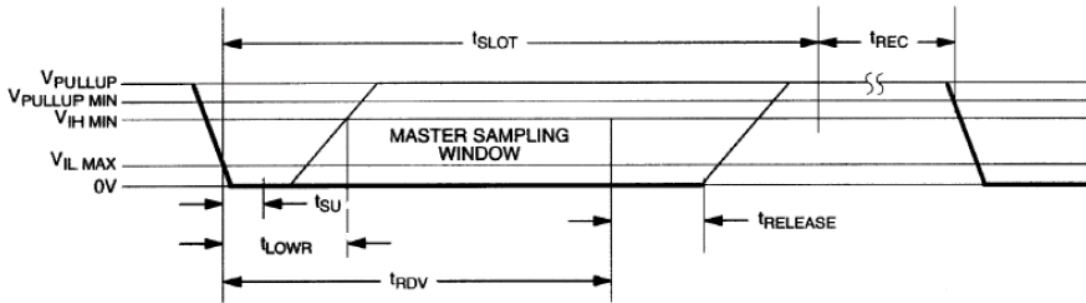
When Write One occurs, the data line must be pulled to a logic low level and then released, allowing the data line to pull up to a high level within 15us after the start of the write time slot. The Write One time slot is showed as follow.



Timing Parameter	Minimum	Maximum
t(SLOT)	60us	120 us
t(LOW1)	1us	15us
t(REC)	1us	

The last signaling type is Read Data. A read time slot is initiated when the bus master pulls the data line from a logic high level to logic low level. The data line must remain at a low logic level for a minimum of 1 us; output data from the slave is then valid within the next 14 us maximum.

The bus master therefore must stop driving the data line low in order to read its state 15 us from the start of the read slot. All read time slots must be a minimum of 60us in duration with a minimum of a 1 us recovery time between individual read slots. The Read Data slot is showed as follow.



Timing Parameter	Minimum	Maximum
t(SU)		1us
t(LOWR)	1us	15us
t(RDV)	( = 15us )	
t(RELEASE)	0us	45us
t(SLOT)	60 us	120 us
t(REC)	1us	

Cyclic Redundancy Check (CRC) is used by One Wire devices to ensure data integrity. Two different CRC are commonly found in Standard Mode. One 8 bit CRC and one 16 bit CRC. CRC8 is used in the ROM section of all devices. CRC8 is also in some devices used to verify other data, like commands issued on the bus. CRC16 is used by some devices to check for errors on larger data sets.



### 8.6.2. One Wire Interface Register List

Module Name	Base Address
HDQ/One Wire	0x08003000

Register Name	Offset	Description
OW_DATA	0x00	One Wire Data Register
OW_CTL	0x04	One Wire Control Register
OW_SMSC	0x08	One Wire Standard Mode Special Control Register
OW_SMCRC	0x0C	One Wire Standard Mode CRC Register
OW_INT_STATUS	0x10	One Wire Interrupt Status Register
OW_INT_Mask	0x14	One Wire Interrupt Mask Register
OW_FCLK	0x18	One Wire Function Clock Register
OW_LC	0x1C	One Wire Line Control Register
SM_WR_RD_TCTL	0x20	Standard Mode Write Read Timing Control Register
SM_RST_PRESENCE_TCTL	0x24	Standard Mode Reset Presence Timing Control Register
SP_WR_RD_TCTL	0x28	Simple Mode Timing Control Register
SP_BR_TCTL	0x2C	Simple Mode Break Timing Control Register

### 8.6.3. One Wire Interface Register Description

#### 8.6.3.1. One Wire Data Register

Offset: 0x00			Register Name: OW_DATA Default Value: 0x00000000
Bit	Read/Write	Default	Description
24:31	/	/	/
23:16	R/W	0	<b>SM_DATA</b> These fields are for Simple Mode data send or receive in a one wire transmission. After this byte data transfer finishing, a transmission complete interrupt will generate.
15:8	/	0	/
7:0	R/W	0	<b>OW_DATA</b> Data byte for transmitting or received  In Simple Mode mode, these fields are for the command byte transmission. When GO bit is set (the INITIALIZATION/BREAK bit is not set at the same time), these fields will be sent as the address and command for a Simple Mode transfer. After the command byte transmission finished, the controller in Simple Mode will send next 8 bit data from SM_DATA when the DIR bit is 1 or receive one byte data to SM_DATA when the DIR bit is 0.  In Standard Mode, if the INITIALIZATION/BREAK bit is not set, the controller samples/sends data to/from these fields determining by the DIR bit when the Go bit is set. When the ONE_WIRE_SINGLE_BIT is enabled, only the first bit of these fields is available.

#### 8.6.3.2. One Wire Control Register

Offset: 0x04			Register Name: OW_CTL Default Value: 0x00030000
Bit	Read/Write	Default	Description

31:24	/	/	/
23:20	/	/	/
19:16	R/W	0x3	<b>SAMPLE_TIME</b> These fields determine the sample times in digital circuit.
15:10	/	/	/
9	R/W	0	<b>INNER_PULL_UP_ENABLE</b> When this bit is set, the inner pull up for one wire bus is determined by inner output (pull up is off when bus is drive 0)  0: inner pull up is on 1: inner pull up is off when bus is drive 0
8	R/W	0	<b>AUTOIDLE</b> Auto Idle  0: Module clock is free-running; 1: Module clock is in power saving mode: the function clock is running only when module is accessed or inside logic is in function to process events.
7	/	/	/
6	R	0	<b>PRESENCEDETECT</b> Slave Presence Indicator  This read-only flag is only used in Standard Mode The value of this field indicates whether there is Presence Pulse responding to the host initialization pulse. The flag is updated when the <b>OW_INT_STATUS[0]</b> Presence Detect Interrupt Flag is set.
5	R/W	0	<b>STANDARD_MODE_SINGLE_BIT</b> The single-bit mode is only supported for Standard Mode. After the bit is transferred, Tx-complete or Rx-complete interrupt will generate for corresponding transfer operation.  0: Disabled 1: Enabled
4	R/W	0	<b>Go</b>

			<p><b>Go Bit</b></p> <p>Write 1 to start the appropriate operation.</p> <p>If the <b>INITIALIZATION/BREAK</b> bit is set, the controller generates the initialization or break pulse.</p> <p>If the <b>INITIALIZATION/BREAK</b> bit is not set, the controller in Standard Mode samples/sends data to/from the <b>OW_DATA</b> fields determining by the <b>DIR</b> bit, or controller in Simple Mode begins a transfer sequence with the command byte in <b>OW_DATA</b>.</p> <p>Bit returns to 0 after the operation is complete.</p>
3	R/W	0	<p><b>INITIALIZATION/BREAK</b></p> <p>Initialization/Break Bit</p> <p>Write 1 to send initialization pulse for the Standard Mode or break pulse for the Simple Mode. The <b>OW_DATA</b> register will be flushed when initialization or the break situation is generating. Bit returns to 0 after pulse is sent.</p> <p>The pulse generates after the <b>Go</b> bit is set.</p>
2	R/W	0	<p><b>DIR</b></p> <p>Direction Bit</p> <p>In Standard Mode, this field determines if next operation (byte operation or bit operation) is read or write.</p> <p>In Simple Mode, this field determines if the current transfer sequence is read or write.</p> <p>0 = read 1 = write</p> <p>The operation starts after the Go bit is set.</p>
1	R/W	0	<p><b>MS</b></p> <p>Mode Selection Bit</p> <p>0: Standard Mode 1: Simple Mode</p>
			<p><b>GEN</b></p> <p>Global Enable</p> <p>This field is used to enable or disable the One Wire Controller. A disable on this bit overrides any other block or channel enables and flushes all</p>

0	R/W	0	FIFOs. 0: Disable 1: Enable
---	-----	---	-----------------------------------

### 8.6.3.3. One WIRE STANDARD MODE SPECIAL CONTROL REGISTER

Offset: 0x08			Register Name: OW_SMSC
Bit			Default Value: 0x00000000
Bit	Read/Write	Default	Description
31:6	/	/	/
5	R/W	0	<b>CRC_ERROR_STATUS</b> These fields indicate the result of the CRC comparing. 0: CRC comparing right 1: CRC comparing wrong
4	/	/	/
3	R/W	0	<b>MEM_CRC_COMPARE</b> This field is only used in Standard Mode. When this field is set, the controller will compare the value in the CRC_RECEV field with the data read from the CRC_CALC_INDICATE field, and then returns corresponding result in the CRC_ERROR_STATUS field and generates CRC finish interrupt. The CRC shift register and CRC_CALC_INDICATE field will be cleaned to 0 then. This field will be automatically cleaned when the CRC compare is finish.
2	R/W	0	<b>CRC_16BIT_EN</b> This field is only used in Standard Mode. and is set to 1 to select 16bit CRC, else the 8bit CRC is select. 0: CRC_8BIT_EN 1: CRC_16BIT_EN
1	R/W	0	<b>WR_MEM_CRC_REQ</b>

			This field is only used in One Wire mode. When this bit is set, the bit send to the device will be took into calculate the CRC value (CRC8 or CRC16). The calculation will stop when this bit is cleaned. The value will be preserved in the corresponding CRC (CRC8 or CRC16) shift register then.
0	R/W	0	<p><b>RD_MEM_CRC_REQ</b></p> <p>This field is only used in Standard Mode.. When this bit is set, the bit received from the device will be took into calculate the CRC value (CRC8 or CRC16). The calculation will stop when this bit is cleaned. The value will be preserved in the corresponding CRC (CRC8 or CRC16) shift register then.</p>

#### 8.6.3.4. One Wire Standard Mode CRC Register

<b>Offset: 0x0c</b>			<b>Register Name: OW_SMCRC</b> <b>Default Value: 0x00000000</b>
Bit	Read/Write	Default	Description
31:16	R		<p><b>CRC_CALC_INDICATE</b></p> <p>This field indicates the CRC value calculated by the CRC shift register.</p>
15:0	R/W	0	<p><b>CRC_RECEV</b></p> <p>The data CRC value (CRC8 or CRC16) will be written to these fields by software for CRC comparing.</p>

#### 8.6.3.5. One Wire Interrupt Status Register

<b>Offset: 0x10</b>			<b>Register Name: OW_INT_STATUS</b> <b>Default Value: 0x00000000</b>
Bit	Read/Write	Default	Description
31:6	/	/	/
5	R/W	0	<p><b>Deglitch Detected Interrupt Flag</b></p> <p>This flag indicates a deglitch in the bus. The controller looks for any</p>

			<p>glitch in the sample window for at least 1us. If the Deglitch Interrupt is enabled, an interrupt will issues when any deglitch occurs in the bus.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
4	R/W	0	<p><b>CRC Comparing Complete Interrupt Flag</b></p> <p>This flag is used in Standard Mode., and is used to indicate the CRC comparing has finished.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
3	R/W	0	<p><b>Transmission Complete Interrupt Flag</b></p> <p>In Standard Mode., the flag is set when a write operation of one byte or one bit in single-bit mode was completed. The interrupt is generated then.</p> <p>In Simple Mode., the flag is set when a write operation of one byte was completed. The interrupt is also generated.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
2	R/W	0	<p><b>Read Complete Interrupt Flag</b></p> <p>In Standard Mode., the flag is set when a byte or a bit in single-bit mode has been successfully read. The interrupt is generated then.</p> <p>In Simple Mode. the flag is set when a byte has been successfully read. The interrupt is also generated then.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
1	R/W	0	<p><b>Time-out Interrupt Flag</b></p> <p>This flag is only used in Simple Mode.. The flag is set when two event happened. The one event is that after a read command initiated by the host, the slave did not pull the line low within the specified time (512 us). The other event is that another bit transfer does not begin after a specified time (512 us) from the pre-bit beginning.</p> <p>When the above situation occurs, the interrupt generates and the value of this field is set.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
			<p><b>Presence Detect Interrupt Flag</b></p> <p>In Standard Mode., this interrupt status is set when the Initialization Pulse is completed. The interrupt is generated then and the</p>

0	R/W	0	<p>PRESENCEDETECT bit is update.</p> <p>In Simple Mode., the flag is set when the successful completion of a break pulse. The interrupt is also generated then.</p> <p>The interrupt condition is cleared by writing “1” to this field.</p>
---	-----	---	---

### 8.6.3.6. One Wire Interrupt Masking Register

Offset: 0x14			Register Name: OW_INT_MASK
Default Value: 0x00000000			
Bit	Read/Write	Default	Description
31:6	/	/	/
5	R/W	0	<p><b>Deglitch Detected Interrupt Enable</b></p> <p>0 = Disable</p> <p>1 = Enable</p>
4	R/W	0	<p><b>CRC Comparing Complete Interrupt Enable</b></p> <p>0 = Disable</p> <p>1 = Enable</p>
3	R/W	0	<p><b>Transmission Complete Interrupt Enable</b></p> <p>0 = Disable</p> <p>1 = Enable</p>
2	R/W	0	<p><b>Read Complete Interrupt Enable</b></p> <p>0 = Disable</p> <p>1 = Enable</p>
1	R/W	0	<p><b>Time-out Interrupt Enable</b></p> <p>0 = Disable</p> <p>1 = Enable</p>
			<p><b>Presence Detect Interrupt Enable</b></p> <p>0 = Disable</p>



0	R/W	0	1 = Enable
---	-----	---	------------

#### 8.6.3.7. HDQ/One Wire Function Clock Register

<b>Offset: 0x18</b>			<b>Register Name: OW_FCLK</b> <b>Default Value: 0x00000000</b>
Bit	Read/Write	Default	Description
31:21	/	/	/
20:16	R/W	0	OW_FCLK (n) A n MHz clock is needed to use as a time reference by the machine. Transitions between the states of the state machine as well as actions triggered at precise time deadlines are expressed using the n – MHz clock.
15:8	/	/	/
7:0	R/W	0	OW_FCLK_D OW_FCLK = SOURCE_CLK/OW_FCLK_D

#### 8.6.3.8. One Wire Line Control Register

<b>Offset: 0x1c</b>			<b>Register Name: OW_LC</b> <b>Default Value: 0x00000000</b>
Bit	Read/Write	Default	Description
31:3	/	/	/
2	R	1	<b>Current state of One Wire Line</b> 0: low 1: high
1	R/W	0	<b>One Wire line state control bit</b> When the line control mode is enabled (bit [0] set), value of this bit decides the output level of the One Wire line. 0: output low level

			1: output high level
0	R/W	0	<b>One Wire line state control enable</b> When this bit is set, the state of One Wire line is control by the value of bit [1]. 0: disable line control mode 1: enable line control mode

### 8.6.3.9. Standard Mode Write Read Timing Control Register

Offset: 0x20			Register Name: OW_SMSC
Default Value: 0x213de0bc			
Bit	Read/Write	Default	Description
31	/	/	/
30:29	R/W	0x1	TSU Read Data Setup. In standard speed, range: $t(SU) < 1$ 00: 0.5us 01: 1us 10: 2us 11: 4us
28	/	/	/
27:24	R/W	1	REC Recovery Time, $t(\text{recovery}) = N$ us. In standard speed, range: $1\text{us} \leq t(\text{recovery})$
23	/	/	/
22:18	R/W	0xf	TRDV Read data valid time, $t(\text{rdv}) = N$ us. In standard speed, range: Exactly 15
17:11	R/W	0x3c	TLOW0 Write Zero time Low, $Tlow0 = N$ us. The range setting for TLOW0 is from 0x3c to 0x77. In standard mode, range: $60 \leq t(\text{low0}) < t(\text{tslot}) < 120$
			TLOW1

10:7	R/W	1	Write One time Low, or TLOWR both are same. $t(\text{low1}) = N$ us. The range setting for TLOW1 and TLOWR here is from 0x1 to 0xf. In standard speed, range: $1 \leq t(\text{low1}) < 15$ . $t(\text{lowR}) = N$ ovr clks. In standard speed, rang = $1 \leq t(\text{lowR}) < 15$
6:0	R/W	0x3c	TSLOT Active time slot for write and read data, $t(\text{slot}) = N$ us. The range setting for TSLOT is from 0x3c to 0x78. In standard mode, range: $60 \leq t(\text{slot}) < 120$

#### 8.6.3.10. Standard Mode Reset Presence Timing Control Register

Offset: 0x24			Register Name: SM_RST_PRESENCE_TCTL Default Value: 0x3c3fc1e0
Bit	Read/Write	Default	Description
31:24	R/W	0x3c	TPDL PRESENCE_DETECT_LOW $t(\text{pdl}) = N$ us. The range setting for TPDL in these fields is from 0 to 0xf0. In standard speed, Range: $60 \leq t(\text{pdl}) < 240$ .
23:18	R/W	0xf	TPDH PRESENCE_DETECT_HIGH $t(\text{pdh}) = N$ us. The range setting for TPDH in these fields is from 0xf to 0x3c. In standard speed, range: $15 \leq t(\text{pdh}) < 60$ .
17:9	R/W	0x1e0	TRSTL RESET_TIME_LOW $t(\text{rstl}) = N$ us. The range setting for TRSTL in these fields is from 0 to 0xff. In standard speed, Range: $480 \leq t(\text{rstl}) < \text{infinity}$
8:0	R/W	0x1e0	TRSTH RESET_TIME_HIGH, $t(\text{rsth}) = N$ us. The range setting for TRSTH in these fields is from 0 to 0xff. In standard speed, Range : $480 \leq t(\text{rsth}) < \text{infinity}$

### 8.6.3.11. Simple Mode Write Read Timing Control Register

Offset: 0x28			Register Name: SP_WR_RD_TCTL			
Default Value: 0x0a0158be						
Bit	Read/Write	Default	Description			
31:28	R/W	0	RD_SAMPLE_POINT			
			When controller of the Simple Mode read, the default sample point is at the middle of the THW1 point and the THW0 point, named S(middle). When these fields are set, the corresponding new sample point will be determined.			
			0000	S(middle)	1000	S(middle)-30us
			0001	S(middle)+5us	1001	S(middle)+40us
			0010	S(middle)-5us	1010	S(middle)-40us
			0011	S(middle)+10us	1011	S(middle)+50us
			0100	S(middle)-10us	1100	S(middle)-50us
			0101	S(middle)+20us	1101	S(middle)+60us
			0110	S(middle)-20us	1110	S(middle)-60us
0111	S(middle)+30us	1111	reserve			
27:22	R/W	0x28	THW1_INT $t(\text{HW1\_INT}) = N \text{ us}$ . The range setting for THW1_INT in these fields is from 0 to 0x3f, which is the integer part of the THW1. In HDQ mode, Range: $t(\text{HW0}) \leq 50 \text{ us}$ .			
			THW1_DEC THW1_DEC is the decimal part of the THW1. $t(\text{HW1\_DEC}) = N \text{ ow\_clks}$ . The value for the THW1 = THW1_INT + THW1_DEC.			

21:18	R/W	0	
17:10	R/W	0x56	THW0 t(HW0) = N us. The range setting for THW0 in these fields is from 0 to 0xff. In HDQ mode, Range: t(HW0) <= 145 us.
9:0	R/W	0xbe	TCYCH t(CYCH) = N us. The range setting for TCYCH in these fields is from 0 to 0x3ff. In HDQ mode, Rang: 190 us <= t(CYCH) <= infinity.

### 8.6.3.12. Simple Mode Break Timing Control Register

<b>Offset: 0x2c</b>			<b>Register Name: HDQ_BR_TCTL</b> <b>Default Value: 0x0be0028</b>
Bit	Read/Write	Default	Description
31:26	/	/	/
25:16	R/W	0xbe	TB t(B) = N us. The range setting for TB in these fields is from 0 to 0x 3ff. In HDQ mode, Rang: 190 us <= t(B) <= infinity.
15:10	/	/	/
9:0	R/W	0x28	TBR t(BR) = N us. The range setting for TBR in these fields is from 0 to 0xff. In HDQ mode, Rang: 40 us <= t(BR) <= infinity.

## 8.7. CIR Receiver

### 8.7.1. Overview

The CIR includes the following features:

- Full physical layer implementation
- Support CIR for remote control
- 64x8 bits FIFO for data buffer
- Programmable FIFO thresholds

For saving CPU resource, CIR receiver is implemented in hardware. The CIR receiver samples the input signal on the programmable frequency and records these samples into RX FIFO when one CIR signal is found on the air. The CIR receiver uses Run-Length Code (RLC) to encode pulse width. The encoded data is buffered in a 64 levels and 8-bit width RX FIFO; the MSB bit is used to record the polarity of the receiving CIR signal. The high level is represented as '1' and the low level is represented as '0'. The rest 7 bits are used for the length of RLC. The maximum length is 128. If the duration of one level (high or low level) is more than 128, another byte is used.

In the air, there is always some noise. One threshold can be set to filter the noise to reduce system loading and improve the system stability.

### 8.7.2. CIR Receiver Register List

Module Name	Base Address
CIR0	0x08002000

Register Name	Offset	Description
CIR_CTL	0x00	CIR Control Register
CIR_RXCTL	0x10	CIR Receiver Configure Register
CIR_RXFIFO	0x20	CIR Receiver FIFO Register
CIR_RXINT	0x2C	CIR Receiver Interrupt Control Register
CIR_RXSTA	0x30	CIR Receiver Status Register
CIR_CONFIG	0x34	CIR Configure Register

### 8.7.3. CIR Receiver Register Description

#### 8.7.3.1. CIR Receiver Control Register

Offset: 0x00			Register Name: CIR_CTL
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:9	/	/	/
8	R/W	0	CGPO General Program Output (GPO) Control in CIR mode for TX Pin 0: Low level 1: High level
7:6	/	/	/
5:4	R/W	0	CIR ENABLE 00~10: Reserved 11: CIR mode enable
3:2	/	/	/
1	R/W	0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0	GEN Global Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disable 1: Enable



### 8.7.3.2. CIR Receiver Configure Register

<b>Offset: 0x10</b>			<b>Register Name: IR_RXCTL</b>
			<b>Default Value: 0x0000_0004</b>
Bit	Read/Write	Default	Description
31:3	/	/	/
2	R/W	1	RPPI Receiver Pulse Polarity Invert  0: Not invert receiver signal 1: Invert receiver signal
1:0	/	/	/

### 8.7.3.3. CIR Receiver FIFO Register

<b>Offset: 0x20</b>			<b>Register Name: IR_RXFIFO</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/
7:0	R	0	Receiver Byte FIFO

### 8.7.3.4. CIR Receiver Interrupt Control Register

<b>Offset: 0x2C</b>			<b>Register Name: IR_RXINT</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:14	/	/	/
13:8	R/W	0	RAL RX FIFO Available Received Byte Level for interrupt and DMA request  TRIGGER_LEVEL = RAL + 1
5	R/W	0	DRQ_EN RX FIFO DMA Enable

			0: Disable  1: Enable  When set to '1', the Receiver FIFO DRQ is asserted if reaching RAL. The DRQ is de-asserted when condition fails.
4	R/W	0	RAI_EN  RX FIFO Available Interrupt Enable  0: Disable  1: Enable  When set to '1', the Receiver FIFO IRQ is asserted if reaching RAL. The IRQ is de-asserted when condition fails.
3:2	/	/	/
1	R/W	0	RPEI_EN  Receiver Packet End Interrupt Enable  0: Disable  1: Enable
0	R/W	0	ROI_EN  Receiver FIFO Overrun Interrupt Enable  0: Disable  1: Enable

#### 8.7.3.5. CIR Receiver Status Register

<b>Offset: 0x30</b>			<b>Register Name: IR_RXSTA</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:15	/	/	/
14:8	R	0	RAC  RX FIFO Available Counter

			<p>0: No available data in RX FIFO</p> <p>1: 1 byte available data in RX FIFO</p> <p>2: 2 byte available data in RX FIFO</p> <p>...</p> <p>64: 64 byte available data in RX FIFO</p>
7	R	0x0	<p>STAT</p> <p>Status of CIR</p> <p>0x0 – Idle</p> <p>0x1 – busy</p>
6:5	/	/	/
4	R/W	0	<p>RA</p> <p>RX FIFO Available</p> <p>0: RX FIFO not available according its level</p> <p>1: RX FIFO available according its level</p> <p>This bit is cleared by writing a '1'.</p>
3:2	/	/	/
1	R/W	0	<p>RPE</p> <p>Receiver Packet End Flag</p> <p>0: STO was not detected. In CIR mode, one CIR symbol is receiving or not detected.</p> <p>1: STO field or packet abort symbol (7'b0000,000 and 8'b0000,0000 for MIR and FIR) is detected. In CIR mode, one CIR symbol is received.</p> <p>This bit is cleared by writing a '1'.</p>
0	R/W	0	<p>ROI</p> <p>Receiver FIFO Overrun</p> <p>0: Receiver FIFO not overrun</p>

			1: Receiver FIFO overrun  This bit is cleared by writing a '1'.
--	--	--	---

### 8.7.3.6. CIR Receiver Configure Register

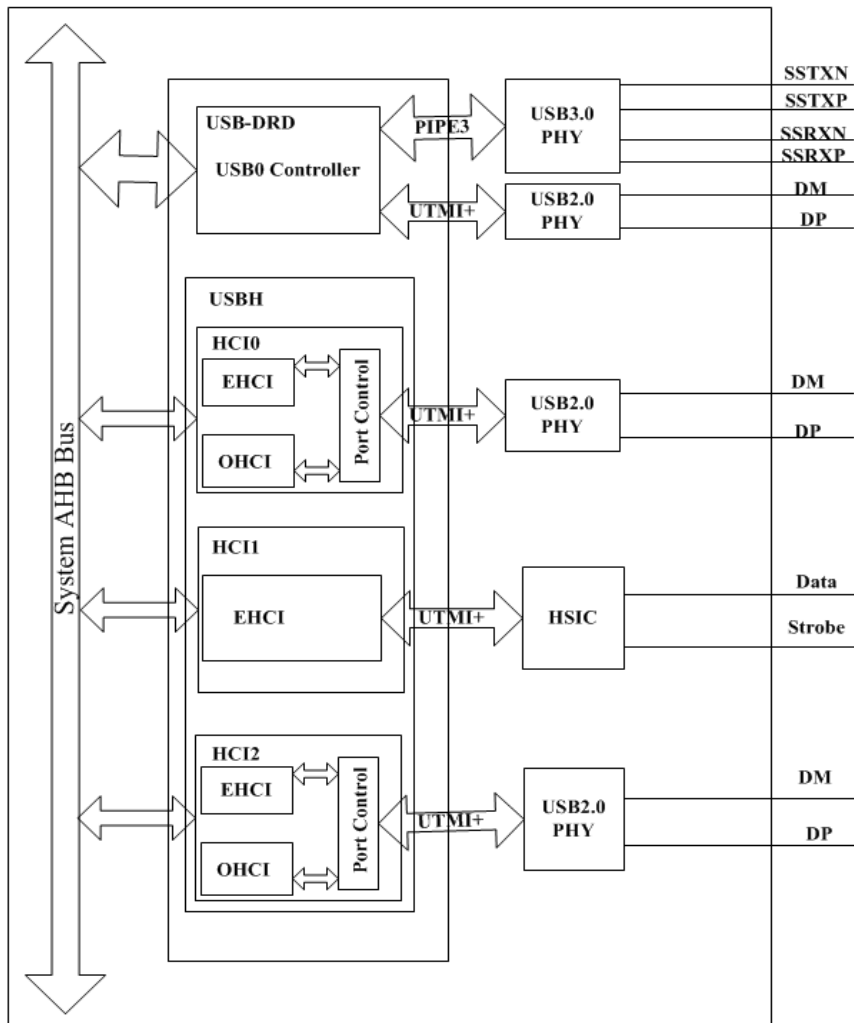
Offset: 0x34			Register Name: IR_CIR
Default Value: 0x0000_1828			
Bit	Read/Write	Default	Description
31	/	/	/
30:25	/	/	/
24	R/W	0x0	SCS2  Bit2 of Sample Clock Select for CIR  This bit is defined by SCS bits below.
23	R/W	0x0	ATHC  Active Threshold Control for CIR  0x0 –ATHR in Unit of (Sample Clock) 0x1 –ATHR in Unit of (128*Sample Clocks)
22:16	R/W	0x0	ATHR  Active Threshold for CIR  These bits control the duration of CIR from Idle to Active State. The duration can be calculated by ((ATHR + 1)*(ATHC? Sample Clock: 128*Sample Clock)).
15:8	R/W	0x18	ITHR  Idle Threshold for CIR  The Receiver uses it to decide whether the CIR command has been received. If there is no CIR signal on the air, the receiver is staying in IDLE status. One active pulse will bring the receiver from IDLE status to Receiving status. After the CIR is end, the inputting signal will keep the specified level (high or low level) for a long time. The receiver can use this idle signal duration to decide that it has received the CIR command. The corresponding flag is asserted. If the corresponding interrupt is

			<p>enable, the interrupt line is asserted to CPU.</p> <p>When the duration of signal keeps one status (high or low level) for the specified duration ( <math>(ITHR + 1) * 128 \text{ sample\_clk}</math> ), this means that the previous CIR command has been finished.</p>																																				
7:2	R/W	0xa	<p><b>NTHR</b></p> <p>Noise Threshold for CIR</p> <p>When the duration of signal pulse (high or low level) is less than NTHR, the pulse is taken as noise and should be discarded by hardware.</p> <p>0: all samples are recorded into RX FIFO</p> <p>1: If the signal is only one sample duration, it is taken as noise and discarded.</p> <p>2: If the signal is less than (<math>\leq</math>) two sample duration, it is taken as noise and discarded.</p> <p>...</p> <p>61: if the signal is less than (<math>\leq</math>) sixty-one sample duration, it is taken as noise and discarded.</p>																																				
1:0	R/W	0	<p><b>SCS</b></p> <p>Sample Clock Select for CIR</p> <table border="1"> <thead> <tr> <th>SCS2</th> <th>SCS[1]</th> <th>SCS[0]</th> <th>Sample Clock</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>ir_clk/64</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>ir_clk/128</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>ir_clk/256</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>ir_clk/512</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>ir_clk</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>Reserved</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>Reserved</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>Reserved</td> </tr> </tbody> </table>	SCS2	SCS[1]	SCS[0]	Sample Clock	0	0	0	ir_clk/64	0	0	1	ir_clk/128	0	1	0	ir_clk/256	0	1	1	ir_clk/512	1	0	0	ir_clk	1	0	1	Reserved	1	1	0	Reserved	1	1	1	Reserved
SCS2	SCS[1]	SCS[0]	Sample Clock																																				
0	0	0	ir_clk/64																																				
0	0	1	ir_clk/128																																				
0	1	0	ir_clk/256																																				
0	1	1	ir_clk/512																																				
1	0	0	ir_clk																																				
1	0	1	Reserved																																				
1	1	0	Reserved																																				
1	1	1	Reserved																																				

## 8.8. USB

### 8.8.1. Block Diagram

The USB block diagram is shown below:



### 8.8.2. USB DRD Controller

The USB3.0 controller has following features:

- Support Device or Host operation at a time
- Support Super-Speed(SS, 5-Gbps), High-Speed(HS, 480-Mbps), Full-Speed(FS, 12-Mbps) in Device mode
- Support Super-Speed(SS, 5-Gbps), High-Speed(HS, 480-Mbps), Full-Speed(FS, 12-Mbps) and Low-Speed(LS, 1.5-Mbps) in Host mode
- USB 3.0 PIPE3 PHY interface
- USB 2.0 UTMI+ (L3) PHY interface
- Support up to 5 bidirectional endpoints, including control endpoint 0 in Device mode
- Simultaneous IN and OUT transfer support in superspeed mode
- Dual-port interfaces for Tx data buffering, Rx data prefetching, and descriptor and register caching
- Three RAMs are configuration as Rx Data FIFO RAM, Tx Data FIFO RAM, Descriptor/Register Cache RAM
- Hardware handles all data transfer
- Support ACA for BC detect
- Implements both static and dynamic power reduction techniques at multiple levels

### 8.8.3. USB Host Controller

#### 8.8.3.1. Overview

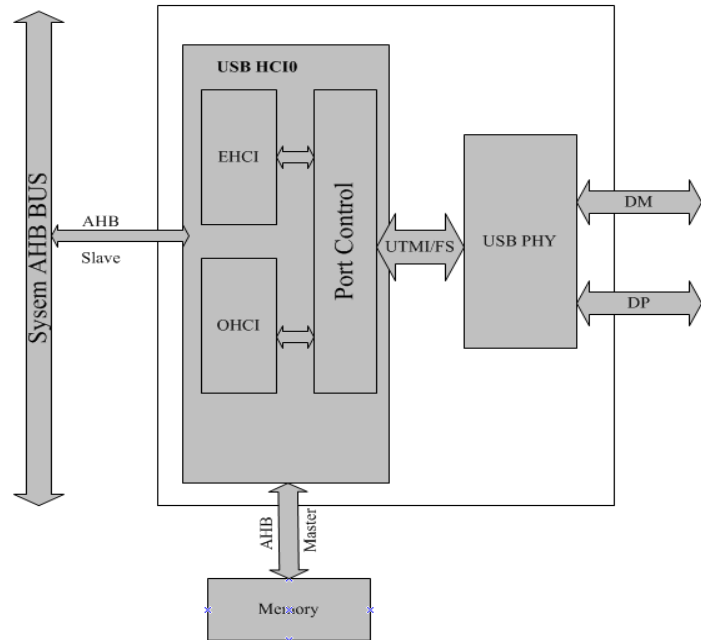
USB Host Controller is fully compliant with the USB 2.0 specification, Enhanced Host Controller Interface (EHCI) Specification, Revision 1.0, and the Open Host Controller Interface (OHCI) Specification Release 1.0a. The controller supports high-speed, 480-Mbps transfers (40 times faster than USB 1.1 full-speed mode) using an EHCI Host Controller, as well as full and low speeds through one or more integrated OHCI Host Controllers.

The USB host controller includes the following features:

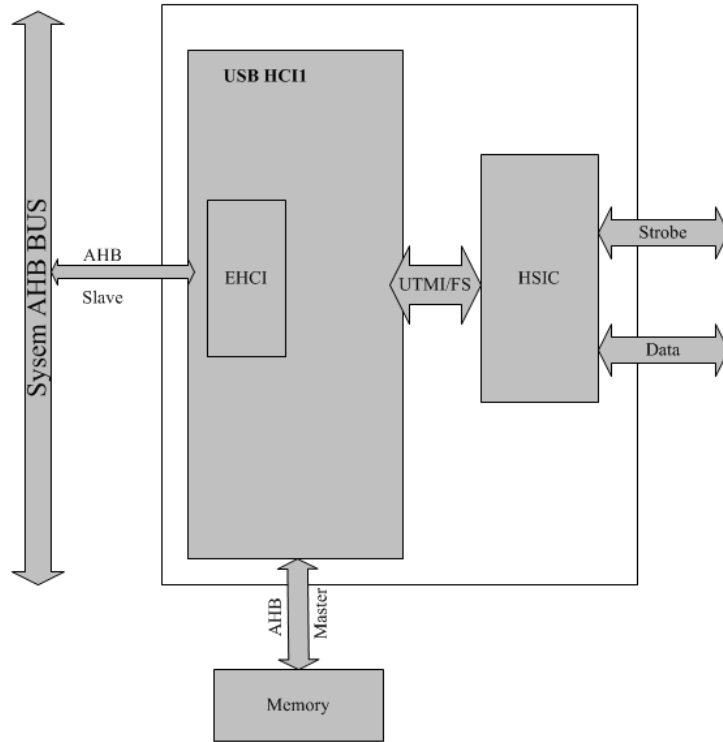
- Supports industry-standard AMBA High-Performance Bus (AHB) and it is fully compliant with the AMBA Specification, Revision 2.0. Supports bus.
- Supports 32-bit Little Endian AMBA AHB Slave Bus for Register Access.
- Supports 32-bit Little Endian AMBA AHB Master Bus for Memory Access.
- Including an internal DMA Controller for data transfer with memory.
- Complies with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0, and the Open Host Controller Interface (OHCI) Specification, Version 1.0a.
- Supports High-Speed (HS, 480-Mbps), Full-Speed (FS, 12-Mbps), and Low-Speed (LS, 1.5-Mbps) Device.
- Supports the UTMI+ Level 3 interface . The 8-bit bidirectional data buses are used.
- Supports only 1 USB Root Port shared between EHCI and OHCI.
- HCI0 support High-Speed(HS, 480-Mbps), Full-Speed(FS, 12-Mbps), and Low-Speed(LS, 1.5-Mbps) USB Device through standard USB difference port
- HCI1 support only High-Speed(HS, 480-Mbps) mode through HSIC port
- HCI2 support High-Speed(HS, 480-Mbps), Full-Speed(FS, 12-Mbps), and Low-Speed(LS, 1.5-Mbps) USB Device through standard USB difference port
- HCI2 support only High-Speed(HS, 480-Mbps) mode through HSIC port



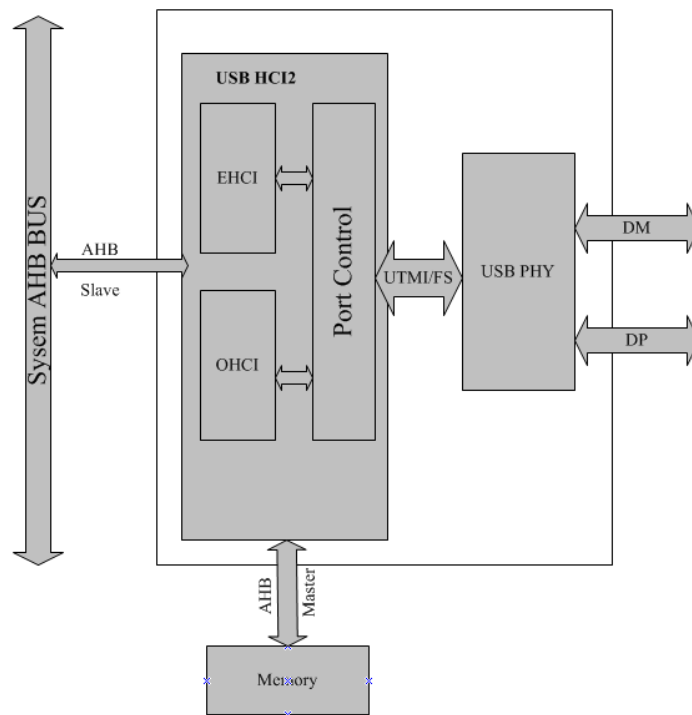
**8.8.3.2. Block Diagram**



USB host controller System-Level block diagram



HCI1 Block Diagram



HCI2 Block Diagram

### 8.8.3.3. USB Host Timing Diagram

Please refer USB2.0 Specification, Enhanced Host Controller Interface (EHCI) Specification, Version 1.0, and the Open Host Controller Interface (OHCI) Specification, Version 1.0a.

### 8.8.3.4. USB Host Register List

Module Name	Base Address
USB_HCI0	0x00A00000
USB_HCI1	0x00A01000
USB_HCI2	0x00A02000

Register Name	Offset	Description
EHCI Capability Register		
E_CAPLENGTH	0x000	EHCI Capability register Length Register
E_HCVERSION	0x002	EHCI Host Interface Version Number Register
E_HCSPARAMS	0x004	EHCI Host Control Structural Parameter Register
E_HCCPARAMS	0x008	EHCI Host Control Capability Parameter Register
E_HCSPPORTROUTE	0x00c	EHCI Companion Port Route Description
EHCI Operational Register		
E_USBCMD	0x010	EHCI USB Command Register
E_USBSTS	0x014	EHCI USB Status Register
E_USBINTR	0x018	EHCI USB Interrupt Enable Register
E_FRINDEX	0x01c	EHCI USB Frame Index Register
E_CTRLDSSEGMENT	0x020	EHCI 4G Segment Selector Register
E_PERIODICLISTBASE	0x024	EHCI Frame List Base Address Register
E_ASYNC_LISTADDR	0x028	EHCI Next Asynchronous List Address Register
E_CONFIGFLAG	0x050	EHCI Configured Flag Register
E_PORTSC	0x054	EHCI Port Status/Control Register
OHCI Control and Status Partition Register		
O_HcRevision	0x400	OHCI Revision Register
O_HcControl	0x404	OHCI Control Register
O_HcCommandStatus	0x408	OHCI Command Status Register
O_HcInterruptStatus	0x40c	OHCI Interrupt Status Register
O_HcInterruptEnable	0x410	OHCI Interrupt Enable Register
O_HcInterruptDisable	0x414	OHCI Interrupt Disable Register
OHCI Memory Pointer Partition Register		
O_HcHCCA	0x418	OHCI HCCA Base
O_HcPeriodCurrentED	0x41c	OHCI Period Current ED Base

O_HcControlHeadED	0x420	OHCI Control Head ED Base
O_HcControlCurrentED	0x424	OHCI Control Current ED Base
O_HcBulkHeadED	0x428	OHCI Bulk Head ED Base
O_HcBulkCurrentED	0x42c	OHCI Bulk Current ED Base
O_HcDoneHead	0x430	OHCI Done Head Base
OHCI Frame Counter Partition Register		
O_HcFmInterval	0x434	OHCI Frame Interval Register
O_HcFmRemaining	0x438	OHCI Frame Remaining Register
O_HcFmNumber	0x43c	OHCI Frame Number Register
O_HcPeriodicStart	0x440	OHCI Periodic Start Register
O_HcLSThreshold	0x444	OHCI LS Threshold Register
OHCI Root Hub Partition Register		
O_HcRhDescriptorA	0x448	OHCI Root Hub Descriptor Register A
O_HcRhDescriptorB	0x44c	OHCI Root Hub Descriptor Register B
O_HcRhStatus	0x450	OHCI Root Hub Status Register
O_HcRhPortStatus	0x454	OHCI Root Hub Port Status Register

### 8.8.3.5. EHCI Register Description

#### 8.8.3.5.1. EHCI Identification Register

<b>Offset:0x00</b>			<b>Register Name: CAPLENGTH</b>
<b>Bit</b>			<b>Default Value: Implementation Dependent</b>
<b>Read/Write</b>	<b>Default</b>	<b>Description</b>	
7:0	R	0x10	<b>CAPLENGTH</b> The value in these bits indicates an offset to add to register base to find the beginning of the Operational Register Space.

#### 8.8.3.5.2. EHCI Host Interface Version Number Register

<b>Offset: 0x02</b>			<b>Register Name: HCIVERSION</b>
<b>Bit</b>			<b>Default Value:0x0100</b>
<b>Read/Write</b>	<b>Default</b>	<b>Description</b>	
15:0	R	0x0100	<b>HCIVERSION</b>

			This is a 16-bits register containing a BCD encoding of the EHCI revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision.
--	--	--	--

### 8.8.3.5.3. EHCI Host Control Structural Parameter Register

Offset: 0x04			Register Name: HCSPARAMS
			Default Value: Implementation Dependent
Bit	Read/Write	Default	Description
31:24	/	0	<b>Reserved.</b> These bits are reserved and should be set to zero.
23:20	R	0	<b>Debug Port Number</b> This register identifies which of the host controller ports is the debug port. The value is the port number (one based) of the debug port. This field will always be '0'.
19:16	/	0	<b>Reserved.</b> These bits are reserved and should be set to zero.
15:12	R	0	<b>Number of Companion Controller (N_CC)</b> This field indicates the number of companion controllers associated with this USB2.0 host controller. A zero in this field indicates there are no companion host controllers. And a value larger than zero in this field indicates there are companion USB1.1 host controller(s). This field will always be '0'.
11:8	R	0	<b>Number of Port per Companion Controller(N_PCC)</b> This field indicates the number of ports supported per companion host controller host controller. It is used to indicate the port routing configuration to system software. This field will always fix with '0'.
7	R	0	<b>Port Routing Rules</b> This field indicates the method used by this implementation for how all ports are mapped to companion controllers. The value of this field has the following interpretation:

			<table border="1"> <tr> <th>Value</th> <th>Meaning</th> </tr> <tr> <td>0</td> <td>The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.</td> </tr> <tr> <td>1</td> <td>The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.</td> </tr> </table> <p>This field will always be '0'.</p>	Value	Meaning	0	The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.	1	The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.
Value	Meaning								
0	The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.								
1	The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.								
6:4	/	0	<p><b>Reserved.</b></p> <p>These bits are reserved and should be set to zero.</p>						
3:0	R	1	<p><b>N_PORTS</b></p> <p>This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 0x1 to 0x0f.</p> <p>This field is always 1.</p>						

#### 8.8.3.5.4. EHCI Host Control Capability Parameter Register

<b>Offset: 0x08</b>			<p><b>Register Name: HCCPARAMS</b></p> <p><b>Default Value: Implementation Dependent</b></p>
Bit	Read/Write	Default	Description
31:16	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>
15:18	R	0	<p><b>EHCI Extended Capabilities Pointer (EECP)</b></p> <p>This optional field indicates the existence of a capabilities list. A value of 00b indicates no extended capabilities are implemented. A non-zero value in this register indicates the offset in PCI configuration space of the first EHCI extended capability. The pointer value must be 40h or greater if implemented to maintain consistency of the PCI header defined for this class of device.</p> <p>The value of this field is always '00b'.</p>
7:4	R		<p><b>Isochronous Scheduling Threshold</b></p> <p>This field indicates, relative to the current position of the executing host</p>

			<p>controller, where software can reliably update the isochronous schedule.</p> <p>When bit[7] is zero, the value of the least significant 3 bits indicates the number of micro-frames a host controller can hold a set of isochronous data structures(one or more) before flushing the state. When bit[7] is a one, then host software assumes the host controller may cache an isochronous data structure for an entire frame.</p>
3	R	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>
2	R		<p><b>Asynchronous Schedule Park Capability</b></p> <p>If this bit is set to a one, then the host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule. The feature can be disabled or enabled and set to a specific level by using the Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count fields in the USBCMD register.</p>
1	R		<p><b>Programmable Frame List Flag</b></p> <p>If this bit is set to a zero, then system software must use a frame list length of 1024 elements with this host controller.The USBCMD register Frame List Size field is a read-only register and should be set to zero.</p> <p>If set to 1,then system software can specify and use the frame list in the USBCMD register Frame List Size field to cofigure the host controller.</p> <p>The frame list must always aligned on a 4K page boundary.This requirement ensures that the frame list is always physically contiguous.</p>
0	R	0	<p><b>Reserved</b></p> <p>These bits are reserved for future use and should return a value of zero when read.</p>

#### 8.8.3.5.5. EHCI Companion Port Route Description

<b>Offset: 0x0C</b>			<p><b>Register Name: HCSP-PORTROUTE</b></p> <p><b>Default Value: UNDEFINED</b></p>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>

31:0	R	<p><b>HCSP-PORTROUTE</b></p> <p>This optional field is valid only if Port Routing Rules field in HCSPARAMS register is set to a one.</p> <p>This field is used to allow a host controller implementation to explicitly describe to which companion host controller each implemented port is mapped. This field is a 15-element nibble array (each 4 bit is one array element). Each array location corresponds one-to-one with a physical port provided by the host controller (e.g. PORTROUTE [0] corresponds to the first PORTSC port, PORTROUTE [1] to the second PORTSC port, etc.). The value of each element indicates to which of the companion host controllers this port is routed. Only the first N_PORTS elements have valid information. A value of zero indicates that the port is routed to the lowest numbered function companion host controller. A value of one indicates that the port is routed to the next lowest numbered function companion host controller, and so on.</p>
------	---	---

#### 8.8.3.5.6. EHCI USB Command Register

Offset: 0x10			<p><b>Register Name: USBCMD</b></p> <p><b>Default Value: 0x00080000(0x00080B00 if Asynchronous Schedule Park Capability is a one)</b></p>														
Bit	Read/Write	Default	Description														
31:24	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>														
23:16	R/W	0x08	<p><b>Interrupt Threshold Control</b></p> <p>The value in this field is used by system software to select the maximum rate at which the host controller will issue interrupts. The only valid values are defined below:</p> <table border="1" data-bbox="560 1783 1385 2092"> <thead> <tr> <th>Value</th> <th>Minimum Interrupt Interval</th> </tr> </thead> <tbody> <tr> <td>0x00</td> <td>Reserved</td> </tr> <tr> <td>0x01</td> <td>1 micro-frame</td> </tr> <tr> <td>0x02</td> <td>2 micro-frame</td> </tr> <tr> <td>0x04</td> <td>4 micro-frame</td> </tr> <tr> <td>0x08</td> <td>8 micro-frame(default, equates to 1 ms)</td> </tr> <tr> <td>0x10</td> <td>16 micro-frame(2ms)</td> </tr> </tbody> </table>	Value	Minimum Interrupt Interval	0x00	Reserved	0x01	1 micro-frame	0x02	2 micro-frame	0x04	4 micro-frame	0x08	8 micro-frame(default, equates to 1 ms)	0x10	16 micro-frame(2ms)
Value	Minimum Interrupt Interval																
0x00	Reserved																
0x01	1 micro-frame																
0x02	2 micro-frame																
0x04	4 micro-frame																
0x08	8 micro-frame(default, equates to 1 ms)																
0x10	16 micro-frame(2ms)																



			<table border="1"> <tr> <td>0x20</td> <td>32 micro-frame(4ms)</td> </tr> <tr> <td>0x40</td> <td>64 micro-frame(8ms)</td> </tr> </table> <p>Any other value in this register yields undefined results.</p> <p>The default value in this field is 0x08 .</p> <p>Software modifications to this bit while HC Halted bit is equal to zero results in undefined behavior.</p>	0x20	32 micro-frame(4ms)	0x40	64 micro-frame(8ms)
0x20	32 micro-frame(4ms)						
0x40	64 micro-frame(8ms)						
15:12	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>				
11	R/W or R	0	<p><b>Asynchronous Schedule Park Mode Enable(OPTIONAL)</b></p> <p>If the Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this bit defaults to a 1 and is R/W. Otherwise the bit must be a zero and is Read Only. Software uses this bit to enable or disable Park mode. When this bit is one, Park mode is enabled. When this bit is zero, Park mode is disabled.</p>				
10	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>				
9:8	R/W or R	0	<p><b>Asynchronous Schedule Park Mode Count(OPTIONAL)</b></p> <p>Asynchronous Park Capability bit in the HCCPARAMS register is a one,</p> <p>Then this field defaults to 0x3 and is W/R. Otherwise it defaults to zero and is R. It contains a count of the number of successive transactions the host controller is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule.</p> <p>Valid value are 0x1 to 0x3. Software must not write a zero to this bit when Park Mode Enable is a one as it will result in undefined behavior.</p>				
7	R/W	0	<p><b>Light Host Controller Reset(OPTIONAL)</b></p> <p>This control bit is not required.</p> <p>If implemented, it allows the driver to reset the EHCI controller without affecting the state of the ports or relationship to the companion host controllers. For example, the PORSTC registers should not be reset to their default values and the CF bit setting should not go to zero (retaining port ownership relationships).</p> <p>A host software read of this bit as zero indicates the Light Host Controller Reset has completed and it is safe for software to re-initialize</p>				

			the host controller. A host software read of this bit as a one indicates the Light Host						
6	R/W	0	<p><b>Interrupt on Async Advance Doorbell</b></p> <p>This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Soft-</p> <p>Ware must write a 1 to this bit to ring the doorbell.</p> <p>When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS. if the Interrupt on Async Advance Enable bit in the USBINTR register is a one then the host controller will assert an interrupt at the next interrupt threshold.</p> <p>The host controller sets this bit to a zero after it has set the Interrupt on Async Advance status bit in the USBSTS register to a one.</p> <p>Software should not write a one to this bit when the asynchronous schedule is disabled. Doing so will yield undefined results.</p>						
5	R/W	0	<p><b>Asynchronous Schedule Enable</b></p> <p>This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean:</p> <table border="1" data-bbox="560 1308 1382 1525"> <thead> <tr> <th>Bit Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Do not process the Asynchronous Schedule.</td> </tr> <tr> <td>1</td> <td>Use the ASYNLISTADDR register to access the Asynchronous Schedule.</td> </tr> </tbody> </table> <p>The default value of this field is '0b'.</p>	Bit Value	Meaning	0	Do not process the Asynchronous Schedule.	1	Use the ASYNLISTADDR register to access the Asynchronous Schedule.
Bit Value	Meaning								
0	Do not process the Asynchronous Schedule.								
1	Use the ASYNLISTADDR register to access the Asynchronous Schedule.								
4	R/W	0	<p><b>Periodic Schedule Enable</b></p> <p>This bit controls whether the host controller skips processing the Periodic Schedule. Values mean:</p> <table border="1" data-bbox="560 1767 1382 1984"> <thead> <tr> <th>Bit Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Do not process the Periodic Schedule.</td> </tr> <tr> <td>1</td> <td>Use the PERIODICLISTBASE register to access the Periodic Schedule.</td> </tr> </tbody> </table> <p>The default value of this field is '0b'.</p>	Bit Value	Meaning	0	Do not process the Periodic Schedule.	1	Use the PERIODICLISTBASE register to access the Periodic Schedule.
Bit Value	Meaning								
0	Do not process the Periodic Schedule.								
1	Use the PERIODICLISTBASE register to access the Periodic Schedule.								

3:2	R/W or R	0	<p><b>Frame List Size</b></p> <p>This field is R/W only if Programmable Frame List Flag in the HCCPARAMS registers is set to a one. This field specifies the size of the Frame list. The size the frame list controls which bits in the Frame Index Register should be used for the Frame List Current index. Values mean:</p> <table border="1" data-bbox="560 611 1380 835"> <thead> <tr> <th>Bits</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>1024 elements(4096bytes)Default value</td> </tr> <tr> <td>01b</td> <td>512 elements(2048byts)</td> </tr> <tr> <td>10b</td> <td>256 elements(1024bytes)For resource-constrained condition</td> </tr> <tr> <td>11b</td> <td>reserved</td> </tr> </tbody> </table> <p>The default value is '00b'.</p>	Bits	Meaning	00b	1024 elements(4096bytes)Default value	01b	512 elements(2048byts)	10b	256 elements(1024bytes)For resource-constrained condition	11b	reserved
Bits	Meaning												
00b	1024 elements(4096bytes)Default value												
01b	512 elements(2048byts)												
10b	256 elements(1024bytes)For resource-constrained condition												
11b	reserved												
1	R/W	0	<p><b>Host Controller Reset</b></p> <p>This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset.</p> <p>When software writes a one to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports.</p> <p>All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s). Software must reinitialize the host controller as described in Section 4.1 of the CHEI Specification in order to return the host controller to an operational state.</p> <p>This bit is set to zero by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a zero to this register.</p> <p>Software should not set this bit to a one when the HC Halted bit in the USBSTS register is a zero. Attempting to reset an actively running host controller will result in undefined behavior.</p>										
0	R/W	0	<p><b>Run/Stop</b></p> <p>When set to a 1, the Host Controller proceeds with execution of the schedule. When set to 0, the Host Controller completes the current and any actively pipelined transactions on the USB and then halts. The Host</p>										

			<p>Controller must halt within 16 micro-frames after software clears this bit.</p> <p>The HC Halted bit indicates when the Host Controller has finished its pending pipelined transactions and has entered the stopped state.</p> <p>Software must not write a one to this field unless the Host Controller is in the Halt State.</p> <p>The default value is 0x0.</p>
--	--	--	--

### 8.8.3.5.7. EHCI USB Status Register

Offset: 0x14			Register Name: USBSTS
Bit			Default Value: 0x00001000
Bit	Read/Write	Default	Description
31:16	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>
15	R	0	<p><b>Asynchronous Schedule Status</b></p> <p>The bit reports the current real status of Asynchronous Schedule. If this bit is a zero then the status of the Asynchronous Schedule is disabled. If this bit is a one then the status of the Asynchronous Schedule is enabled. The Host Controller is not required to immediately disable or enable the Asynchronous Schedule when software transitions the Asynchronous Schedule Enable bit in the USBCMD register. When this bit and the Asynchronous Schedule Enable bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).</p>
14	R	0	<p><b>Periodic Schedule Status</b></p> <p>The bit reports the current real status of the Periodic Schedule. If this bit is a zero then the status of the Periodic Schedule is disabled. If this bit is a one then the status of the Periodic Schedule is enabled. The Host Controller is not required to <i>immediately</i> disable or enable the Periodic Schedule when software transitions the <i>Periodic Schedule Enable</i> bit in the USBCMD register. When this bit and the <i>Periodic Schedule Enable</i> bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).</p>
13	R	0	<p><b>Reclamation</b></p> <p>This is a read-only status bit, which is used to detect an empty</p>

			asynchronous schedule.
12	R	1	<p><b>HC Halted</b></p> <p>This bit is a zero whenever the Run/Stop bit is a one. The Host Controller Sets this bit to one after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller Hardware (e.g. internal error).</p> <p>The default value is '1'.</p>
11:6	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be set to zero.</p>
5	R/WC	0	<p><b>Interrupt on Async Advance</b></p> <p>System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a one to the Interrupt on Async Advance Doorbell bit in the USBCMD register. This status bit indicates the assertion of that interrupt source.</p>
4	R/WC	0	<p><b>Host System Error</b></p> <p>The Host Controller set this bit to 1 when a serious error occurs during a host system access involving the Host Controller module. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs.</p>
3	R/WC	0	<p><b>Frame List Rollover</b></p> <p>The Host Controller sets this bit to a one when the Frame List Index rolls over from its maximum value to zero. The exact value at which the rollover occurs depends on the frame list size. For example, if the frame list size is 1024, the Frame Index Register rolls over every time FRINDEX [13] toggles. Similarly, if the size is 512, the Host Controller sets this bit to a one every time FRINDEX [12] toggles.</p>
2	R/WC	0	<p><b>Port Change Detect</b></p> <p>The Host Controller sets this bit to a one when any port for which the Port Owner bit is set to zero has a change bit transition from a zero to a one or a Force Port Resume bit transition from a zero to a one as a result of a J-K transition detected on a suspended port. This bit will also be set as a result of the Connect Status Chang being set to a one after system software has relinquished ownership of a connected port by writing a one to a port's Port Owner bit.</p>
1	R/WC	0	<p><b>USB Error Interrupt(USBERRINT)</b></p>

			<p>The Host Controller sets this bit to 1 when completion of USB transaction results in an error condition(e.g. error counter underflow).If the TD on which the error interrupt occurred also had its IOC bit set, both.</p> <p>This bit and USBINT bit are set.</p>
0	R/WC	0	<p><b>USB Interrupt(USBINT)</b></p> <p>The Host Controller sets this bit to a one on the completion of a USB transaction, which results in the retirement of a Transfer Descriptor that had its IOC bit set.</p> <p>The Host Controller also sets this bit to 1 when a short packet is detected</p> <p>(actual number of bytes received was less than the expected number of bytes)</p>

#### 8.8.3.5.8. EHCI USB Interrupt Enable Register

<b>Offset: 0x18</b>			<b>Register Name: USBINTR</b>
			<b>Default Value:0x00000000</b>
Bit	Read/Write	Default	Description
31:6	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be zero.</p>
5	R/W	0	<p><b>Interrupt on Async Advance Enable</b></p> <p>When this bit is 1, and the Interrupt on Async Advance bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.</p>
4	R/W	0	<p><b>Host System Error Enable</b></p> <p>When this bit is 1, and the Host System Error Status bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.</p>
3	R/W	0	<p><b>Frame List Rollover Enable</b></p> <p>When this bit is 1, and the Frame List Rollover bit in the USBSTS register</p>

			is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit.
2	R/W	0	<p><b>Port Change Interrupt Enable</b></p> <p>When this bit is 1, and the Port Chang Detect bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Chang Detect bit.</p>
1	R/W	0	<p><b>USB Error Interrupt Enable</b></p> <p>When this bit is 1, and the USBERRINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.</p> <p>The interrupt is acknowledged by software clearing the USBERRINT bit.</p>
0	R/W	0	<p><b>USB Interrupt Enable</b></p> <p>When this bit is 1, and the USBINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.</p> <p>The interrupt is acknowledged by software clearing the USBINT bit</p>

#### 8.8.3.5.9. EHCI Frame Index Register

<b>Offset: 0x1c</b>			<b>Register Name: FRINDEX</b>			
			<b>Default Value: 0x00000000</b>			
Bit	Read/Write	Default	Description			
31:14	/	0	<p><b>Reserved</b></p> <p>These bits are reserved and should be zero.</p>			
13:0	R/W	0	<p><b>Frame Index</b></p> <p>The value in this register increment at the end of each time frame (e.g. micro-frame). Bits[N:3] are used for the Frame List current index. It Means that each location of the frame list is accessed 8 times(frames or Micro-frames) before moving to the next index. The following illustrates Values of N based on the value of the Frame List Size field in the USBCMD register.</p> <table border="1" data-bbox="555 2011 1380 2056"> <tr> <td>USBCMD[Frame List Size]</td> <td>Number Elements</td> <td>N</td> </tr> </table>	USBCMD[Frame List Size]	Number Elements	N
USBCMD[Frame List Size]	Number Elements	N				

			00b	1024	12
			01b	512	11
			10b	256	10
			11b	Reserved	

Note: This register must be written as a DWord. Byte writes produce undefined results.

#### 8.8.3.5.10. EHCI Periodic Frame List Base Address Register

<b>Offset: 0x24</b>			<b>Register Name: PERIODICLISTBASE</b>
			<b>Default Value: Undefined</b>
Bit	Read/Write	Default	Description
31:12	R/W		<p><b>Base Address</b></p> <p>These bits correspond to memory address signals [31:12], respectively.</p> <p>This register contains the beginning address of the Periodic Frame List in the system memory.</p> <p>System software loads this register prior to starting the schedule execution by the Host Controller. The memory structure referenced by this physical memory pointer is assumed to be 4-K byte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence.</p>
11:0	/		<p><b>Reserved</b></p> <p>Must be written as 0x0 during runtime, the values of these bits are undefined.</p>

Note: Writes must be Dword Writes.

#### 8.8.3.5.11. EHCI Current Asynchronous List Address Register

<b>Offset: 0x28</b>			<b>Register Name: ASYNCLISTADDR</b>
			<b>Default Value: Undefined</b>
Bit	Read/Write	Default	Description
31:5	R/W		<p><b>Link Pointer (LP)</b></p> <p>This field contains the address of the next asynchronous queue head to</p>



			be executed.  These bits correspond to memory address signals [31:5], respectively.
4:0	/	/	<b>Reserved</b>  These bits are reserved and their value has no effect on operation.  Bits in this field cannot be modified by system software and will always return a zero when read.

Note: Write must be DWord Writes.

#### 8.8.3.5.12. EHCI Configure Flag Register

<b>Offset: 0x50</b>			<b>Register Name: CONFIGFLAG</b>  <b>Default Value: 0x00000000</b>						
Bit	Read/Write	Default	Description						
31:1	/	0	<b>Reserved</b>  These bits are reserved and should be set to zero.						
0	R/W	0	<b>Configure Flag(CF)</b>  Host software sets this bit as the last action in its process of configuring the Host Controller. This bit controls the default port-routing control logic as follow: <table border="1" data-bbox="558 1406 1380 1630"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Port routing control logic default-routs each port to an implementation dependent classic host controller.</td> </tr> <tr> <td>1</td> <td>Port routing control logic default-routs all ports to this host controller.</td> </tr> </tbody> </table> The default value of this field is '0'.	Value	Meaning	0	Port routing control logic default-routs each port to an implementation dependent classic host controller.	1	Port routing control logic default-routs all ports to this host controller.
Value	Meaning								
0	Port routing control logic default-routs each port to an implementation dependent classic host controller.								
1	Port routing control logic default-routs all ports to this host controller.								

Note: This register is not use in the normal implementation.

#### 8.8.3.5.13. EHCI Port Status and Control Register

<b>Offset: 0x54</b>	<b>Register Name: PORTSC</b>  <b>Default Value: 0x00002000(w/PPC set to one);0x00003000(w/PPC set to a zero)</b>
---------------------	--

Bit	Read/Write	Default	Description																		
31:22	/	0	<p><b>Reserved</b></p> <p>These bits are reserved for future use and should return a value of zero when read.</p>																		
21	R/W	0	<p><b>Wake on Disconnect Enable(WKDSCNNT_E)</b></p> <p>Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>																		
20	R/W	0	<p><b>Wake on Connect Enable(WKCNNT_E)</b></p> <p>Writing this bit to a one enable the port to be sensitive to device connects as wake-up events.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>																		
19:16	R/W	0	<p><b>Port Test Control</b></p> <p>The value in this field specifies the test mode of the port. The encoding of the test mode bits are as follow:</p> <table border="1" data-bbox="555 1301 1386 1780"> <thead> <tr> <th>Bits</th> <th>Test Mode</th> </tr> </thead> <tbody> <tr> <td>0000b</td> <td>The port is NOT operating in a test mode.</td> </tr> <tr> <td>0001b</td> <td>Test J_STATE</td> </tr> <tr> <td>0010b</td> <td>Test K_STATE</td> </tr> <tr> <td>0011b</td> <td>Test SEO_NAK</td> </tr> <tr> <td>0100b</td> <td>Test Packet</td> </tr> <tr> <td>0101b</td> <td>Test FORCE_ENABLE</td> </tr> <tr> <td>0110b</td> <td rowspan="3">Reserved</td> </tr> <tr> <td>-</td> </tr> <tr> <td>1111b</td> </tr> </tbody> </table> <p>The default value in this field is '0000b'.</p>	Bits	Test Mode	0000b	The port is NOT operating in a test mode.	0001b	Test J_STATE	0010b	Test K_STATE	0011b	Test SEO_NAK	0100b	Test Packet	0101b	Test FORCE_ENABLE	0110b	Reserved	-	1111b
Bits	Test Mode																				
0000b	The port is NOT operating in a test mode.																				
0001b	Test J_STATE																				
0010b	Test K_STATE																				
0011b	Test SEO_NAK																				
0100b	Test Packet																				
0101b	Test FORCE_ENABLE																				
0110b	Reserved																				
-																					
1111b																					
15:14	R/W	0	<p><b>Reserved</b></p> <p>These bits are reserved for future use and should return a value of zero when read.</p>																		
13	R/W	1	<p><b>Port Owner</b></p>																		

			<p>This bit unconditionally goes to a 0b when the Configured bit in the CONFIGFLAG register makes a 0b to 1b transition. This bit unconditionally goes to 1b whenever the Configured bit is zero.</p> <p>System software uses this field to release ownership of the port to selected host controller (in the event that the attached device is not a high-speed device). Software writes a one to this bit when the attached device is not a high-speed device. A one in this bit means that a companion host controller owns and controls the port.</p> <p>Default Value = 1b.</p>															
12	/	0	<p><b>Reserved</b></p> <p>These bits are reserved for future use and should return a value of zero when read.</p>															
11:10	R	0	<p><b>Line Status</b></p> <p>These bits reflect the current logical levels of the D+ (bit11) and D-(bit10) signal lines. These bits are used for detection of low-speed USB devices prior to port reset and enable sequence. This read only field is valid only when the port enable bit is zero and the current connect status bit is set to a one.</p> <p>The encoding of the bits are:</p> <table border="1" data-bbox="558 1265 1380 1653"> <thead> <tr> <th>Bit[11:10]</th> <th>USB State</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>SE0</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> <tr> <td>10b</td> <td>J-state</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> <tr> <td>01b</td> <td>K-state</td> <td>Low-speed device, release ownership of port.</td> </tr> <tr> <td>11b</td> <td>Undefined</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> </tbody> </table> <p>This value of this field is undefined if Port Power is zero.</p>	Bit[11:10]	USB State	Interpretation	00b	SE0	Not Low-speed device, perform EHCI reset.	10b	J-state	Not Low-speed device, perform EHCI reset.	01b	K-state	Low-speed device, release ownership of port.	11b	Undefined	Not Low-speed device, perform EHCI reset.
Bit[11:10]	USB State	Interpretation																
00b	SE0	Not Low-speed device, perform EHCI reset.																
10b	J-state	Not Low-speed device, perform EHCI reset.																
01b	K-state	Low-speed device, release ownership of port.																
11b	Undefined	Not Low-speed device, perform EHCI reset.																
9	/	0	<p><b>Reserved</b></p> <p>This bit is reserved for future use, and should return a value of zero when read.</p>															
8	R/W	0	<p><b>Port Reset</b></p> <p>1=Port is in Reset. 0=Port is not in Reset. Default value = 0.</p> <p>When software writes a one to this bit (from a zero), the bus reset</p>															

		<p>sequence as defined in the USB Specification Revision 2.0 is started. Software writes a zero to this bit to terminate the bus reset sequence. Software must keep this bit at a one long enough to ensure the reset sequence, as specified in the USB Specification Revision 2.0, completes. Notes: when software writes this bit to a one , it must also write a zero to the Port Enable bit.</p> <p>Note that when software writes a zero to this bit there may be a delay before the bit status changes to a zero. The bit status will not read as a zero until after the reset has completed. If the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (e.g. set the Port Enable bit to a one). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from a one to a zero. For example: if the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state with 2ms of software writing this bit to a zero.</p> <p>The HC Halted bit in the USBSTS register should be a zero before software attempts to use this bit. The host controller may hold Port Reset asserted to a one when the HC Halted bit is a one.</p> <p>This field is zero if Port Power is zero.</p>								
7	R/W	<p><b>Suspend</b></p> <p>Port Enabled Bit and Suspend bit of this register define the port states as follows:</p> <table border="1" data-bbox="558 1400 1380 1572"> <thead> <tr> <th>Bits[Port Enables, Suspend]</th> <th>Port State</th> </tr> </thead> <tbody> <tr> <td>0x</td> <td>Disable</td> </tr> <tr> <td>10</td> <td>Enable</td> </tr> <tr> <td>11</td> <td>Suspend</td> </tr> </tbody> </table> <p>When in suspend state, downstream propagation of data is blocked on this port, except for port reset. The blocking occurs at the end of the current transaction, if a transaction was in progress when this bit was written to 1. In the suspend state, the port is sensitive to resume detection. Not that the bit status does not change until the port is suspend and that there may be a delay in suspending a port if there is a transaction currently in progress on the USB.</p> <p>A write of zero to this bit is ignored by the host controller. The host controller will unconditionally set this bit to a zero when:</p>	Bits[Port Enables, Suspend]	Port State	0x	Disable	10	Enable	11	Suspend
Bits[Port Enables, Suspend]	Port State									
0x	Disable									
10	Enable									
11	Suspend									

			<p>① Software sets the Force Port Resume bit to a zero(from a one).</p> <p>② Software sets the Port Reset bit to a one(from a zero).</p> <p>If host software sets this bit to a one when the port is not enabled(i.e. Port enabled bit is a zero), the results are undefined.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>
			<p><b>Force Port Resume</b></p> <p>1 = Resume detected/driven on port. 0 = No resume (K-state) detected/driven on port. Default value = 0.</p> <p>This functionality defined for manipulating this bit depends on the value of the Suspend bit. For example, if the port is not suspend and software transitions this bit to a one, then the effects on the bus are undefined.</p> <p>Software sets this bit to a 1 drive resume signaling. The Host Controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a one because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register is also set to a one. If software sets this bit to a one, the host controller must not set the Port Change Detect bit.</p> <p>Note that when the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB Specification Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this remains a one. Software must appropriately time the Resume and set this bit to a zero when the appropriate amount of time has elapsed. Writing a zero (from one) causes the port to return high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a one until the port has switched to high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to a zero.</p>
6	R/W	0	<p>This field is zero if Port Power is zero.</p>
5	R/WC	0	<p><b>Over-current Change</b></p> <p>Default = 0. This bit gets set to a one when there is a change to Over-current Active. Software clears this bit by writing a one to this bit position.</p>
4	R	0	<p><b>Over-current Active</b></p>

			<p>0 = This port does not have an over-current condition. 1 = This port currently has an over-current condition. This bit will automatically transition from a one to a zero when the over current condition is removed.</p> <p>The default value of this bit is '0'.</p>
3	R/WC	0	<p><b>Port Enable/Disable Change</b></p> <p>Default = 0. 1 = Port enabled/disabled status has changed. 0 = No change.</p> <p>For the root hub, this bit gets set to a one only when a port is disabled due to the appropriate conditions existing at the EOF2 point (See Chapter 11 of the USB Specification for the definition of a Port Error). Software clears this bit by writing a 1 to it.</p> <p>This field is zero if Port Power is zero.</p>
2	R/W	0	<p><b>Port Enabled/Disabled</b></p> <p>1=Enable, 0=Disable. Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a one to this field. The host controller will only set this bit to a one when the reset sequence determines that the attached device is a high-speed device.</p> <p>Ports can be disabled by either a fault condition(disconnect event or other fault condition) or by host software. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events.</p> <p>When the port is disabled, downstream propagation of data is blocked on this port except for reset.</p> <p>The default value of this field is '0'.</p> <p>This field is zero if Port Power is zero.</p>
1	R/WC	0	<p><b>Connect Status Change</b></p> <p>1=Change in Current Connect Status, 0=No change, Default=0.</p> <p>Indicates a change has occurred in the port's Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. For example, the insertion status changes twice</p>

			<p>before system software has cleared the changed condition, hub hardware will be “setting” an already-set bit. Software sets this bit to 0 by writing a 1 to it.</p> <p>This field is zero if Port Power is zero.</p>
0	R	0	<p><b>Current Connect Status</b></p> <p>Device is present on port when the value of this field is a one, and no device is present on port when the value of this field is a zero. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change(Bit 1) to be set.</p> <p>This field is zero if Port Power zero.</p>

Note: This register is only reset by hardware or in response to a host controller reset.

### 8.8.3.6. OHCI Register Description

#### 8.8.3.6.1. HcRevision Register

<b>Offset: 0x400</b>				<b>Register Name: HcRevision</b>
				<b>Default Value:0x10</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:8	/	/	0x00	<b>Reserved</b>
7:0	R	R	0x10	<p><b>Revision</b></p> <p>This read-only field contains the BCD representation of the version of the HCI specification that is implemented by this HC. For example, a value of 0x11 corresponds to version 1.1. All of the HC implementations that are compliant with this specification will have a value of 0x10.</p>

#### 8.8.3.6.2. HcControl Register

<b>Offset: 0x404</b>			<b>Register Name: HcRevision</b>
			<b>Default Value:0x0</b>
Bit	Read/Write	Default	Description

	HCD	HC										
31:11	/	/	0x00	<b>Reserved</b>								
10	R/W	R	0x0	<p><b>RemoteWakeupEnable</b></p> <p>This bit is used by HCD to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the <b>ResumeDetected</b> bit in <b>HcInterruptStatus</b> is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupt.</p>								
9	R/W	R/W	0x0	<p><b>RemoteWakeupConnected</b></p> <p>This bit indicates whether HC supports remote wakeup signaling. If remote wakeup is supported and used by the system, it is the responsibility of system firmware to set this bit during POST. HC clear the bit upon a hardware reset but does not alter it upon a software reset. Remote wakeup signaling of the host system is host-bus-specific and is not described in this specification.</p>								
8	R/W	R	0x0	<p><b>InterruptRouting</b></p> <p>This bit determines the routing of interrupts generated by events registered in HcInterruptStatus. If clear, all interrupt are routed to the normal host bus interrupt mechanism. If set interrupts are routed to the System Management Interrupt. HCD clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. HCD uses this bit as a tag to indicate the ownership of HC.</p>								
7:6	R/W	R/W	0x0	<p>HostControllerFunctionalState for USB</p> <table border="1" data-bbox="558 1366 1380 1702"> <tbody> <tr> <td>00 b</td> <td>USBReset</td> </tr> <tr> <td>01 b</td> <td>USBResume</td> </tr> <tr> <td>10 b</td> <td>USBOperational</td> </tr> <tr> <td>11 b</td> <td>USBSuspend</td> </tr> </tbody> </table> <p>A transition to USBOperational from another state causes SOF generation to begin 1 ms later. HCD may determine whether HC has begun sending SOFs by reading the StartoFrame field of HcInterruptStatus.</p> <p>This field may be changed by HC only when in the USBsuspend state. HC may move from the USBsuspend state to the USBRESUME state</p>	00 b	USBReset	01 b	USBResume	10 b	USBOperational	11 b	USBSuspend
00 b	USBReset											
01 b	USBResume											
10 b	USBOperational											
11 b	USBSuspend											



				<p>after detecting the resume signaling from a downstream port.</p> <p>HC enters USB_SUSPEND after a software reset, whereas it enters USB_RESET after a hardware reset. The latter also resets the Root Hub and asserts subsequent reset signaling to downstream ports.</p>
5	R/W	R	0x0	<p><b>BulkListEnable</b></p> <p>This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If <i>HcBulkCurrentED</i> is pointing to an ED to be removed, HCD must advance the pointer by updating <i>HcBulkCurrentED</i> before re-enabling processing of the list.</p>
4	R/W	R	0x0	<p><b>ControlListEnable</b></p> <p>This bit is set to enable the processing of the Control list in the next Frame. If cleared by HCD, processing of the Control list does not occur after the next SOF. HC must check this bit whenever it determines to process the list. When disabled, HCD may modify the list. If <i>HcControlCurrentED</i> is pointing to an ED to be removed, HCD must advance the pointer by updating <i>HcControlCurrentED</i> before re-enabling processing of the list.</p>
3	R/W	R	0x0	<p><b>IsochronousEnable</b></p> <p>This bit is used by HCD to enable/disable processing of isochronous EDs. While processing the periodic list in a Frame, HC checks the status of this bit when it finds an Isochronous ED (F=1). If set (enabled), HC continues processing the EDs. If cleared (disabled), HC halts processing of the periodic list (which now contains only isochronous EDs) and begins processing the Bulk/Control lists.</p> <p>Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).</p>
2	R/W	R	0x0	<p><b>PeriodicListEnable</b></p> <p>This bit is set to enable the processing of periodic list in the next Frame. If cleared by HCD, processing of the periodic list does not occur after the next SOF. HC must check this bit before it starts processing the list.</p>
1:0	R/W	R	0x0	<p><b>ControlBulkServiceRatio</b></p> <p>This specifies the service ratio between Control and Bulk EDs. Before processing any of the nonperiodic lists, HC must compare the ratio</p>

				<p>specified with its internal count on how many nonempty Control EDs have been processed, in determining whether to continue serving another Control ED or switching to Bulk EDs. The internal count will be retained when crossing the frame boundary. In case of reset, HCD is responsible for restoring this value.</p> <table border="1"> <thead> <tr> <th>CBSR</th> <th>No. of Control EDs Over Bulk EDs Served</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1:1</td> </tr> <tr> <td>1</td> <td>2:1</td> </tr> <tr> <td>2</td> <td>3:1</td> </tr> <tr> <td>3</td> <td>4:1</td> </tr> </tbody> </table> <p>The default value is 0x0.</p>	CBSR	No. of Control EDs Over Bulk EDs Served	0	1:1	1	2:1	2	3:1	3	4:1
CBSR	No. of Control EDs Over Bulk EDs Served													
0	1:1													
1	2:1													
2	3:1													
3	4:1													

### 8.8.3.6.3. HcCommandStatus Register

Offset: 0x408				Register Name: HcCommandStatus
Default Value: 0x0				
Bit	Read/Write		Default	Description
	HCD	HC		
31:18	/	/	0x0	<b>Reserved</b>
17:16	R	R/W	0x0	<b>SchedulingOverrunCount</b> These bits are incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if <b>SchedulingOverrun</b> in <i>HcInterruptStatus</i> has already been set. This is used by HCD to monitor any persistent scheduling problem.
15:4	/	/	0x0	<b>Reserved</b>
3	R/W	R/W	0x0	<b>OwershipChangeRequest</b> This bit is set by an OS HCD to request a change of control of the HC. When set HC will set the <b>OwnershipChange</b> field in <i>HcInterruptStatus</i> . After the changeover, this bit is cleared and remains so until the next request from OS HCD.
2	R/W	R/W	0x0	<b>BulkListFilled</b> This bit is used to indicate whether there are any TDs on the Bulk list. It is set by HCD whenever it adds a TD to an ED in the Bulk list. When HC begins to process the head of the Bulk list, it checks BLF. As long as <b>BulkListFilled</b> is 0, HC will not start processing the Bulk list. If

				<p><b>BulkListFilled</b> is 1, HC will start processing the Bulk list and will set BF to 0. If HC finds a TD on the list, then HC will set <b>BulkListFilled</b> to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set <b>BulkListFilled</b>, then <b>BulkListFilled</b> will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop.</p>
1	R/W	R/W	0x0	<p><b>ControlListFilled</b></p> <p>This bit is used to indicate whether there are any TDs on the Control list. It is set by HCD whenever it adds a TD to an ED in the Control list.</p> <p>When HC begins to process the head of the Control list, it checks CLF. As long as <b>ControlListFilled</b> is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set <b>ControlListFilled</b> to 0. If HC finds a TD on the list, then HC will set <b>ControlListFilled</b> to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if the HCD does not set <b>ControlListFilled</b>, then <b>ControlListFilled</b> will still be 0 when HC completes processing the Control list and Control list processing will stop.</p>
0	R/W	R/E	0x0	<p><b>HostControllerReset</b></p> <p>This bit is by HCD to initiate a software reset of HC. Regardless of the functional state of HC, it moves to the USBsuspend state in which most of the operational registers are reset except those stated otherwise; e.g, the InterruptRouting field of HcControl, and no Host bus accesses are allowed. This bit is cleared by HC upon the completion of the reset operation. The reset operation must be completed within 10 ms. This bit,</p> <p>when set, should not cause a reset to the Root Hub and no subsequent reset signaling should be asserted to its downstream ports.</p>

#### 8.8.3.6.4. HcInterruptStatus Register

<b>Offset: 0x40c</b>				<b>Register Name: HcInterruptStatus</b>
				<b>Default Value:0x00</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:7	/	/	0x0	<b>Reserved</b>
6	R/W	R/W	0x0 0x1?	<b>RootHubStatusChange</b>

				This bit is set when the content of <i>HcRhStatus</i> or the content of any of <i>HcRhPortStatus</i> [ <b>NumberOfDownstreamPort</b> ] has changed.
5	R/W	R/W	0x0	<b>FrameNumberOverflow</b>  This bit is set when the MSb of <i>HcFmNumber</i> (bit 15) changes value, from 0 to 1 or from 1 to 0, and after <i>HccaFrameNumber</i> has been updated.
4	R/W	R/W	0x0	<b>UnrecoverableError</b>  This bit is set when HC detects a system error not related to USB. HC should not proceed with any processing nor signaling before the system error has been corrected. HCD clears this bit after HC has been reset.
3	R/W	R/W	0x0	<b>ResumeDetected</b>  This bit is set when HC detects that a device on the USB is asserting resume signaling. It is the transition from no resume signaling to resume signaling causing this bit to be set. This bit is not set when HCD sets the USBRseume state.
2	R/W	R/W	0x0	<b>StartofFrame</b>  This bit is set by HC at each start of frame and after the update of <i>HccaFrameNumber</i> . HC also generates a SOF token at the same time.
1	R/W	R/W	0x0	<b>WritebackDoneHead</b>  This bit is set immediately after HC has written <i>HcDoneHead</i> to <i>HccaDoneHead</i> . Further updates of the <i>HccaDoneHead</i> will not occur until this bit has been cleared. HCD should only clear this bit after it has saved the content of <i>HccaDoneHead</i> .
0	R/W	R/W	0x0	<b>SchedulingOverrun</b>  This bit is set when the USB schedule for the current Frame overruns and after the update of <i>HccaFrameNumber</i> . A scheduling overrun will also cause the <b>SchedulingOverrunCount</b> of <i>HcCommandStatus</i> to be incremented.

#### 8.8.3.6.5. HcInterruptEnable Register

<b>Offset: 0x410</b>			<b>Register Name: HcInterruptEnable Register</b>
			<b>Default Value: 0x0</b>
Bit	Read/Write	Default	Description

	HCD	HC						
				<b>MasterInterruptEnable</b>				
31	R/W	R	0x0	A '0' written to this field is ignored by HC. A '1' written to this field enables interrupt generation due to events specified in the other bits of this register. This is used by HCD as Master Interrupt Enable.				
30:7	/	/	0x0	<b>Reserved</b>				
				<b>RootHubStatusChange Interrupt Enable</b>				
6	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Root Hub Status Change;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Root Hub Status Change;
0	Ignore;							
1	Enable interrupt generation due to Root Hub Status Change;							
				<b>FrameNumberOverflow Interrupt Enable</b>				
5	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Frame Number Over Flow;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Frame Number Over Flow;
0	Ignore;							
1	Enable interrupt generation due to Frame Number Over Flow;							
				<b>UnrecoverableError Interrupt Enable</b>				
4	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Unrecoverable Error;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Unrecoverable Error;
0	Ignore;							
1	Enable interrupt generation due to Unrecoverable Error;							
				<b>ResumeDetected Interrupt Enable</b>				
3	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Resume Detected;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Resume Detected;
0	Ignore;							
1	Enable interrupt generation due to Resume Detected;							
				<b>StartofFrame Interrupt Enable</b>				
2	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Start of Flame;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Start of Flame;
0	Ignore;							
1	Enable interrupt generation due to Start of Flame;							
				<b>WritebackDoneHead Interrupt Enable</b>				
1	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Write back Done Head;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Write back Done Head;
0	Ignore;							
1	Enable interrupt generation due to Write back Done Head;							
				<b>SchedulingOverrun Interrupt Enable</b>				
0	R/W	R	0x0	<table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Enable interrupt generation due to Scheduling Overrun;</td> </tr> </table>	0	Ignore;	1	Enable interrupt generation due to Scheduling Overrun;
0	Ignore;							
1	Enable interrupt generation due to Scheduling Overrun;							

### 8.8.3.6.6. HcInterruptDisable Register

<b>Offset: 0x414</b>	<b>Register Name: HcInterruptDisable Register</b>
	<b>Default Value: 0x0</b>

Bit	Read/Write		Default	Description				
	HCD	HC						
31	R/W	R	0x0	<b>MasterInterruptEnable</b> A written '0' to this field is ignored by HC. A '1' written to this field disables interrupt generation due events specified in the other bits of this register. This field is set after a hardware or software reset.				
30:7	/	/	0x00	<b>Reserved</b>				
6	R/W	R	0x0	<b>RootHubStatusChange Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Root Hub Status Change;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Root Hub Status Change;
0	Ignore;							
1	Disable interrupt generation due to Root Hub Status Change;							
5	R/W	R	0x0	<b>FrameNumberOverflow Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Frame Number Over Flow;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Frame Number Over Flow;
0	Ignore;							
1	Disable interrupt generation due to Frame Number Over Flow;							
4	R/W	R	0x0	<b>UnrecoverableError Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Unrecoverable Error;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Unrecoverable Error;
0	Ignore;							
1	Disable interrupt generation due to Unrecoverable Error;							
3	R/W	R	0x0	<b>ResumeDetected Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Resume Detected;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Resume Detected;
0	Ignore;							
1	Disable interrupt generation due to Resume Detected;							
2	R/W	R	0x0	<b>StartofFrame Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Start of Flame;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Start of Flame;
0	Ignore;							
1	Disable interrupt generation due to Start of Flame;							
1	R/W	R	0x0	<b>WritebackDoneHead Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Write back Done Head;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Write back Done Head;
0	Ignore;							
1	Disable interrupt generation due to Write back Done Head;							
0	R/w	R	0x0	<b>SchedulingOverrun Interrupt Disable</b> <table border="1"> <tr> <td>0</td> <td>Ignore;</td> </tr> <tr> <td>1</td> <td>Disable interrupt generation due to Scheduling Overrun;</td> </tr> </table>	0	Ignore;	1	Disable interrupt generation due to Scheduling Overrun;
0	Ignore;							
1	Disable interrupt generation due to Scheduling Overrun;							

### 8.8.3.6.7. HcHCCA Register

<b>Offset: 0x418</b>	<b>Register Name: HcHCCA</b>  <b>Default Value:0x0</b>
----------------------	--

Bit	Read/Write		Default	Description
	HCD	HC		
31:8	R/W	R	0x0	<b>HCCA[31:8]</b> This is the base address of the Host Controller Communication Area. This area is used to hold the control structures and the Interrupt table that are accessed by both the Host Controller and the Host Controller Driver.
7:0	R	R	0x0	<b>HCCA[7:0]</b> The alignment restriction in HcHCCA register is evaluated by examining the number of zeros in the lower order bits. The minimum alignment is 256 bytes, therefore, bits 0 through 7 must always return 0 when read.

#### 8.8.3.6.8. HcPeriodCurrentED Register

<b>Offset: 0x41c</b>				<b>Register Name: HcPeriodCurrentED(PCED)</b> <b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:4	R	R/W	0x0	<b>PCED[31:4]</b> This is used by HC to point to the head of one of the Periodec list which will be processed in the current Frame. The content of this register is updated by HC after a periodic ED has been processed. HCD may read the content in determining which ED is currently being processed at the time of reading.
3:0	R	R	0x0	<b>PCED[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

#### 8.8.3.6.9. HcControlHeadED Register

<b>Offset: 0x420</b>				<b>Register Name: HcControlHeadED[CHED]</b> <b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		

31:4	R/W	R	0x0	<b>EHCD[31:4]</b>  The HcControlHeadED register contains the physical address of the first Endpoint Descriptor of the Control list. HC traverse the Control list starting with the HcControlHeadED pointer. The content is loaded from HCCA during the initialization of HC.
3:0	R	R	0x0	<b>EHCD[3:0]</b>  Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

#### 8.8.3.6.10. HcControlCurrentED Register

<b>Offset: 0x424</b>				<b>Register Name: HcControlCurrentED[CCED]</b>  <b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:4	R/W	R/W	0x0	<b>CCED[31:4]</b>  The pointer is advanced to the next ED after serving the present one. HC will continue processing the list from where it left off in the last Frame. When it reaches the end of the Control list, HC checks the ControlListFilled of in HcCommandStatus. If set, it copies the content of HcControlHeadED to HcControlCurrentED and clears the bit. If not set, it does nothing.  HCD is allowed to modify this register only when the ControllListEnable of HcControl is cleared. When set, HCD only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list.
3:0	R	R	0x0	<b>CCED[3:0]</b>  Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.



**8.8.3.6.11. HcBulkHeadED Register**

Offset: 0x428				Register Name: HcBulkHeadED[BHED]
Default Value: 0x0				
Bit	Read/Write		Default	Description
	HCD	HC		
31:4	R/W	R	0x0	<b>BHED[31:4]</b> The HcBulkHeadED register contains the physical address of the first Endpoint Descriptor of the Bulk list. HC traverses the Bulk list starting with the HcBulkHeadED pointer. The content is loaded from HCCA during the initialization of HC.
3:0	R	R	0x0	<b>BHED[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

**8.8.3.6.12. HcBulkCurrentED Register**

Offset: 0x42c				Register Name: HcBulkCurrentED [BCED]
Default Value: 0x00				
Bit	Read/Write		Default	Description
	HCD	HC		
31:4	R/W	R/W	0x0	<b>BulkCurrentED[31:4]</b> This is advanced to the next ED after the HC has served the present one. HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, HC checks the <b>ControlListFilled</b> of HcControl. If set, it copies the content of <i>HcBulkHeadED</i> to <i>HcBulkCurrentED</i> and clears the bit. If it is not set, it does nothing. HCD is only allowed to modify this register when the <b>BulkListEnable</b> of <i>HcControl</i> is cleared. When set, the HCD only reads the instantaneous value of this register. This is initially set to zero to indicate the end of the Bulk list.
3:0	R	R	0x0	<b>BulkCurrentED [3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

### 8.8.3.6.13. HcDoneHead Register

Offset: 0x430				Register Name: HcDoneHead
Default Value: 0x00				
Bit	Read/Write		Default	Description
	HCD	HC		
31:4	R	R/W	0x0	<b>HcDoneHead[31:4]</b>  When a TD is completed, HC writes the content of <i>HcDoneHead</i> to the NextTD field of the TD. HC then overwrites the content of <i>HcDoneHead</i> with the address of this TD. This is set to zero whenever HC writes the content of this register to HCCA. It also sets the <b>WritebackDoneHead</b> of <i>HcInterruptStatus</i> .
3:0	R	R	0x0	<b>HcDoneHead[3:0]</b>  Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

### 8.8.3.6.14. HcFmInterval Register

Offset: 0x434				Register Name: HcFmInterval Register
Default Value: 0x2EDF				
Bit	Read/Write		Default	Description
	HCD	HC		
31	R/W	R	0x0	<b>FrameIntervalToggler</b>  HCD toggles this bit whenever it loads a new value to <b>FrameInterval</b> .
30:16	R/W	R	0x0	<b>FSLargestDataPacket</b>  This field specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame. The counter value represents the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The field value is calculated by the HCD.
15:14	/	/	0x0	<b>Reserved</b>
13:0	R/W	R	0x2edf	<b>FrameInterval</b>

				This specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999. HCD should store the current value of this field before resetting HC. By setting the <b>HostControllerReset</b> field of <i>HcCommandStatus</i> as this will cause the HC to reset this field to its nominal value. HCD may choose to restore the stored value upon the completion of the Reset sequence.
--	--	--	--	---

#### 8.8.3.6.15. HcFmRemaining Register

<b>Offset: 0x438</b>				<b>Register Name: HcFmRemaining</b> <b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31	R	R/W	0x0	<b>FrameRemaining Toggle</b> This bit is loaded from the <b>FrameIntervalToggle</b> field of <i>HcFmInterval</i> whenever <b>FrameRemaining</b> reaches 0. This bit is used by HCD for the synchronization between <b>FrameInterval</b> and <b>FrameRemaining</b> .
30:14	/	/	0x0	<b>Reserved</b>
13:0	R	RW	0x0	<b>FramRemaining</b> This counter is decremented at each bit time. When it reaches zero, it is reset by loading the <b>FrameInterval</b> value specified in <i>HcFmInterval</i> at the next bit time boundary. When entering the USBOPERATIONAL state, HC re-loads the content with the <b>FrameInterval</b> of <i>HcFmInterval</i> and uses the updated value from the next SOF.

#### 8.8.3.6.16. HcFmNumber Register

<b>Offset: 0x43c</b>				<b>Register Name: HcFmNumber</b> <b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:16				<b>Reserved</b>
15:0	R	R/W	0x0	<b>FrameNumber</b> This is incremented when <i>HcFmRemaining</i> is re-loaded. It will be rolled over to 0x0 after 0x0ffff. When entering the USBOPERATIONAL state, this will be incremented automatically. The content will be written to

				HCCA after HC has incremented the <b>FrameNumber</b> at each frame boundary and sent a SOF but before HC reads the first ED in that Frame. After writing to HCCA, HC will set the <b>StartofFrame</b> in <i>HcInterruptStatus</i> .
--	--	--	--	---

#### 8.8.3.6.17. HcPeriodicStart Register

<b>Offset: 0x440</b>				<b>Register Name: HcPeriodicStatus</b>
				<b>Default Value: 0x0</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:14				<b>Reserved</b>
13:0	R/W	R	0x0	<b>PeriodicStart</b> After a hardware reset, this field is cleared. This is then set by HCD during the HC initialization. The value is calculated roughly as 10% off from <i>HcFmInterval</i> . A typical value will be 0x2A3F (0x3e67??). When <i>HcFmRemaining</i> reaches the value specified, processing of the periodic lists will have priority over Control/Bulk processing. HC will therefore start processing the Interrupt list after completing the current Control or Bulk transaction that is in progress.

#### 8.8.3.6.18. HclSThreshold Register

<b>Offset: 0x444</b>				<b>Register Name: HclSThreshold</b>
				<b>Default Value: 0x0628</b>
Bit	Read/Write		Default	Description
	HCD	HC		
31:12				<b>Reserved</b>
11:0	R/W	R	0x0628	<b>LSThreshold</b> This field contains a value which is compared to the <b>FrameRemaining</b> field prior to initiating a Low Speed transaction. The transaction is started only if <b>FrameRemaining</b> <sup>3</sup> this field. The value is calculated by HCD with the consideration of transmission and setup overhead.

**8.8.3.6.19. HcRhDescriptorA Register**

Offset: 0x448				Register Name: HcRhDescriptorA				
Default Value:								
Bit	Read/Write		Default	Description				
	HCD	HC						
31:24	R/W	R	0x2	<b>PowerOnToPowerGoodTime[POTPGT]</b> This byte specifies the duration HCD has to wait before accessing a powered-on port of the Root Hub. It is implementation-specific. The unit of time is 2 ms. The duration is calculated as POTPGT * 2ms.				
23:13				<b>Reserved</b>				
12	R/W	R	1	<b>NoOverCurrentProtection</b> This bit describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the <b>OverCurrentProtectionMode</b> field specifies global or per-port reporting. <table border="1" data-bbox="558 1052 1380 1187"> <tr> <td>0</td> <td>Over-current status is reported collectively for all downstream ports.</td> </tr> <tr> <td>1</td> <td>No overcurrent protection supported.</td> </tr> </table>	0	Over-current status is reported collectively for all downstream ports.	1	No overcurrent protection supported.
0	Over-current status is reported collectively for all downstream ports.							
1	No overcurrent protection supported.							
11	R/W	R	0	<b>OverCurrentProtectionMode</b> This bit describes how the overcurrent status for the Root Hub ports are reported. At reset, these fields should reflect the same mode as <b>PowerSwitchingMode</b> . This field is valid only if the <b>NoOverCurrentProtection</b> field is cleared. <table border="1" data-bbox="558 1467 1380 1601"> <tr> <td>0</td> <td>Over-current status is reported collectively for all downstream ports.</td> </tr> <tr> <td>1</td> <td>Over-current status is reported on per-port basis.</td> </tr> </table>	0	Over-current status is reported collectively for all downstream ports.	1	Over-current status is reported on per-port basis.
0	Over-current status is reported collectively for all downstream ports.							
1	Over-current status is reported on per-port basis.							
10	R	R	0x0	<b>Device Type</b> This bit specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read/write 0.				
9	R/W	R	1	<b>PowerSwitchingMode</b> This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the <b>NoPowerSwitching</b> field is cleared.				

				<table border="1"> <tr> <td>0</td> <td>All ports are powered at the same time.</td> </tr> <tr> <td>1</td> <td>Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands (<b>Set/ClearPortPower</b>). If the port mask is cleared, then the port is controlled only by the global power switch (<b>Set/ClearGlobalPower</b>).</td> </tr> </table>	0	All ports are powered at the same time.	1	Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands ( <b>Set/ClearPortPower</b> ). If the port mask is cleared, then the port is controlled only by the global power switch ( <b>Set/ClearGlobalPower</b> ).
0	All ports are powered at the same time.							
1	Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands ( <b>Set/ClearPortPower</b> ). If the port mask is cleared, then the port is controlled only by the global power switch ( <b>Set/ClearGlobalPower</b> ).							
8	R/W	R	0	<p><b>NoPowerSwitcing</b></p> <p>These bits are used to specify whether power switching is supported or ports are always powered. It is implementation-specific. When this bit is cleared, the <b>PowerSwitchingMode</b> specifies global or per-port switching.</p> <table border="1"> <tr> <td>0</td> <td>Ports are power switched.</td> </tr> <tr> <td>1</td> <td>Ports are always powered on when the HC is powered on.</td> </tr> </table>	0	Ports are power switched.	1	Ports are always powered on when the HC is powered on.
0	Ports are power switched.							
1	Ports are always powered on when the HC is powered on.							
7:0	R	R	0x01	<p><b>NumberDownstreamPorts</b></p> <p>These bits specify the number of downstream ports supported by the Root Hub. It is implementation-specific. The minimum number of ports is 1. The maximum number of ports supported.</p>				

#### 8.8.3.6.20. HcRhDescriptorB Register

<b>Offset: 0x44c</b>				<b>Register Name: HcRhDescriptorB Register</b>			
				<b>Default Value:</b>			
Bit	Read/Write		Default	Description			
	HCD	HC					
31:16	R/W	R	0x0	<p><b>PortPowerControlMask</b></p> <p>Each bit indicates if a port is affected by a global power control command when <b>PowerSwitchingMode</b> is set. When set, the port's power state is only affected by per-port power control (<b>Set/ClearPortPower</b>). When cleared, the port is controlled by the global power switch (<b>Set/ClearGlobalPower</b>). If the device is configured to global switching mode (<b>PowerSwitchingMode = 0</b>), this field is not valid.</p> <table border="1"> <tr> <td>Bit0</td> <td>Reserved</td> </tr> </table>		Bit0	Reserved
Bit0	Reserved						

				Bit1	Ganged-power mask on Port #1.
				Bit2	Ganged-power mask on Port #2.
				...	
				Bit15	Ganged-power mask on Port #15.
				<b>DeviceRemovable</b>	
				Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable.	
				Bit0	Reserved
				Bit1	Device attached to Port #1.
				Bit2	Device attached to Port #2.
				...	
15:0	R/W	R	0x0	Bit15	Device attached to Port #15.

#### 8.8.3.6.21. HcRhStatus Register

<b>Offset: 0x450</b>				<b>Register Name: HcRhStatus Register</b>	
				<b>Default Value:</b>	
Bit	Read/Write		Default	Description	
	HCD	HC			
31	W	R	0	<b>(write)ClearRemoteWakeupEnable</b> Write a '1' clears <b>DeviceRemoteWakeupEnable</b> . Write a '0' has no effect.	
30:18	/	/	0x0	<b>Reserved</b>	
17	R/W	R	0	<b>OverCurrentIndicatorChang</b> This bit is set by hardware when a change has occurred to the <b>OverCurrentIndicator</b> field of this register. The HCD clears this bit by writing a '1'.Writing a '0' has no effect.	
16	R/W	R	0x0	<b>(read)LocalPowerStartusChange</b> The Root Hub does not support the local power status features, thus, this bit is always read as '0'.  <b>(write)SetGlobalPower</b>	

				<p>In global power mode (<b>PowerSwitchingMode=0</b>), This bit is written to '1' to turn on power to all ports (clear <b>PortPowerStatus</b>). In per-port power mode, it sets <b>PortPowerStatus</b> only on ports whose <b>PortPowerControlMask</b> bit is not set. Writing a '0' has no effect.</p>				
15	R/W	R	0x0	<p><b>(read)DeviceRemoteWakeupEnable</b></p> <p>This bit enables a <b>ConnectStatusChange</b> bit as a resume event, causing a USBsuspend to USBResume state transition and setting the <b>ResumeDetected</b> interrupt.</p> <table border="1"> <tr> <td>0</td> <td><b>ConnectStatusChange</b> is not a remote wakeup event.</td> </tr> <tr> <td>1</td> <td><b>ConnectStatusChange</b> is a remote wakeup event.</td> </tr> </table> <p><b>(write)SetRemoteWakeupEnable</b></p> <p>Writing a '1' sets DeviceRemoveWakeupEnable. Writing a '0' has no effect.</p>	0	<b>ConnectStatusChange</b> is not a remote wakeup event.	1	<b>ConnectStatusChange</b> is a remote wakeup event.
0	<b>ConnectStatusChange</b> is not a remote wakeup event.							
1	<b>ConnectStatusChange</b> is a remote wakeup event.							
14:2				<b>Reserved</b>				
1	R	R/W	0x0	<p><b>OverCurrentIndicator</b></p> <p>This bit reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal.</p> <p>If per-port overcurrent protection is implemented this bit is always '0'</p>				
0	R/W	R	0x0	<p><b>(Read)LocalPowerStatus</b></p> <p>When read, this bit returns the LocalPowerStatus of the Root Hub. The Root Hub does not support the local power status feature; thus, this bit is always read as '0'.</p> <p><b>(Write)ClearGlobalPower</b></p> <p>When write, this bit is operated as the ClearGlobalPower. In global power mode (<b>PowerSwitchingMode=0</b>), This bit is written to '1' to turn off power to all ports (clear <b>PortPowerStatus</b>). In per-port power mode, it clears <b>PortPowerStatus</b> only on ports whose <b>PortPowerControlMask</b> bit is not set. Writing a '0' has no effect.</p>				



## 8.8.3.6.22. HcRhPortStatus Register

Offset: 0x454				Register Name: HcRhPortStatus
Default Value: 0x100				
Bit	Read/Write		Default	Description
	HCD	HC		
31:21	/	/	0x0	<b>Reserved</b>
20	R/W	R/W	0x0	<b>PortResetStatusChange</b>  This bit is set at the end of the 10-ms port reset signal. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>port reset is not complete</td> </tr> <tr> <td>1</td> <td>port reset is complete</td> </tr> </table>
0	port reset is not complete			
1	port reset is complete			
19	R/W	R/W	0x0	<b>PortOverCurrentIndicatorChange</b>  This bit is valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the <b>PortOverCurrentIndicator</b> bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>no change in <b>PortOverCurrentIndicator</b></td> </tr> <tr> <td>1</td> <td><b>PortOverCurrentIndicator</b> has changed</td> </tr> </table>
0	no change in <b>PortOverCurrentIndicator</b>			
1	<b>PortOverCurrentIndicator</b> has changed			
18	R/W	R/W	0x0	<b>PortSuspendStatusChange</b>  This bit is set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resynchronization delay. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. This bit is also cleared when <b>ResetStatusChange</b> is set.
				<table border="1"> <tr> <td>0</td> <td>resume is not completed</td> </tr> <tr> <td>1</td> <td>resume completed</td> </tr> </table>
0	resume is not completed			
1	resume completed			
17	R/W	R/W	0x0	<b>PortEnableStatusChange</b>  This bit is set when hardware events cause the <b>PortEnableStatus</b> bit to be cleared. Changes from HCD writes do not set this bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>no change in <b>PortEnableStatus</b></td> </tr> <tr> <td>1</td> <td>change in <b>PortEnableStatus</b></td> </tr> </table>
0	no change in <b>PortEnableStatus</b>			
1	change in <b>PortEnableStatus</b>			

				<p><b>ConnectStatusChange</b></p> <p>This bit is set whenever a connect or disconnect event occurs. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. If</p> <p><b>CurrentConnectStatus</b> is cleared when a <b>SetPortReset</b>, <b>SetPortEnable</b>, or <b>SetPortSuspend</b> write occurs, this bit is set to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected.</p> <table border="1"> <tr> <td>0</td> <td>no change in <b>PortEnableStatus</b></td> </tr> <tr> <td>1</td> <td>change in <b>PortEnableStatus</b></td> </tr> </table> <p>Note: If the <b>DeviceRemovable[NDP]</b> bit is set, this bit is set only</p>	0	no change in <b>PortEnableStatus</b>	1	change in <b>PortEnableStatus</b>
0	no change in <b>PortEnableStatus</b>							
1	change in <b>PortEnableStatus</b>							
16	R/W	R/W	0x0	after a Root Hub reset to inform the system that the device is attached.				
15:10	/	/	0x0	<b>Reserved</b>				
				<p>(read)<b>LowSpeedDeviceAttached</b></p> <p>This bit indicates the speed of the device attached to this port. When set, a Low Speed device is attached to this port. When clear, a Full Speed device is attached to this port. This field is valid only when the <b>CurrentConnectStatus</b> is set.</p> <table border="1"> <tr> <td>0</td> <td>full speed device attached</td> </tr> <tr> <td>1</td> <td>low speed device attached</td> </tr> </table> <p>(write)<b>ClearPortPower</b></p> <p>The HCD clears the <b>PortPowerStatus</b> bit by writing a '1' to this bit. Writing a '0' has no effect.</p>	0	full speed device attached	1	low speed device attached
0	full speed device attached							
1	low speed device attached							
9	R/W	R/W	-					
				<p>(read)<b>PortPowerStatus</b></p> <p>This bit reflects the port's power status, regardless of the type of power switching implemented. This bit is cleared if an overcurrent condition is detected. HCD sets this bit by writing <b>SetPortPower</b> or <b>SetGlobalPower</b>. HCD clears this bit by writing <b>ClearPortPower</b> or <b>ClearGlobalPower</b>. Which power control switches are enabled is determined by <b>PowerSwitchingMode</b> and <b>PortPortControlMask[NumberDownstreamPort]</b>. In global switching mode(<b>PowerSwitchingMode=0</b>), only <b>Set/ClearGlobalPower</b> controls</p>				
8	R/W	R/W	0x1					

				<p>this bit. In per-port power switching (<b>PowerSwitchingMode=1</b>), if the <b>PortPowerControlMask[NDP]</b> bit for the port is set, only</p> <p><b>Set/ClearPortPower</b> commands are enabled. If the mask is not set, only <b>Set/ClearGlobalPower</b> commands are enabled. When port power is disabled, <b>CurrentConnectStatus</b>, <b>PortEnableStatus</b>, <b>PortSuspendStatus</b>, and <b>PortResetStatus</b> should be reset.</p> <table border="1"> <tr> <td>0</td> <td>port power is off</td> </tr> <tr> <td>1</td> <td>port power is on</td> </tr> </table> <p>(write)<b>SetPortPower</b></p> <p>The HCD writes a '1' to set the <b>PortPowerStatus</b> bit. Writing a '0' has no effect.</p> <p>Note: This bit is always reads '1b' if power switching is not supported.</p>	0	port power is off	1	port power is on
0	port power is off							
1	port power is on							
7:5	/	/	0x0	<b>Reserved</b>				
4	R/W	R/W	0x0	<p>(read)<b>PortResetStatus</b></p> <p>When this bit is set by a write to <b>SetPortReset</b>, port reset signaling is asserted. When reset is completed, this bit is cleared when <b>PortResetStatusChange</b> is set. This bit cannot be set if <b>CurrentConnectStatus</b> is cleared.</p> <table border="1"> <tr> <td>0</td> <td>port reset signal is not active</td> </tr> <tr> <td>1</td> <td>port reset signal is active</td> </tr> </table> <p>(write)<b>SetPortReset</b></p> <p>The HCD sets the port reset signaling by writing a '1' to this bit. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this write does not set <b>PortResetStatus</b>, but instead sets <b>ConnectStatusChange</b>. This informs the driver that it attempted to reset a disconnected port.</p>	0	port reset signal is not active	1	port reset signal is active
0	port reset signal is not active							
1	port reset signal is active							
3	R/W	R/W	0x0	<p>(read)<b>PortOverCurrentIndicator</b></p> <p>This bit is only valid when the Root Hub is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is not supported, this bit is set to 0. If cleared, all power operations are normal for this port. If set, an overcurrent</p>				

				<p>condition exists on this port. This bit always reflects the overcurrent input signal.</p> <table border="1"> <tr> <td>0</td> <td>no overcurrent condition.</td> </tr> <tr> <td>1</td> <td>overcurrent condition detected.</td> </tr> </table> <p>(write)<b>ClearSuspendStatus</b></p> <p>The HCD writes a '1' to initiate a resume. Writing a '0' has no effect. A resume is initiated only if <b>PortSuspendStatus</b> is set.</p>	0	no overcurrent condition.	1	overcurrent condition detected.
0	no overcurrent condition.							
1	overcurrent condition detected.							
2	R/W	R/W	0x0	<p>(read)<b>PortSuspendStatus</b></p> <p>This bit indicates the port is suspended or in the resume sequence. It is set by a <b>SetSuspendState</b> write and cleared when <b>PortSuspendStatusChange</b> is set at the end of the resume interval. This bit cannot be set if <b>CurrentConnectStatus</b> is cleared. This bit is also cleared when <b>PortResetStatusChange</b> is set at the end of the port reset or when the HC is placed in the USBRESUME state. If an upstream resume is in progress, it should propagate to the HC.</p> <table border="1"> <tr> <td>0</td> <td>port is not suspended</td> </tr> <tr> <td>1</td> <td>port is suspended</td> </tr> </table> <p>(write)<b>SetPortSuspend</b></p> <p>The HCD sets the <b>PortSuspendStatus</b> bit by writing a '1' to this bit. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this write does not set <b>PortSuspendStatus</b>; instead it sets <b>ConnectStatusChange</b>. This informs the driver that it attempted to suspend a disconnected port.</p>	0	port is not suspended	1	port is suspended
0	port is not suspended							
1	port is suspended							
1	R/W	R/W	0x0	<p>(read)<b>PortEnableStatus</b></p> <p>This bit indicates whether the port is enabled or disabled. The Root Hub may clear this bit when an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected. This change also causes <b>PortEnabledStatusChange</b> to be set. HCD sets this bit by writing <b>SetPortEnable</b> and clears it by writing <b>ClearPortEnable</b>. This bit cannot be set when <b>CurrentConnectStatus</b> is cleared. This bit is also set, if not already, at the completion of a port reset when <b>ResetStatusChange</b> is set or port suspend when</p>				

				<p><b>SuspendStatusChange</b> is set.</p> <table border="1"> <tr> <td>0</td> <td>port is disabled</td> </tr> <tr> <td>1</td> <td>port is enabled</td> </tr> </table> <p>(write)<b>SetPortEnable</b></p> <p>The HCD sets <b>PortEnableStatus</b> by writing a '1'. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this write does not set <b>PortEnableStatus</b>, but instead sets <b>ConnectStatusChange</b>. This informs the driver that it attempted to enable a disconnected Port.</p>	0	port is disabled	1	port is enabled
0	port is disabled							
1	port is enabled							
0	R/W	R/W	0x0	<p>(read)<b>CurrentConnectStatus</b></p> <p>This bit reflects the current state of the downstream port.</p> <table border="1"> <tr> <td>0</td> <td>No device connected</td> </tr> <tr> <td>1</td> <td>Device connected</td> </tr> </table> <p>(write)<b>ClearPortEnable</b></p> <p>The HCD writes a '1' to clear the PortEnableStatus bit. Writing '0' has no effect. The CurrentConnectStatus is not affected by any write.</p> <p>Note: This bit is always read '1' when the attached device is nonremovable(DviceRemoveable[NumberDownstreamPort]).</p>	0	No device connected	1	Device connected
0	No device connected							
1	Device connected							

### 8.8.3.7. HCI Interface Control and Status Register

#### 8.8.3.7.1. HCI Interface Control Register

Offset: 0x800			Register Name: HCI_ICR
Bit	Read/ Write	Default /Hex	Description

31:21	/	/	Reserved.
20	R/W	0	EHCI HS force  Set 1 to this field force the ehci enter the high speed mode during bus reset.  This field only valid when the bit 1 is set.
19:18	/	/	/
17	R/W	0	HSIC Connect detect  1 in this field enable the hsic phy to detect device connect pulse on the bus.  This field only valid when the bit 1 is set.
16	R/W	0	HSIC Connect Interrupt Enable  Enable the HSIC connect interrupt.  This field only valid when the bit 1 is set.
15:13	/	/	/
12	/	/	/
11	R/W	0	AHB Master interface INCR16 enable  1: Use INCR16 when appropriate  0: do not use INCR16,use other enabled INCRX or unspecified length burst INCR
10	R/W	0	AHB Master interface INCR8 enable  1: Use INCR8 when appropriate  0: do not use INCR8,use other enabled INCRX or unspecified length burst INCR
9	R/W	0	AHB Master interface burst type INCR4 enable  1: Use INCR4 when appropriate  0: do not use INCR4,use other enabled INCRX or unspecified length burst INCR
8	R/W	0	AHB Master interface INCRX align enable  1: start INCRx burst only on burst x-align address  0: Start burst on any double word boundary

			Note: This bit must enable if any bit of 11:9 is enabled
7:2	/	/	Reserved
1	R/W	0	HSIC  0:/  1:HSIC  This meaning is only valid when the controller is HCI1.
0	R/W	0	ULPI bypass enable。  1: Enable UTMI interface, disable ULPI interface(SP used utmi interface)  0: Enable ULPI interface, disable UTMI interface

#### 8.8.3.7.2. HSIC status Register

<b>Offset: 0x804</b>			<b>Register Name: HSIC_STATUS</b>
			<b>Default Value: 0x00</b>
Bit	Read/Write	Default	Description
31:17	/	/	/
			HSIC Connect Status  1 in this field indicates a device connect pulse being detected on the bus. This field only valid when the EHCI HS force bit and the HSIC Phy Select bit is set.  When the HSIC Connect Interrupt Enable is set, 1 in this bit will generate an interrupt to the system.
16	R/W	0	This register is valid on HCI1.
15:0	/	/	/

### 8.8.3.8. HCI Clock Control Register

#### 8.8.3.8.1. HCI SIE Control Register

Offset: 0x00A08000			Register Name: HCI_SCR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:20	/	/	/
19	R/W	0	<b>USB HCI2 Module Reset</b> 0: assert 1: de-assert
18	R/W	0	<b>USB HCI1 Module Reset</b> 0: assert 1: de-assert
17	R/W	0	<b>USB HCI0 Module Reset</b> 0: assert 1: de-assert
16:7	/	/	/
6	R/W	0	<b>USB OHCI2 Special Clock(12M and 48M) Gating</b> 0: mask 1: pass
5	R/W	0	<b>USB HCI2 AHB Gating</b> 0: mask 1: pass
4	/	/	/
3	R/W	0	<b>USB HCI1 AHB Gating</b> 0: mask 1: pass
2	R/W	0	<b>USB OHCI0 Special Clock(12M and 48M) Gating</b>



			0: mask 1: pass
1	R/W	0	<b>USB HClO AHB Gating</b> 0: mask 1: pass
0	/	/	/

#### 8.8.3.8.2. HCI PHY Control Register

Offset: 0x00A08004			Register Name: HCI_PCR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:22	/	/	/
			<b>HCI2_UTMIPHY_RST</b>
			0: Assert 1: De-assert
21	R/W	0	
20:19	/	/	/
			<b>HCI1_HSIC_RST</b>
			0: Assert 1: De-assert
18	R/W	0	
			<b>HClO_PHY_RST</b>
			0: Assert 1: De-assert
17	R/W	0	
16:11	/	/	/
			<b>12M_GATING_HCI1_HSIC</b>
			0: Clock is OFF 1: Clock is ON
10	R/W	0	
9:6	/	/	/
			<b>SCLK_GATING_HCI2_UTMIPHY</b>
5	R/W	0	

			0: Clock is OFF 1: Clock is ON
4	R/W	0	<b>480M_GATING_HCI2_HSIC</b> 0: Clock is OFF 1: Clock is ON
3	/	/	/
2	R/W	0	<b>480M_GATING_HCI1_HSIC</b> 0: Clock is OFF 1: Clock is ON
1	R/W	0	<b>SCLK_GATING_HCIO_PHY</b> 0: Clock is OFF 1: Clock is ON
0	/	/	/

### 8.8.3.9. USB Host Special Requirement

#### 8.8.3.9.1. USB Host Clock Requirement

Name	Description
HCLK	System clock (provided by AHB bus clock). This clock needs to be >30MHz.
CLK60M	Clock from PHY for HS SIE, is constant to be 60MHz.
CLK48M	Clock from PLL for FS/LS SIE, is constant to be 48MHz.

## 8.9. Digital Audio Interface

### 8.9.1. Overview

The Digital Audio Interface Controller has been designed to transfer streaming audio-data between the system memory and the codec chip. The controller supports standard I2S format, Left-justified Mode format, Right-justified Mode format and PCM Mode format. Additionally, the PCM/I2S will also support multi-slot.

The digital audio interface controller includes the following features:

- Compliant with standard Philips Inter-IC sound (I2S) bus specification
- Compliant with Left-justified, Right-justified, PCM mode, and PCM/I2S multi-slot format
- Support different sample period width in each interface when using LRCK and LRCKR at the same time
- Support full-duplex synchronous work mode
- Support Master / Slave mode
- Support adjustable interface voltage
- Support clock up to 100MHz
- Support adjustable audio sample resolution from 8-bit to 32-bit.
- Support up to 8 slots which has adjustable width from 8-bit to 32-bit.
- Support sample rate from 8KHz to 192KHz
- Support up to 4 data output pin
- Support 8-bits u-law and 8-bits A-law companded sample
- One 128 depth x 32-bit width FIFO for data transmit, one 64 depth x 32-bit width FIFO for data receive
- Support programmable PCM frame width: 1 BCLK width (short frame) and 2 BCLKs width (long frame)
- Programmable FIFO thresholds
- Interrupt and DMA Support
- Support loopback mode for test

## 8.9.2. Signal Description

### 8.9.2.1. Digital Audio Interface Pin List

Signal Name	Direction(M)	Description	Pin
MCLK	O	Digital Audio 0 MCLK Output	PM10
BCLK	I/O	Digital Audio 0 Sample Rate Clock/Sync	PM11
LRCK	I/O	Digital Audio 0 Serial Clock	PM12
SDI	I	Digital Audio 0 Serial Data Output	PM13
SDO	O	Digital Audio 0 Serial Data Output	PM14/5/6/7
LRCKR	I/O	Digital Audio 0 Sample Rate Clock/Sync Input	PM4

### 8.9.2.2. Digital Audio Interface Clock Source and Frequency

	Description
Audio_PLL	24.576Mhz or 22.5792Mhz generated by AUDIO-PLL to produce 48KHz or 44.1KHz serial frequency

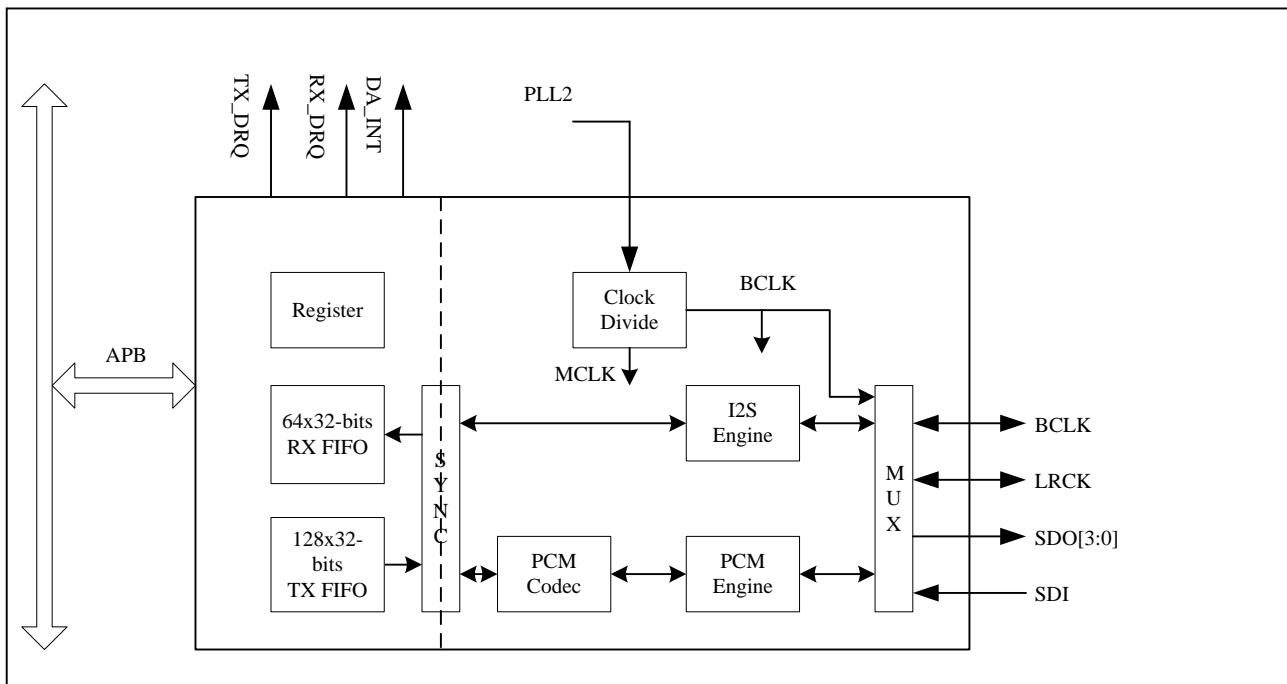
### 8.9.3. Functionalities Description

#### 8.9.3.1. Typical Applications

The I2S and PCM which provides a serial bus interface for stereo and multichannel audio data. This interface is most commonly used by consumer audio market, including compact disc, digital audio tape, digital sound processors, and digital TV-sound.

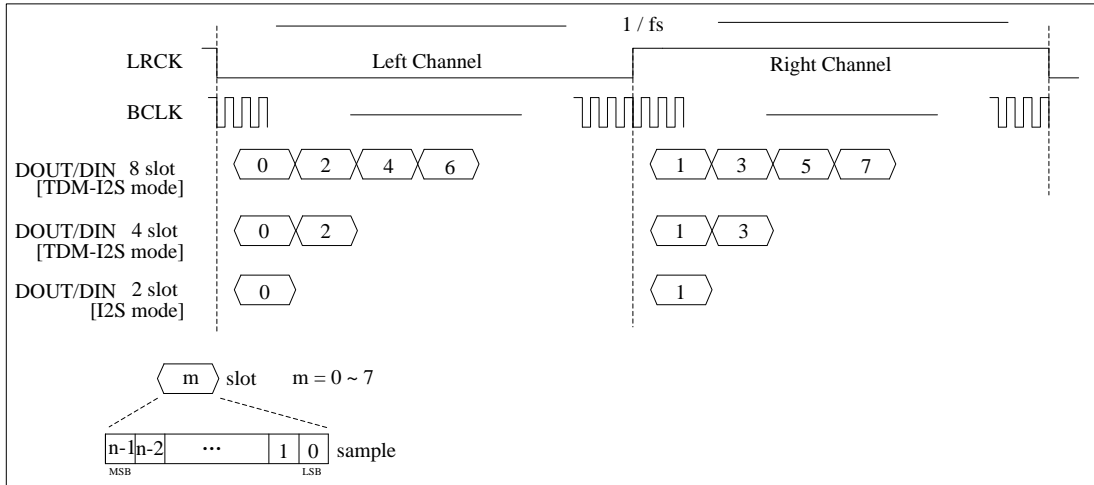
#### 8.9.3.2. Functional Block Diagram

The Digital Audio Interface block diagram is shown below:

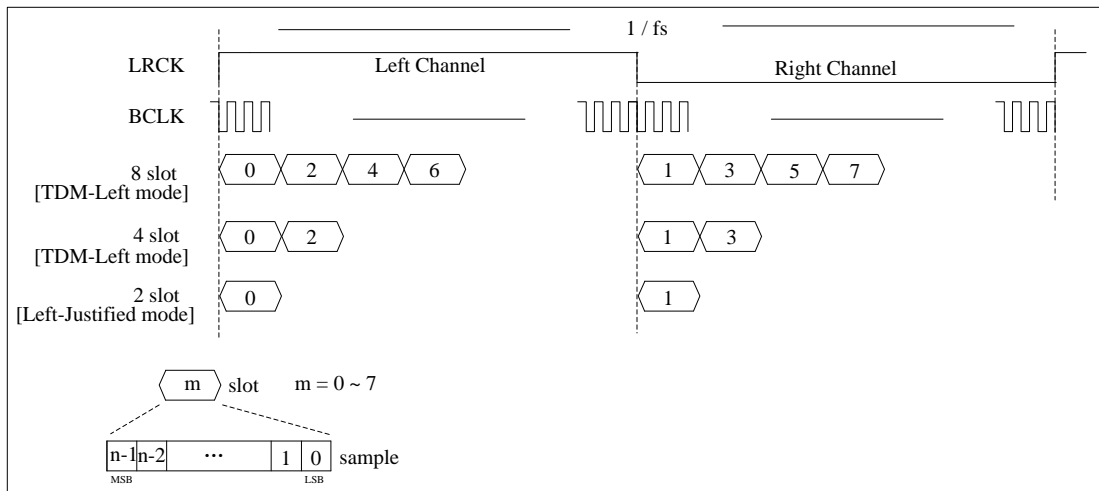


Digital Audio Interface System Block Diagram

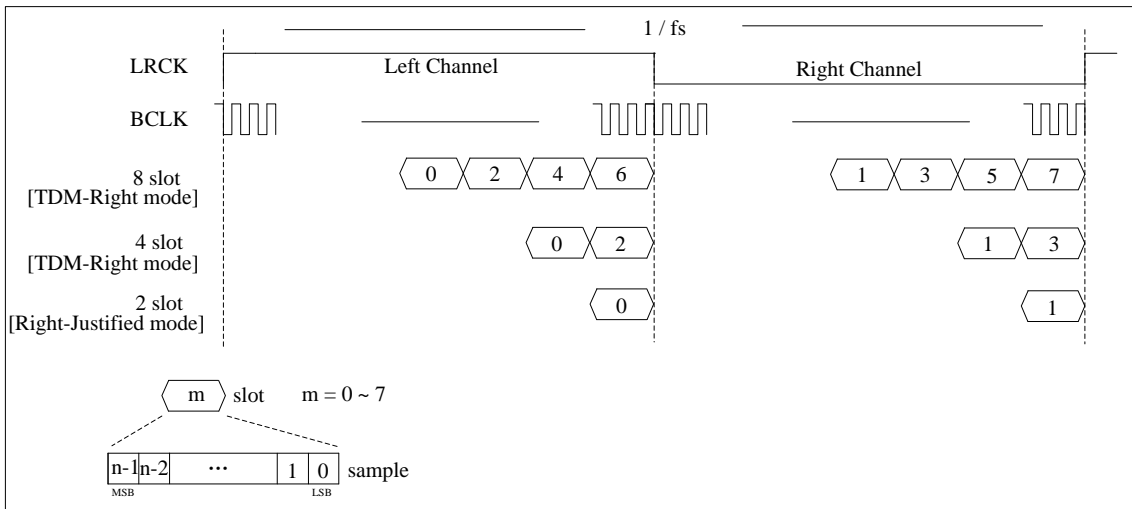
### 8.9.4. Timing Diagram



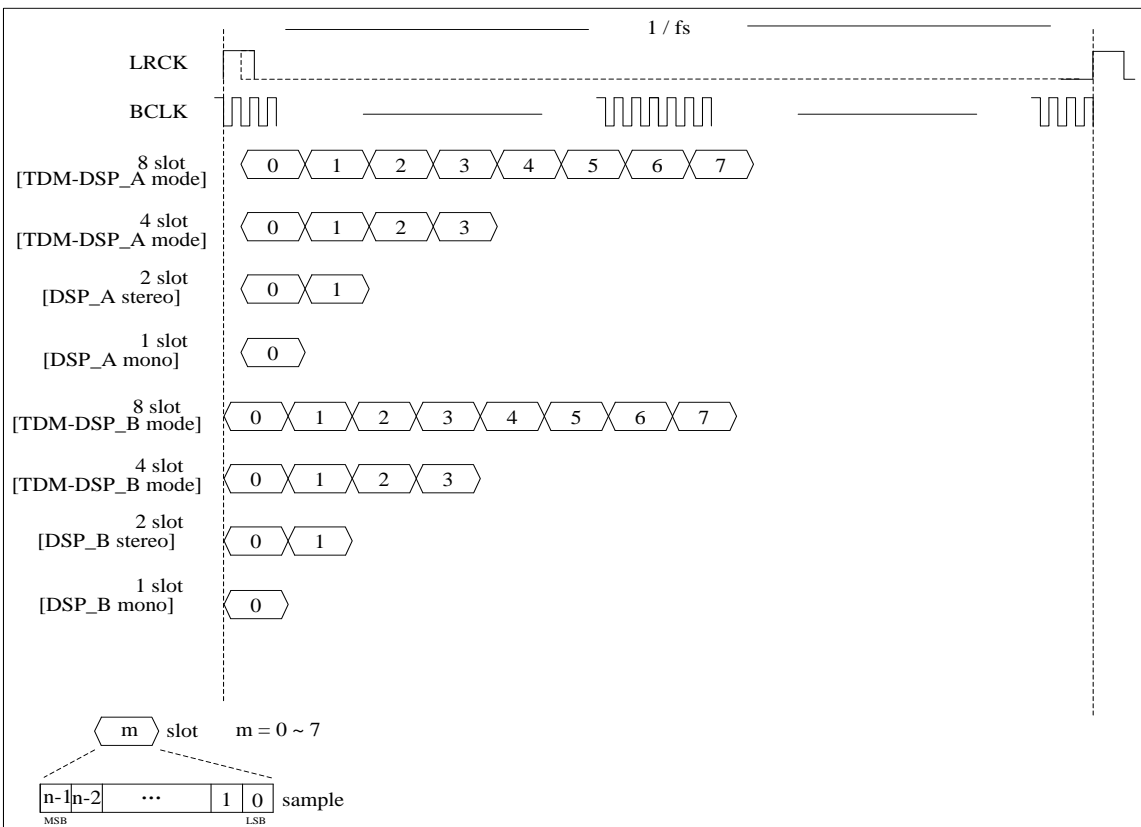
Timing diagram for I2S and Multi-Slot mode



Timing diagram for Left-justified and Multi-Slot mode



Timing diagram for Right-justified and Multi-Slot mode



Timing diagram for PCM and Multi-Slot PCM mode

### 8.9.5. Operation Modes

The software operation of the PCM/I2S is divided into five steps: system setup, PCM/I2S initialization, the channel setup, DMA setup and Enable/Disable module. These five setup are described in detail in the following sections.

#### 8.9.5.1. System setup and PCM/I2S initialization

The first step In the system setup is properly programming the GPIO. The PCM/I2S port is a multiplex pin and its function can be found in the pin multiplex specification. The clock source for the PCM/I2S should be followed. At first you must reset the audio PLL though the PLL\_ENABLE bit of PLL\_AUDIO\_CTRL\_REG in the CCU. The second step, you must setup the frequency of the audio pll in the PLL\_AUDIO\_CTRL\_REG. After that, you must open the PCM/I2S gating though the R\_DAUDIO1\_CLK\_REG when you checkout that the LOCK bit of PLL\_AUDIO\_CTRL\_REG become 1. At last, you must reset the PCM/I2S the APBS\_SOFT\_RST\_REG's bit[18] and open the PCM/I2S bus gating in the APBS\_CLK\_GATING\_REG's bit[18].

After the system setup, the register of PCM/I2S can be setup. At first, you should initialization the PCM/I2S. You should closed the globe enable bit(DA\_CTL[0]) , TX enable bit(DA\_CTL[2]) and RX enable bit(DA\_CTL[1]) by write 0 to it. After that, you must clear the TX/RX FIFO by write 0 to register DA\_FCTL[25:24]. At last, you can clear the TX/RX FIFO counter by write 0 to DA\_TXCNT/DA\_RXCNT.

#### 8.9.5.2. The channel setup and DMA setup

At first, you can setup the PCM/I2S of mater and slave by set DA\_CTL[18:16]. The configuration can be referred to the the protocol of PCM/I2S. Then, you can set the translation mode, the sample precision, the wide of slot, the frame mode and the trigger level. The register set can be found in the spec.

The PCM/I2S support three method to transfer the data. The most common way is DMA, the set of DMA can be found in the DMA part. In this module, you just to enable the DRQ. Because the PCM/I2S is in the CPUS domain, you can only use the rDMA which is also in the CPUS domain.

#### 8.9.5.3. Enable and disable the PCM/I2S

To enable the function, you can enable TX/RX by write the DA\_CTL[2:1]. After that, you must enable PCM/I2S by write the Globe Enable bit to 1 in the DA\_CTL. The disable process is write the Globe Enable to 0.



### 8.9.6. Digital Audio Interface Register List

Module Name	Base Address
DAUDIO1	0x08006000

Register Name	Offset	Description
DA_CTL	0x00	Digital Audio Control Register
DA_FMT0	0x04	Digital Audio Format Register 0
DA_FMT1	0x08	Digital Audio Format Register 1
DA_ISTA	0x0C	Digital Audio Interrupt Status Register
DA_RXFIFO	0x10	Digital Audio RX FIFO Register
DA_FCTL	0x14	Digital Audio FIFO Control Register
DA_FSTA	0x18	Digital Audio FIFO Status Register
DA_INT	0x1C	Digital Audio DMA & Interrupt Control Register
DA_TXFIFO	0x20	Digital Audio TX FIFO Register
DA_CLKD	0x24	Digital Audio Clock Divide Register
DA_TXCNT	0x28	Digital Audio TX Sample Counter Register
DA_RXCNT	0x2C	Digital Audio RX Sample Counter Register
DA_CHCFG	0x30	Digital Audio Channel Configuration register
DA_CHCFG	0x34	Digital Audio Channel Configuration register
DA_TX1CHSEL	0x38	Digital Audio TX1 Channel Select Register
DA_TX2CHSEL	0x3C	Digital Audio TX2 Channel Select Register
DA_TX3CHSEL	0x40	Digital Audio TX3 Channel Select Register
DA_TX0CHMAP	0x44	Digital Audio TX0 Channel Mapping Register
DA_TX0CHMAP	0x48	Digital Audio TX1 Channel Mapping Register
DA_TX0CHMAP	0x4C	Digital Audio TX2 Channel Mapping Register
DA_TX0CHMAP	0x50	Digital Audio TX3 Channel Mapping Register
DA_RXCHSEL	0x54	Digital Audio RX Channel Select register
DA_RXCHMAP	0x58	Digital Audio RX Channel Mapping Register

## 8.9.7. Digital Audio Interface Register Description

### 8.9.7.1. Digital Audio Control Register

Offset: 0x00			Register Name: DA_CTL Default Value: 0x0006_0000
Bit	Read/Write	Default	Description
31:19	/	/	/
18	R/W	1	BCLK_OUT 0: input 1: output
17	R/W	1	LRCK_OUT 0: input 1: output
16	R/W	0	LRCKR_OUT 0: input 1: output
15:12	/	/	/
11	R/W	0	SDO3_EN 0: Disable, Hi-Z state 1: Enable
10	R/W	0	SDO2_EN 0: Disable, Hi-Z state 1: Enable
9	R/W	0	SDO1_EN 0: Disable, Hi-Z state 1: Enable
8	R/W	0	SDO0_EN 0: Disable, Hi-Z state 1: Enable
7	/	/	/
6	R/W	0	OUT Mute 0: normal transfer 1: force DOUT to output 0
5:4	R/W	0	MODE_SEL Mode Selection 0: PCM mode (offset 0: DSP_B; offset 1: DSP_A) 1: Left mode (offset 0: LJ mode; offset 1: I2S mode) 2: Right-Justified mode 3: Reserved
3	R/W	0	LOOP Loop back test 0: Normal mode

			1: Loop back test When set '1', connecting the SDO0 with the SDI
2	R/W	0	TXEN Transmitter Block Enable 0: Disable 1: Enable
1	R/W	0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0	GEN Globe Enable A disable on this bit overrides any other block or channel enables. 0: Disable 1: Enable

### 8.9.7.2. Digital Audio Format Register 0

Offset: 0x04			Register Name: DA_FMT0 Default Value: 0x0000_0033
Bit	Read/Write	Default	Description
31	R/W	0	SDI_SYNC_SEL 0: SDI use LRCK 1: SDI use LRCKR
30	R/W	0	LRCK_WIDTH (only apply in PCM mode ) LRCK width 0: LRCK = 1 BCLK width (short frame) 1: LRCK = 2 BCLK width (long frame)
29:20	R/W	0	LRCKR_PERIOD It is used to program the number of BCLKs per channel of sample frame. This value is interpreted as follow: PCM mode: Number of BCLKs within (Left + Right) channel width I2S / Left-Justified / Right-Justified mode: Number of BCLKs within each individual channel width (Left or Right) N+1 For example: n = 7: 8 BCLK width ... n = 1023: 1024 BCLKs width
19	R/W	0	LRCK_POLARITY/LRCKR_POLARITY When apply in I2S / Left-Justified / Right-Justified mode: 0: Left channel when LRCK is low

			1: Left channel when LRCK is high When apply in PCM mode: 0: PCM LRCK/LRCKR asserted at the negative edge 1: PCM LRCK/LRCKR asserted at the positive edge
18	/	/	/
17:8	R/W	0	LRCK_PERIOD It is used to program the number of BCLKs per channel of sample frame. This value is interpreted as follow: PCM mode: Number of BCLKs within (Left + Right) channel width I2S / Left-Justified / Right-Justified mode: Number of BCLKs within each individual channel width (Left or Right) N+1 For example: n = 7: 8 BCLK width ... n = 1023: 1024 BCLKs width
7	R/W	0	BCLK_POLARITY 0: normal mode, negative edge drive and positive edge sample 1: invert mode, positive edge drive and negative edge sample
6:4	R/W	3	SR Sample Resolution 0: Reserved 1: 8-bit 2: 12-bit 3: 16-bit 4: 20-bit 5: 24-bit 6: 28-bit 7: 32-bit
3	R/W	0	EDGE_TRANSFER 0: SDO drive data and SDI sample data at the different BCLK edge 1: SDO drive data and SDI sample data at the same BCLK edge BCLK_POLARITY = 0, use negative edge BCLK_POLARITY = 1, use positive edge
2:0	R/W	0x3	SW Slot Width Select 0: Reserved 1: 8-bit 2: 12-bit 3: 16-bit 4: 20-bit 5: 24-bit 6: 28-bit 7: 32-bit

### 8.9.7.3. Digital Audio Format Register 1

Offset: 0x08			Register Name: DA_FMT1 Default Value: 0x0000_0030
Bit	Read/Write	Default	Description
31:8	/	/	
7	R/W	0	RX MLS MSB / LSB First Select 0: MSB First 1: LSB First
6	R/W	0	TX MLS MSB / LSB First Select 0: MSB First 1: LSB First
5:4	R/W	3	SEXT Sign Extend in slot [sample resolution < slot width] 0: Zeros or audio gain padding at LSB position 1: Sign extension at MSB position 2: Reserved 3: Transfer 0 after each sample in each slot
3:2	R/W	0	RX_PDM PCM Data Mode 0: Linear PCM 1: reserved 2: 8-bits u-law 3: 8-bits A-law
1:0	R/W	0	TX_PDM PCM Data Mode 0: Linear PCM 1: reserved 2: 8-bits u-law 3: 8-bits A-law

### 8.9.7.4. Digital Audio Interrupt Status Register

Offset: 0x0C			Register Name: DA_ISTA Default Value: 0x0000_0010
Bit	Read/Write	Default	Description
31:7	/	/	/
6	R/W	0	TXU_INT

			TX FIFO Under run Pending Interrupt 0: No Pending Interrupt 1: FIFO Under run Pending Interrupt Write 1 to clear this interrupt
5	R/W	0	TXO_INT TX FIFO Overrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Overrun Pending Interrupt Write '1' to clear this interrupt
4	R/W	1	TXE_INT TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt when data in TX FIFO are less than TX trigger level Write '1' to clear this interrupt or automatic clear if interrupt condition fails.
3	/	/	/
2	R/W	0	R XU_INT RX FIFO Under run Pending Interrupt 0: No Pending Interrupt 1:FIFO Under run Pending Interrupt Write 1 to clear this interrupt
1	R/W	0	R XO_INT RX FIFO Overrun Pending Interrupt 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt
0	R/W	0	R XA_INT RX FIFO Data Available Pending Interrupt 0: No Pending IRQ 1: Data Available Pending IRQ when data in RX FIFO are more than RX trigger level Write '1' to clear this interrupt or automatic clear if interrupt condition fails.

#### 8.9.7.5. Digital Audio RX FIFO register

<b>Offset: 0x10</b>			<b>Register Name: DA_RXFIFO</b> <b>Default Value: 0x0000_0000</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:0	R	0	RX_DATA RX Sample

			Host can get one sample by reading this register. The left channel sample data is first and then the right channel sample.
--	--	--	--

### 8.9.7.6. Digital Audio FIFO Control Register

Offset: 0x14			Register Name: DA_FCTL Default Value: 0x0004_00F0
Bit	Read/Write	Default	Description
31:26	/	/	/
25	R/W	0	FTX Write '1' to flush TX FIFO, self clear to '0'.
24	R/W	0	FRX Write '1' to flush RX FIFO, self clear to '0'.
23:19	/	/	/
18:12	R/W	0x40	TXTL TX FIFO Empty Trigger Level Interrupt and DMA request trigger level for TXFIFO normal condition Trigger Level = TXTL
11:10	/	/	/
9:4	R/W	0xF	RXTL RX FIFO Trigger Level Interrupt and DMA request trigger level for RXFIFO normal condition Trigger Level = RXTL + 1
3	/	/	/
2	R/W	0	TXIM TX FIFO Input Mode (Mode 0, 1) 0: Valid data at the MSB of TXFIFO register 1: Valid data at the LSB of TXFIFO register Example for 20-bits transmitted audio sample: Mode 0: FIFO_I[31:0] = {APB_WDATA[31:12], 12'h0} Mode 1: FIFO_I[31:0] = {APB_WDATA[19:0], 12'h0}
1:0	R/W	0	RXOM RX FIFO Output Mode (Mode 0, 1, 2, 3) 00: Expanding '0' at LSB of DA_RXFIFO register. 01: Expanding received sample sign bit at MSB of DA_RXFIFO register. 10: Truncating received samples at high half-word of DA_RXFIFO register and low half-word of DA_RXFIFO register is filled by '0'. 11: Truncating received samples at low half-word of DA_RXFIFO register and high half-word of DA_RXFIFO register is expanded by its sign bit. Example for 20-bits received audio sample: Mode 0: APB_RDATA[31:0] = {FIFO_O[31:12], 12'h0} Mode 1: APB_RDATA [31:0] = {12{FIFO_O[31]}, FIFO_O[31:12]}

		Mode 2: APB_RDATA [31:0] = {FIFO_O[31:16], 16'h0} Mode 3: APB_RDATA [31:0] = {16{FIFO_O[31], FIFO_O[31:16]}
--	--	--

### 8.9.7.7. Digital Audio FIFO Status Register

Offset: 0x18			Register Name: DA_FSTA Default Value: 0x1080_0000
Bit	Read/Write	Default	Description
31:29	/	/	/
28	R	1	TXE TX FIFO Empty 0: No room for new sample in TX FIFO 1: More than one room for new sample in TX FIFO (>= 1 word)
27:24	/	/	/
23:16	R	0x80	TXE_CNT TX FIFO Empty Space Word Counter
15:9	/	/	/
8	R	0	RXA RX FIFO Available 0: No available data in RX FIFO 1: More than one sample in RX FIFO (>= 1 word)
7	/	/	/
6:0	R	0	RXA_CNT RX FIFO Available Sample Word Counter

### 8.9.7.8. Digital Audio DMA & Interrupt Control Register

Offset: 0x1C			Register Name: DA_INT Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:8	/	/	/
7	R/W	0	TX_DRQ TX FIFO Empty DRQ Enable 0: Disable 1: Enable
6	R/W	0	TXUI_EN TX FIFO Under run Interrupt Enable 0: Disable 1: Enable
5	R/W	0	TXOI_EN TX FIFO Overrun Interrupt Enable



			0: Disable 1: Enable When set to '1', an interrupt happens when writing new audio data if TX FIFO is full.
4	R/W	0	TXEI_EN TX FIFO Empty Interrupt Enable 0: Disable 1: Enable
3	R/W	0	RX_DRQ RX FIFO Data Available DRQ Enable 0: Disable 1: Enable When set to '1', RXFIFO DMA Request line is asserted if Data is available in RX FIFO.
2	R/W	0	RXUI_EN RX FIFO Under run Interrupt Enable 0: Disable 1: Enable
1	R/W	0	RXOI_EN RX FIFO Overrun Interrupt Enable 0: Disable 1: Enable
0	R/W	0	RXAI_EN RX FIFO Data Available Interrupt Enable 0: Disable 1: Enable

#### 8.9.7.9. Digital Audio TX FIFO register

<b>Offset: 0x20</b>			<b>Register Name: DA_TXFIFO</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	W	0	TX_DATA TX Sample Transmitting left, right channel sample data should be written this register one by one. The left channel sample data is first and then the right channel sample.

#### 8.9.7.10. Digital Audio Clock Divide Register

<b>Offset: 0x24</b>	<b>Register Name: DA_CLKD</b>
---------------------	-------------------------------

			Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:9	/	/	/
8	R/W	0	MCLKO_EN 0: Disable MCLK Output 1: Enable MCLK Output Notes: Whether in Slave or Master mode, when this bit is set to 1, MCLK should be output.
7:4	R/W	0	BCLKDIV BCLK Divide Ratio from PLL2 0: reserved 1: Divide by 1 2: Divide by 2 3: Divide by 4 4: Divide by 6 5: Divide by 8 6: Divide by 12 7: Divide by 16 8: Divide by 24 9: Divide by 32 10: Divide by 48 11: Divide by 64 12: Divide by 96 13: Divide by 128 14: Divide by 176 15: Divide by 192
3:0	R/W	0	MCLKDIV MCLK Divide Ratio from PLL2 Output 0: reserved 1: Divide by 1 2: Divide by 2 3: Divide by 4 4: Divide by 6 5: Divide by 8 6: Divide by 12 7: Divide by 16 8: Divide by 24 9: Divide by 32 10: Divide by 48 11: Divide by 64 12: Divide by 96 13: Divide by 128 14: Divide by 176 15: Divide by 192

**8.9.7.11. Digital Audio TX Counter register**

<b>Offset: 0x28</b>			<b>Register Name: DA_TXCNT</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	TX_CNT TX Sample Counter The audio sample number of sending into TXFIFO. When one sample is put into TXFIFO by DMA or by host IO, the TX sample counter register increases by one. The TX sample counter register can be set to any initial value at any time. After been updated by the initial value, the counter register should count on base of this initial value.

**8.9.7.12. Digital Audio RX Counter register**

<b>Offset: 0x2C</b>			<b>Register Name: DA_RXCNT</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	RX_CNT RX Sample Counter The audio sample number of writing into RXFIFO. When one sample is written by Digital Audio Engine, the RX sample counter register increases by one. The RX sample counter register can be set to any initial value at any time. After been updated by the initial value, the counter register should count on base of this initial value.

**8.9.7.13. Digital Audio Channel Configuration register**

<b>Offset: 0x30</b>			<b>Register Name: DA_CHCFG</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:10	/	/	/
9	R/W	0	TX_SLOT_HIZ 0: normal mode for the last half cycle of BCLK in the slot 1: turn to hi-z state for the last half cycle of BCLK in the slot
8	R/W	0	TXn_STATE 0: transfer level 0 when not transferring slot 1: turn to hi-z state when not transferring slot

7	/	/	/
6:4	R/W	0	RX_SLOT_NUM RX Channel/Slot Number which between CPU/DMA and FIFO 0: 1 channel or slot ... 7: 8 channels or slots
3	/	/	/
2:0	R/W	0	TX_SLOT_NUM TX Channel/Slot Number which between CPU/DMA and FIFO 0: 1 channel or slot ... 7: 8 channels or slots

#### 8.9.7.14. Digital Audio TXn Channel Select register

Offset: 0x34 + n*4 (n = 0, 1, 2, 3)			Register Name: DA_TXnCHSEL Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:14	/	/	/
13:12	R/W	0	TXn_OFFSET TXn offset tune, TXn data offset to LRCK 0: no offset n: data is offset by n BCLKs to LRCK
11:4	R/W	0	TXn_CHEN TXn Channel (slot) enable, bit[11:4] refer to slot [7:0]. When one or more slot(s) is(are) disabled, the affected slot(s) is(are) set to disable state 0: disable 1: enable
3	/	/	/
2:0	R/W	0	TXn_CHSEL TXn Channel (slot) number Select for each output 0: 1 channel / slot ... 7: 8 channels / slots

#### 8.9.7.15. Digital Audio TXn Channel Mapping Register

Offset: 0x44 + n*4 (n = 0, 1, 2, 3)			Register Name: DA_TXnCHMAP Default Value: 0x0000_0000
Bit	Read/Write	Default	Description

31	/	/	/
30:28	R/W	0	TXn_CH7_MAP TXn Channel7 Mapping 0: 1st sample ... 7: 8th sample
27	/	/	/
26:24	R/W	0	TXn_CH6_MAP TXn Channel6 Mapping 0: 1st sample ... 7: 8th sample
23	/	/	/
22:20	R/W	0	TXn_CH5_MAP TXn Channel5 Mapping 0: 1st sample ... 7: 8th sample
19	/	/	/
18:16	R/W	0	TXn_CH4_MAP TXn Channel4 Mapping 0: 1st sample ... 7: 8th sample
15	/	/	/
14:12	R/W	0	TXn_CH3_MAP TXn Channel3 Mapping 0: 1st sample ... 7: 8th sample
11	/	/	/
10:8	R/W	0	TXn_CH2_MAP TXn Channel2 Mapping 0: 1st sample ... 7: 8th sample
7	/	/	/
6:4	R/W	0	TXn_CH1_MAP TXn Channel1 Mapping 0: 1st sample ... 7: 8th sample
3	/	/	/

2:0	R/W	0	TXn_CH0_MAP TXn Channel0 Mapping 0: 1st sample ... 7: 8th sample
-----	-----	---	--

#### 8.9.7.16. Digital Audio RX Channel Select register

Offset: 0x54			Register Name: DA_RXCHSEL Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31:14	/	/	/
13:12	R/W	0	RX_OFFSET RX offset tune, RX data offset to LRCK 0: no offset n: data is offset by n BCLKs to LRCK
11:3	/	/	
2:0	R/W	0	RX_CHSEL RX Channel (slot) number Select for input 0: 1 channel / slot ... 7: 8 channels / slots

#### 8.9.7.17. Digital Audio RX Channel Mapping Register

Offset: 0x58			Register Name: DA_RXCHMAP Default Value: 0x0000_0000
Bit	Read/Write	Default	Description
31	/	/	/
30:28	R/W	0	RX_CH7_MAP RX Channel7 Mapping 0: 1st sample ... 7: 8th sample
27	/	/	/
26:24	R/W	0	RX_CH6_MAP RX Channel6 Mapping 0: 1st sample ... 7: 8th sample
23	/	/	/

22:20	R/W	0	RX_CH5_MAP RX Channel5 Mapping 0: 1st sample ... 7: 8th sample
19	/	/	/
18:16	R/W	0	RX_CH4_MAP RX Channel4 Mapping 0: 1st sample ... 7: 8th sample
15	/	/	/
14:12	R/W	0	RX_CH3_MAP RX Channel3 Mapping 0: 1st sample ... 7: 8th sample
11	/	/	/
10:8	R/W	0	RX_CH2_MAP RX Channel2 Mapping 0: 1st sample ... 7: 8th sample
7	/	/	/
6:4	R/W	0	RX_CH1_MAP TX Channel1 Mapping 0: 1st sample ... 7: 8th sample
3	/	/	/
2:0	R/W	0	RX_CH0_MAP RX Channel0 Mapping 0: 1st sample ... 7: 8th sample

## 8.10. Transport Stream

### 8.10.1. Transport Stream Controller Overview

The transport stream controller is responsible for de-multiplexing and pre-processing the inputting multimedia data defined in ISO/IEC 13818-1.

The transport stream controller receives multimedia data stream from SSI (Synchronous Serial Port)/SPI (Synchronous Parallel Port) inputs and de-multiplexing the data into Packets by PID (Packet Identify). Before the Packet is stored to memory by DMA, it can be pre-processed by the Transport Stream Descrambler.

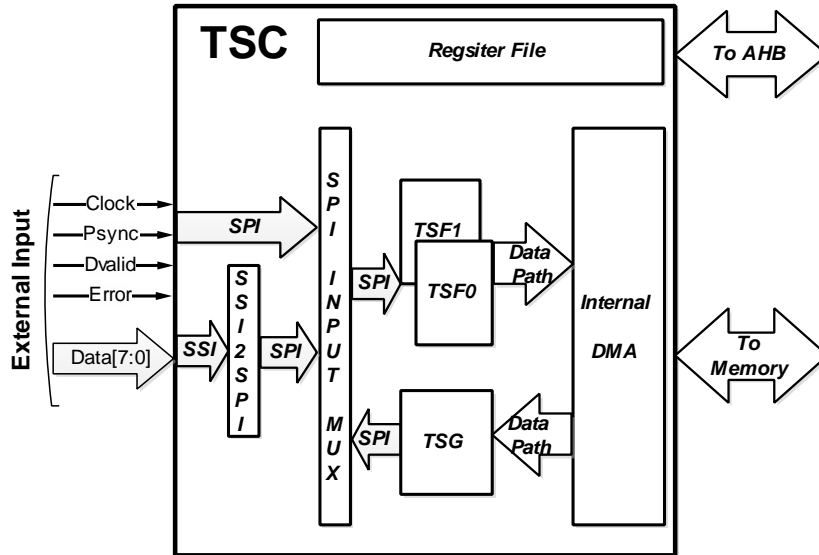
The transport stream controller can be used for almost all multi-media application cases, for example: DVB Set top Box, IPTV, stream media box, multimedia players and so on.

The Transport Stream Controller (TSC) includes the following features:

- One external Synchronous Parallel Interface (SPI) or one external Synchronous Serial Interface (SSI)
- 32 channels PID filter
- Multiple transport stream packet (188, 192, 204) format support
- SPI and SSI timing parameters are configurable
- Hardware packet synchronous byte error detection
- Hardware PCR packet detection
- Configurable SPI transport stream generator for streams in DRAM memory
- DMA is supported for data transfer
- Support DVB-CSA V1.1 Descrambler



**8.10.2. Block Diagram**

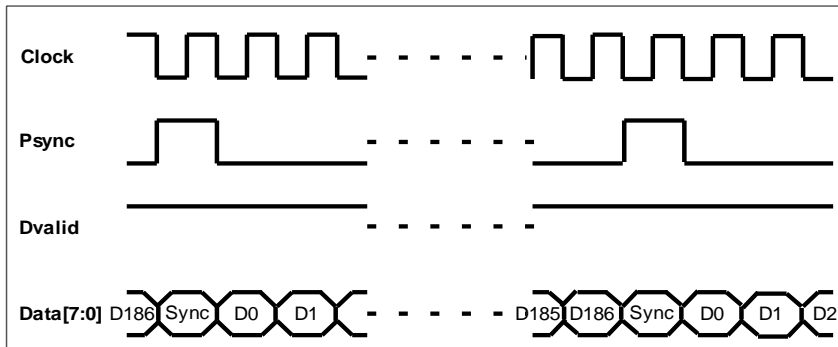


TSC – TS Controller; TSF – TS Filter; TSG – TS Generator

### 8.10.3. Transport Stream Input Timing Diagram

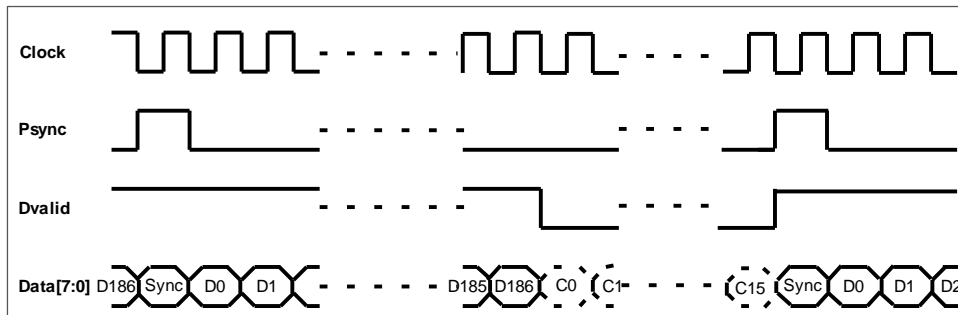
#### Input Signals Description

Name	Description
Clock	Clock of SPI/SSI data input
Psync	Packet sync (or Start flag) for TS packet
Dvalid	Data valid flag for TS data input
Error	Error flag for TS data, but do not used by TSC
Data[7:0]	TS data input.  Data[7:0] are used in SPI mode;  Only Data[0] is used in SSI mode.



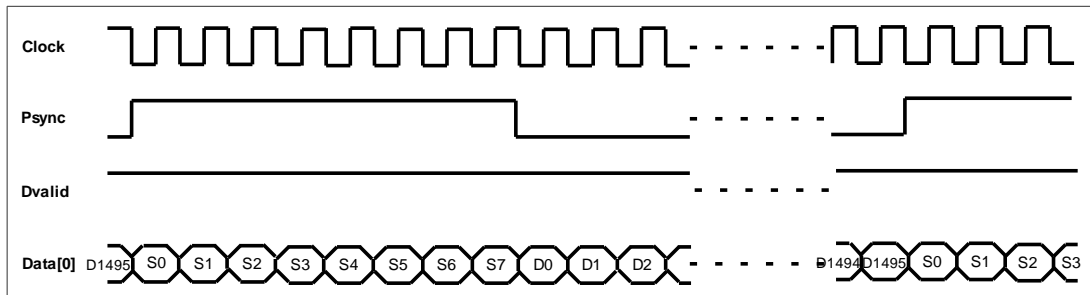
Input Timing for SPI mode

(CLOCK = Rising Edge, PSYNC = High Active, DVALID = High Active, Packet Size = 188 Bytes)



Alternative Input Timing for SPI mode

(CLOCK = Rising Edge, PSYNC = High Active, DVALID = High Active, Packet Size = 188 Bytes)



Input Timing for SSI mode

(CLOCK = Rising Edge, PSYNC = High Active, DVALID = High Active, Packet Size = 188 Bytes)

#### 8.10.4. TSC Register List

Module Name	Base Address
TSC_BASE	0x01C06000
TSG_OFFSET	0x00000040
TSF0_OFFSET	0x00000080
TSF1_OFFSET	0x00000100

Register Name	Offset	Description
TSC_CTLR	TSC + 0x00	TSC Control Register
TSC_STAR	TSC + 0x04	TSC Status Register
TSC_PCTLR	TSC + 0x10	TSC Port Control Register
TSC_PPARR	TSC + 0x14	TSC Port Parameter Register
TSC_TSFMUXR	TSC + 0x20	TSC TSF Input Multiplex Control Register
TSC_OUTMUXR	TSC + 0x28	TSC Port Output Multiplex Control Register
TSG_CTLR	TSG + 0x00	TSG Control Register
TSG_PPR	TSG + 0x04	TSG Packet Parameter Register
TSG_STAR	TSG + 0x08	TSG Status Register
TSG_CCR	TSG + 0x0c	TSG Clock Control Register
TSG_BBAR	TSG + 0x10	TSG Buffer Base Address Register
TSG_BSZR	TSG + 0x14	TSG Buffer Size Register
TSG_BPR	TSG + 0x18	TSG Buffer Pointer Register
TSF_CTLR	TSF + 0x00	TSF Control Register
TSF_PPR	TSF + 0x04	TSF Packet Parameter Register
TSF_STAR	TSF + 0x08	TSF Status Register
TSF_DIER	TSF + 0x10	TSF DMA Interrupt Enable Register
TSF_OIER	TSF + 0x14	TSF Overlap Interrupt Enable Register
TSF_DISR	TSF + 0x18	TSF DMA Interrupt Status Register
TSF_OISR	TSF + 0x1c	TSF Overlap Interrupt Status Register
TSF_PCRCR	TSF + 0x20	TSF PCR Control Register
TSF_PCRDR	TSF + 0x24	TSF PCR Data Register
TSF_CENR	TSF + 0x30	TSF Channel Enable Register
TSF_CPER	TSF + 0x34	TSF Channel PES Enable Register
TSF_CDERR	TSF + 0x38	TSF Channel Descramble Enable Register
TSF_CINDR	TSF + 0x3c	TSF Channel Index Register
TSF_CCTLR	TSF + 0x40	TSF Channel Control Register
TSF_CSTAR	TSF + 0x44	TSF Channel Status Register
TSF_CCWIR	TSF + 0x48	TSF Channel CW Index Register
TSF_CPIDR	TSF + 0x4c	TSF Channel PID Register
TSF_CBBAR	TSF + 0x50	TSF Channel Buffer Base Address Register

TSF_CBSZR	TSF + 0x54	TSF Channel Buffer Size Register
TSF_CBWPR	TSF + 0x58	TSF Channel Buffer Write Pointer Register
TSF_CBRPR	TSF + 0x5c	TSF Channel Buffer Read Pointer Register

### 8.10.5. TSC Register Description

#### 8.10.5.1. TSC Control Register

<b>Offset: 0x00</b>			<b>Register Name: TSC_CTLR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.2. TSC Status Register

<b>Offset: 0x04</b>			<b>Register Name: TSC_STAR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.3. TSC Port Control Register

<b>Offset: 0x10</b>			<b>Register Name: TSC_PCTLR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:1	/	/	/
0	R/W	0	TS Input Port0 Control  0 – SPI 1 – SSI

#### 8.10.5.4. TSC Port Parameter Register

<b>Offset: 0x14</b>			<b>Register Name: TSC_PPARR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:8	/	/	/

7:0	R/W	0x00	TS Input Port0 Parameters	
			<b>Bit</b>	<b>Definition</b>
			7:5	Reserved
			4	SSI data order  0: MSB first for one byte data  1: LSB first for one byte data
			3	CLOCK signal polarity  0 : Rise edge capturing  1: Fall edge capturing
			2	ERROR signal polarity  0: High level active  1: Low level active
			1	DVALID signal polarity  0: High level active  1: Low level active
			0	PSYNC signal polarity  0: High level active  1: Low level active

#### 8.10.5.5. TSC TSF Input Multiplex Control Register

<b>Offset: 0x20</b>			<b>Register Name: TSC_TSFMUXR</b>
			<b>Default Value: 0x0000_0000</b>
<b>Bit</b>	<b>Read/Write</b>	<b>Default</b>	<b>Description</b>
31:4	/	/	/
3:0	R/W	0x0	TSF0 Input Multiplex Control  0x0 –Data from TSG

			0x1 –Data from TS IN Port0
			Others – Reserved

#### 8.10.5.6. TSC Port Output Multiplex Control Register

<b>Offset: 0x28</b>			<b>Register Name: TSC_TSFMUXR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.7. TSG Control and Status Register

<b>Offset: TSG+0x00</b>			<b>Register Name: TSG_CSR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:26	/	/	/
			Status for TS Generator
			0: IDLE state
			1: Running state
			2: PAUSE state
25:24	R	0	Others: Reserved
23:10	/	/	/
			Loop Buffer Mode
9	R/W	0	When set to '1', the TSG external buffer is in loop mode.
			Sync Byte Check Enable
			Enable/ Disable check SYNC byte fro receiving new packet
			0: Disable
			1: Enable
8	R/W	0	If enable check SYNC byte and an error SYNC byte is receiver, TS Generator would come into PAUSE state. If the correspond interrupt is



			enable, the interrupt would happen.
7:3	/	/	/
2	R/W	0	<p>Pause Bit for TS Generator</p> <p>Write '1' to pause TS Generator. TS Generator would stop fetch new data from DRAM. After finishing this operation, this bit will clear to zero by hardware. In PAUSE state, write '1' to resume this state.</p>
1	R/W	0	<p>Stop Bit for TS Generator</p> <p>Write '1' to stop TS Generator. TS Generator would stop fetch new data from DRAM. The data already in its FIFO should be sent to TS filter. After finishing this operation, this bit will clear to zero by hardware.</p>
0	R/W	0	<p>Start Bit for TS Generator</p> <p>Write '1' to start TS Generator. TS Generator would fetch data from DRAM and generate SPI stream to TS filter. This bit will clear to zero by hardware after TS Generator is running.</p>

#### 8.10.5.8. TSG Packet Parameter Register

<b>Offset: TSG+0x04</b>			<b>Register Name: TSG_PPR</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
23:16	R/W	0x47	<p>Sync Byte Value</p> <p>This is the value of sync byte used in the TS Packet.</p>
15:8	/	/	/
7	R/W	0	<p>Sync Byte Position</p> <p>0: the 1st byte position</p> <p>1: the 5th byte position</p> <p>Notes: This bit is only used for 192 bytes packet size.</p>
6:2	/	/	/
1:0	R/W	0	<p>Packet Size</p> <p>Byte Size for one TS packet</p> <p>0: 188 bytes</p>

			Others: Reserved
--	--	--	------------------

#### 8.10.5.9. TSG Interrupt Enable and Status Register

Offset: TSG+0x08			Register Name: TSG_IESR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:20	/	/	/
19	R/W	0	TS Generator (TSG) End Interrupt Enable  0: Disable  1: Enable  If set this bit, the interrupt would assert to CPU when all data in external DRAM are sent to TS PID filter.
18	R/W	0	TS Generator (TSG) Full Finish Interrupt Enable  0: Disable  1: Enable
17	R/W	0	TS Generator (TSG) Half Finish Interrupt Enable  0: Disable  1: Enable
16	R/W	0	TS Generator (TSG) Error Sync Byte Interrupt Enable  0: Disable  1: Enable
15:4	/	/	/
3	R/W	0	TS Generator (TSG) End Status  Write '1' to clear.
2	R/W	0	TS Generator (TSG) Full Finish Status  Write '1' to clear.
1	R/W	0	TS Generator (TSG) Half Finish Status  Write '1' to clear.

0	R/W	0	TS Generator (TSG) Error Sync Byte Status  Write '1' to clear.
---	-----	---	--

#### 8.10.5.10. TSG Clock Control Register

<b>Offset: TSG+0x0c</b>			<b>Register Name: TSG_CCR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:16	R/W	0x0	TSG Clock Divide Factor (N)  The Numerator part of TSG Clock Divisor Factor.
15:0	R/W	0x0	TSG Clock Divide Factor (D)  The Denominator part of TSG Clock Divisor Factor.  Frequency of output clock:  $F_o = (F_i * (N+1)) / (8 * (D+1))$ .  F <sub>i</sub> is the input special clock of TSC, and D must not less than N.

#### 8.10.5.11. TSG Buffer Base Address Register

<b>Offset: TSG+0x10</b>			<b>Register Name: TSG_BBAR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:28	/	/	/
27:0	RW	0x0	Buffer Base Address  This value is a start address of TSG buffer.  Note: This value should be 4-word (16Bytes) align, and the lowest 4-bit of this value should be zero.

#### 8.10.5.12. TSG Buffer Size Register

<b>Offset: TSG+0x14</b>			<b>Register Name: TSG_BSZR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
23:0	R/W	0	Data Buffer Size for TS Generator  It is in byte unit.  The size should be 4-word (16Bytes) align, and the lowest 4 bits should be zero.

#### 8.10.5.13. TSG Buffer Pointer Register

<b>Offset: TSG+0x18</b>			<b>Register Name: TSG_BPR</b>
			<b>Default Value: 0x1fff_0000</b>
Bit	Read/Write	Default	Description
31:24	/	/	/
23:0	R	0	Data Buffer Pointer for TS Generator  Current TS generator data buffer read pointer (in byte unit)

#### 8.10.5.14. TSF Control and Status Register

<b>Offset: TSF+0x00</b>			<b>Register Name: TSF_CSR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:3	/	/	/
2	R/W	0	TSF Enable  0: Disable TSF Input  1: Enable TSF Input
1	/	/	/
0			TSF Global Soft Reset  A software writing '1' will reset all status and state machine of TSF. And

			it's cleared by hardware after finish reset.  A software writing '0' has no effect.
--	--	--	---

#### 8.10.5.15. TSF Packet Parameter Register

Offset: TSF+0x04			Register Name: TSF_PPR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:28	R/W	0	Lost Sync Packet Threshold It is used for packet sync lost by checking the value of sync byte.
27:24	R/W	0	Sync Packet Threshold It is used for packet sync by checking the value of sync byte.
23:16	R/W	0x47	Sync Byte Value This is the value of sync byte used in the TS Packet.
15:10	/	/	/
9:8	R/W	0	Packet Sync Method 0: By PSYNC signal 1: By sync byte 2: By both PSYNC and Sync Byte 3: Reserved
7	R/W	0	Sync Byte Position 0: the 1st byte position 1: the 5th byte position Notes: This bit is only used for 192 bytes packet size.
6:2	/	/	/
1:0	R/W	0	Packet Size Byte Size for one TS packet 0: 188 bytes

			1: 192 bytes
			2: 204 bytes
			3: Reserved

#### 8.10.5.16. TSF Interrupt Enable and Status Register

Offset: TSF+0x08			Register Name: TSF_IESR
Default Value: 0x0000_0000			
Bit	Read/Write	Default	Description
31:20	/	/	/
19	R/W	0	TS PID Filter (TSF) Internal FIFO Overrun Interrupt Enable 0: Disable 1: Enable
18	R/W	0	TS PCR Packet Detect Interrupt Enable 0: Disable 1: Enable
17	R/W	0	TS PID Filter (TSF) Channel Overlap Interrupt Global Enable 0: Disable 1: Enable
16	R/W	0	TS PID Filter (TSF) Channel DMA Interrupt Global Enable 0: Disable 1: Enable
15:4	/	/	/
3	R/W	0	TS PID Filter (TSF) Internal FIFO Overrun Status Write '1' to clear.
2	R/W	0	TS PCR Packet Found Status When it is '1', one TS PCR Packet is found. Write '1' to clear.
1	R	0	TS PID Filter (TSF) Channel Overlap Status It is global status for 16 channel. It would clear to zero after all channels status bits are clear.

0	R	0	TS PID Filter (TSF) Channel DMA status  It is global status for 16 channel. It would clear to zero after all channels status bits are clear.
---	---	---	--

#### 8.10.5.17. TSF DMA Interrupt Enable Register

<b>Offset: TSF+0x10</b>			<b>Register Name: TSF_DIER</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0x0	DMA Interrupt Enable  DMA interrupt enable bits for channel 0~31.

#### 8.10.5.18. TSF Overlap Interrupt Enable Register

<b>Offset: TSF+0x14</b>			<b>Register Name: TSF_OIER</b>  <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0x0	Overlap Interrupt Enable  Overlap interrupt enable bits for channel 0~31.

#### 8.10.5.19. TSF DMA Interrupt Status Register

<b>Offset: TSF+0x18</b>			<b>Register Name: TSF_DISR</b>  <b>Default Value: 0x3FFF_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0x0	DMA Interrupt Status  DMA interrupt Status bits for channel 0~31.  Set by hardware, and can be cleared by software writing '1'.  When both these bits and the corresponding DMA Interrupt Enable bits set, the TSF interrupt will generate.

**8.10.5.20. TSF Overlap Interrupt Status Register**

<b>Offset: TSF+0x1c</b>			<b>Register Name: TSF_OISR</b>
<b>Bit</b>			<b>Default Value: 0x0000_0000</b>
<b>Read/Write</b>	<b>Default</b>	<b>Description</b>	
31:0	R/W	0x0	<p>Overlap Interrupt Status</p> <p>Overlap interrupt Status bits for channel 0~31.</p> <p>Set by hardware, and can be cleared by software writing '1'.</p> <p>When both these bits and the corresponding Overlap Interrupt Enable bits set, the TSF interrupt will generate.</p>

**8.10.5.21. TSF PCR Control Register**

<b>Offset: TSF+0x20</b>			<b>Register Name: TSF_PCRCR</b>
<b>Bit</b>			<b>Default Value: 0x0000_0000</b>
<b>Read/Write</b>	<b>Default</b>	<b>Description</b>	
31:17	/	/	/
16	R/W	0	<p>PCR Detecting Enable</p> <p>0: Disable</p> <p>1: Enable</p>
15:13	/	/	/
12:8	R/W	0	Channel Index m for Detecting PCR packet (m from 0 to 31)
7:1	/	/	/
0	R	0	<p>PCR Contest LSB 1 bit</p> <p>PCR[0]</p>

**8.10.5.22. TSF PCR Data Register**

<b>Offset: TSF+0x24</b>			<b>Register Name: TSF_PCRDR</b>
			<b>Default Value: 0x0000_0000</b>



Bit	Read/Write	Default	Description
31:0	R	0	PCR Data High 32 bits PCR[33:1]

#### 8.10.5.23. TSF Channel Enable Register

<b>Offset: TSF+0x30</b>			<b>Register Name: TSF_CENR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0x0	Filter Enable for Channel 0~31  0: Disable  1: Enable  From Disable to Enable, internal status of the corresponding filter channel will be reset.

#### 8.10.5.24. TSF Channel PES Enable Register

<b>Offset: TSF+0x34</b>			<b>Register Name: TSF_CPER</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0x0	PES Packet Enable for Channel 0~31  0: Disable  1: Enable  These bits should not be changed during the corresponding channel enable.

#### 8.10.5.25. TSF Channel Descramble Enable Register

<b>Offset: TSF+0x38</b>			<b>Register Name: TSF_CDERR</b> <b>Default Value: 0x0000_0000</b>
-------------------------	--	--	--

Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.26. TSF Channel Index Register

<b>Offset: TSF+0x3c</b>			<b>Register Name: TSF_CINDR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:5	/	/	/
4:0	R/W	0x0	Channel Index This value is the channel index for channel private registers access. Range is from 0x00 to 0x1f. Address range of channel private registers is 0x40~0x7f.

#### 8.10.5.27. TSF Channel Control Register

<b>Offset: TSF+0x40</b>			<b>Register Name: TSF_CCTLR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.28. TSF Channel Status Register

<b>Offset: TSF+0x44</b>			<b>Register Name: TSF_CSTAR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.29. TSF Channel CW Index Register

<b>Offset: TSF+0x48</b>			<b>Register Name: TSF_CCWIR</b>
			<b>Default Value: 0x0000_0000</b>

Bit	Read/Write	Default	Description
31:0	/	/	/

#### 8.10.5.30. TSF Channel PID Register

<b>Offset: TSF+0x4c</b>			<b>Register Name: TSF_CPIDR</b>
			<b>Default Value: 0x1fff_0000</b>
Bit	Read/Write	Default	Description
31:16	R/W	0x1fff	Filter PID Mask for Channel
			Filter PID value for Channel
15:0	R/W	0x0	Filter Fit: Input PID & PID Mask == PID Value

#### 8.10.5.31. TSF Channel Buffer Base Address Register

<b>Offset: TSF+0x50</b>			<b>Register Name: TSF_CBBAR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:0	R/W	0	Data Buffer Base Address for Channel It is 4-word (16Bytes) align address. The LSB four bits should be zero.

#### 8.10.5.32. TSF Channel Buffer Size Register

<b>Offset: TSF+0x54</b>			<b>Register Name: TSF_CBSZR</b>
			<b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:26	/	/	/
25:24	R/W	0	DMA Interrupt Threshold for Channel m (m from 1 to 15)  The unit is TS packet size. When received packet (has also stored in DRAM) size is beyond ( $\geq$ ) threshold value, the corresponding channel interrupt is generated to CPU. TSC should count the new received packet again, when exceed the specified threshold value, one new interrupt is generated again.

			0: 1/2 data buffer packet size 1: 1/4 data buffer packet size 2: 1/8 data buffer packet size 3: 1/16 data buffer packet size
23:21	/	/	/
20:0	R/W	0	Data Buffer Packet Size for Channel  The exact buffer size of buffer is N+1 bytes.  The maximum buffer size is 2MB.  This size should be 4-word (16Bytes) aligned. The LSB four bits should be zero.

#### 8.10.5.33. TSF Channel Buffer Write Pointer Register

<b>Offset: TSF+0x58</b>			<b>Register Name: TSF_CBWPR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:21	/	/	/
20:0	R/W	0	Data Buffer Write Pointer (in Bytes)  This value is changed by hardware, when data is filled into buffer, this pointer is increased.  And this pointer can be set by software, but it should not be changed by software during the corresponding channel is enable.

#### 8.10.5.34. TSF Channel Buffer Read Pointer Register

<b>Offset: TSF+0x5c</b>			<b>Register Name: TSF_CBRPR</b> <b>Default Value: 0x0000_0000</b>
Bit	Read/Write	Default	Description
31:21	/	/	/
20:0	R/W	0	Data Buffer Read Pointer (in Bytes)  This pointer should be changed by software after the data of buffer is

			read.	
--	--	--	-------	--

## 8.11. EMAC

### 8.11.1. Overview

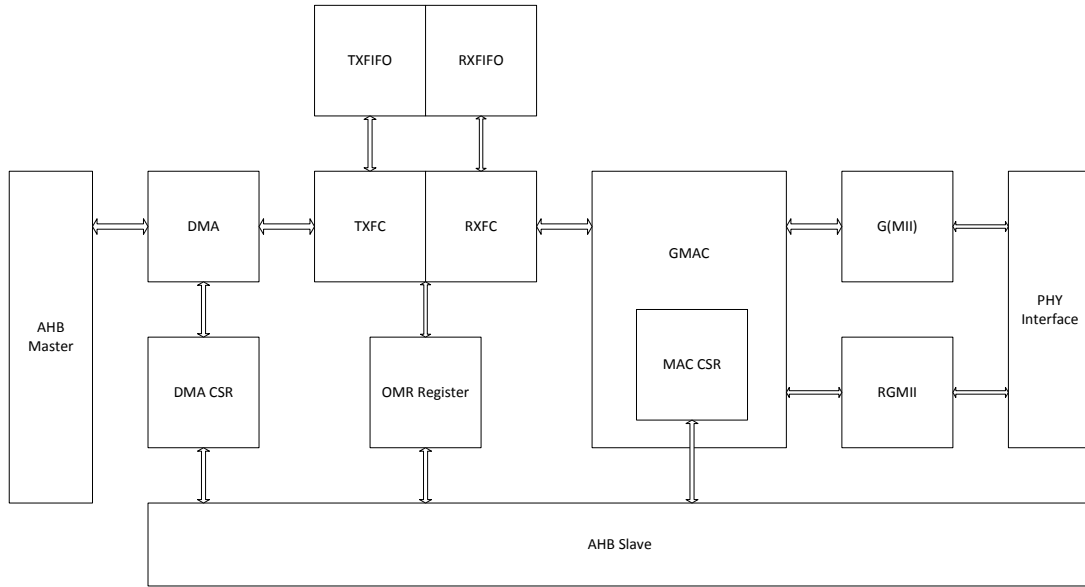
The Ethernet MAC(EMAC) controller enables a host to transmit and receive data over Ethernet in compliance with the IEEE 802.3-2002 standard. It supports 10M/100M/1000M external PHY with MII/ RGMII interface in both full and half duplex mode. The Ethernet MAC-DMA is designed for packet-oriented data transfers based on a linked list of descriptors. 4K Byte TXFIFO and 16K Byte RXFIFO are provided to keep continuous transmission and reception. Flow Control, CRC Pad & Stripping, and address filtering are also supported in this module.

The Ethernet MAC Controller includes the following features:

- Supports 10/100/1000Mbps data transfer rates
- Supports MII/RGMII PHY interface
- Supports both full-duplex and half-duplex operation
- Automatic CRC and pad generation controllable on a per-frame basis
- Options for Automatic Pad/CRC Stripping on receive frames
- Programmable frame length to support Standard or Jumbo Ethernet frames with sizes up to 16 KB
- Programmable Inter Frame Gap (40-96 bit times in steps of 8)
- Supports a variety of flexible address filtering modes
- Separate 32-bit status returned for transmission and reception packets
- Optimization for packet-oriented DMA transfers with frame delimiters
- Dual-buffer (ring) or linked-list (chained) descriptor chaining
- Descriptor architecture, allowing large blocks of data transfer with minimum CPU intervention; each descriptor can transfer up to 4 KB of data
- Comprehensive status reporting for normal operation and transfers with errors
- 4KB TXFIFO for transmission packets and 16KB RXFIFO for reception packets
- Programmable interrupt options for different operational conditions

**8.11.2. Block Diagram**

The EMAC Controller system block diagram is shown below:



EMAC Block Diagram

# Appendix

## Control signal and data port mapping for TCON:

IF	Color	cycle	I00	I01	I02	I03	D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0		
CPU	256K		CS	RD	WR	A1	R5	R4	R3	R2	R1	R0			G5	G4	G3	G2	G1	G0			B5	B4	B3	B2	B1	B0				
CPU 16BIT	256K	1 <sup>st</sup>	CS	RD	WR	A1	R5	R4	R3	R2	R1	R0						G5	G4	G3			G2	G1	G0							
		B5					B4	B3	B2	B1	B0																					
		2 <sup>nd</sup>	CS	RD	WR	A1	R5	R4	R3	R2	R1	R0							G5	G4	G3			G2	G1	G0						
		B5					B4	B3	B2	B1	B0																					
		1 <sup>st</sup>	CS	RD	WR	A1	R5	R4	R3	R2	R1	R0								G5	G4	G3			G2	G1	G0					
		B5					B4	B3	B2	B1	B0																					
2 <sup>nd</sup>	CS	RD	WR	A1	R5	R4	R3	R2	R1	R0								G5	G4	G3			G2	G1	G0							
B5					B4	B3	B2	B1	B0																							
CPU	256K		CS	RD	WR	A1	R4	R3	R2	R1	R0	G5		G4	G3		G2	G1	G0			B4	B3	B2	B1	B0						
CPU	256K		CS	RD	WR	A1												R5	R4	R3			R2	R1	R0	G5	G4	G3				
CPU 8BIT	256K	1 <sup>st</sup>	CS	RD	WR	A1												R5	R4	R3			R2	R1	R0							
		2 <sup>nd</sup>	CS	RD	WR	A1													G5	G4	G3			G2	G1	G0						
3 <sup>rd</sup>	CS	RD					WR	A1													B5	B4	B3			B2	B1	B0				
65K			CS	RD	WR	A1															R4	R3	R2			R1	R0	G5	G4	G3		
2 <sup>nd</sup>	CS	RD					WR	A1													G2	G1	G0			B4	B3	B2	B1	B0		
CPU CMD				CS	RD	WR			A1	D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1
HV	24BIT parallel		VSYNC	HSYNC	DCLK	LDE	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0		
	24BIT serial	1 <sup>st</sup>	VSYNC	HSYNC	DCLK	LDE													D17	D16	D15			D14	D13	D12	D11	D10				
		2 <sup>nd</sup>																					D27	D26	D25			D24	D23	D22	D21	D20
3 <sup>rd</sup>																			D37	D36	D35			D34	D33	D32	D31	D30				
TTL			STV	CKV	STH	OEH	R7	R6	R5	R4	R3	R2	CKH	REV	G7	G6	G5	G4	G3	G2	REVB	CKH1	B7	B6	B5	B4	B3	B2	OEV	CKH2		
HDTV HDMI	24BIT parallel		VS	HS	CLKO	BLANK	HD23	HD22	HD21	HD20	HD19	HD18	HD17	HD16	HD15	HD14	HD13	HD12	HD11	HD10	HD9	HD8	HD7	HD6	HD5	HD4	HD3	HD2	HD1	HD0		