M200

Wearable Application Processor

Programming Manual

Release Date: Nov. 20, 2014

北京君正集成电路股份有限公司
Ingenic Semiconductor Co.,Ltd.

M200 Wearable Application Processor

Programming Manual

Copyright © 2005-2014 Ingenic Semiconductor Co. Ltd. All rights reserved.

Disclaimer

This documentation is provided for use with Ingenic products. No license to Ingenic property rights is granted. Ingenic assumes no liability, provides no warranty either expressed or implied relating to the usage, or intellectual property right infringement except as provided for by Ingenic Terms and Conditions of Sale.

Ingenic products are not designed for and should not be used in any medical or life sustaining or supporting equipment.

All information in this document should be treated as preliminary. Ingenic may make changes to this document without notice. Anyone relying on this documentation should contact Ingenic for the current documentation and errata.

Ingenic Semiconductor Co., Ltd.

Ingenic Headquarters, East Bldg. 14, Courtyard #10,
XIbeiwang East Road, Haidian District, Beijing 100193, China,
Tel: 86-10-56345000
Fax: 86-10-56345001
Http: //www.ingenic.com

CONTENTS

CONTENTS i

TABLES i

FIGURES i

Section 1 Overview

1 Overview 2

1.1 Block Diagram 2
1.2 Features 2
1.2.1 CPU Core 2
1.2.2 VPU Core 3
1.2.3 GPU Core 3
1.2.4 ISP Core 4
1.2.5 Display/Camera/Audio 4
1.2.6 Memory Interface 7
1.2.7 System Functions 8
1.2.8 Peripherals 9
1.2.9 Bootrom 11

Section 2 Core Functions

2 CPU 13

2.1 Overview 13
2.2 M200-CPU 13

3 VPU 14

3.1 Overview 14
3.2 Block Diagram 14
3.3 Features of VPU 15
3.4 Internal physical address base definition 17

4 GPU 18

5 ISP 19

5.1 Block Diagram 19
5.2 Features of ISP 20
5.3 Internal ram physical address 22

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 3 Display/Camera/Audio

6 LCD Controller 24

6.1 Overview 24
6.2 Pin Description 25
6.3 Block Diagram 26
6.4 LCD Display Timing 28
6.5 OSD Graphic 29
6.5.1 Color Key 29
6.6 Register Description 31
6.6.1 Configure Register (LCDCFG) 34
6.6.2 Control Register (LCDCTRL) 36
6.6.3 Status Register (LCDSTATE) 37
6.6.4 OSD Configure Register (LCDOSDC) 38
6.6.5 OSD Control Register (LCDOSDCTRL) 39
6.6.6 OSD State Register (LCDOSDS) 40
6.6.7 Background0 Color Register (LCDBGC0) 40
6.6.8 Background1 Color Register (LCDBGC1) 41
6.6.9 Foreground Color Key Register 0 (LCDKEY0) 41
6.6.10 Foreground Color Key Register 1 (LCDKEY1) 42
6.6.11 ALPHA Register (LCDALPHA) 42
6.6.12 RGB Control (LCDRGBC) 43
6.6.13 Virtual Area Setting (LCDVAT) 44
6.6.14 Display Area Horizontal Start/End Point (LCDDAH) 44
6.6.15 Display Area Vertical Start/End Point (LCDDAV) 45
6.6.16 Foreground 0 XY Position Register (LCDXYP0) 45
6.6.17 Foreground 1 XY Position Register (LCDXYP1) 45
6.6.18 Foreground 0 Size Register (LCDSIZE0) 46
6.6.19 Foreground 1 Size Register (LCDSIZE1) 46
6.6.20 Vertical Synchronize Register (LCDVSYNC) 46
6.6.21 Horizontal Synchronize Register (LCDHSYNC) 47
6.6.22 PS Signal Setting (LCDPS) 47
6.6.23 CLS Signal Setting (LCDCLS) 48
6.6.24 SPL Signal Setting (LCDSPL) 48
6.6.25 REV Signal Setting (LCDREV) 48
6.6.26 Interrupt ID Register (LCDIID) 49
6.6.27 Descriptor Address Registers (LCDDAx) 49
6.6.28 Source Address Registers (LCDSA) 51
6.6.29 Frame ID Registers (LCDFIDx) 51
6.6.30 DMA Command Registers (LCDCMDx) 51
6.6.31 DMA OFFSIZE Registers (LCDOFFSx) 52
6.6.32 DMA Page Width Registers (LCDPWx) 53

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.6.33 DMA Commend Counter Registers (LCDCNUMx) 53
6.6.34 DMA Commend Counter Registers (LCDCPOSx) 54
6.6.35 Foreground x Size in Descriptor (LCDDESSIZEx) 55
6.6.36 Priority level threshold configure Register (LCDPCFG) 55
6.7 LCD Controller Pin Mapping 56
6.7.1 TFT and CCIR Pin Mapping 56
6.7.2 Data mapping to GPIO function. 58
6.8 Display Timing 59
6.8.1 General 16-bit and 18-bit TFT Timing 59
6.8.2 8-bit Serial TFT Timing 60
6.8.3 Delta RGB panel timing 60
6.8.4 RGB Dummy mode timing 61
6.9 Format of Frame Buffer 62
6.9.1 16bpp 62
6.9.2 18bpp 63
6.9.3 24bpp 63
6.9.4 30bpp 63
6.9.5 16bpp with alpha 63
6.9.6 18bpp with alpha 63
6.9.7 24bpp with alpha 63
6.9.8 24bpp compressed 64
6.10 Format of Data Pin Utilization 64
6.10.1 18-bit Parallel TFT 64
6.10.2 16-bit Parallel TFT 64
6.10.3 8-bit Serial TFT (24bpp) 64
6.11 LCD Controller Operation 65
6.11.1 Set LCD Controller AHB Clock and Pixel Clock 65
6.11.2 Enabling the Controller 65
6.11.3 Disabling the Controller 66
6.11.4 Resetting the Controller 66
6.11.5 Frame Buffer 66
6.11.6 OSD Operation 67
6.11.7 Descriptor Operation 67

7 Smart LCD Controller 69

7.1 Overview 69
7.2 Structure 69
7.3 Pin Description 70
7.4 Register Description 70
Address Setup and Hold Time Register 71
Slow Mode Wait Time Register 71
7.4.1 SLCD Configure Register (MCFG) 71
7.4.2 SLCD Configure Register (MCFG_NEW) 72

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4.3 SLCD Control Register (MCTRL) 75
7.4.4 SLCD Status Register (MSTATE) 76
7.4.5 SLCD Data Register (MDATA) 77
7.4.6 SLCD Wait Time Register (WTIME) 77
7.4.7 Address Setup and Hold Time Register (TASH) 78
7.4.8 Slow Mode Wait Time Register(SMWT) 78
7.5 System Memory Format 78
7.5.1 Data format 78
7.5.2 Command Format 78
7.6 Transfer Mode 79
7.6.1 DMA Transfer Mode 79
7.6.2 Register Transfer Mode 80
7.7 Timing 81
7.7.1 Parallel Timing 81
7.7.2 Serial Timing 81
7.8 Operation 81
7.8.1 DMA Operation 81
7.8.2 Register Operation 82

8 MIPI DSI Controller 84

8.1 Overview 84
8.2 Registers 84
8.2.1 Registers Memory Map 85
8.2.2 Registers and Fields Description 87
8.2.3 Static and Dynamic Registers 117
8.3 MIPI DSI Host Controller Guide 118
8.3.1 Initializing MIPI DSI Host Controller and D-PHY 118
8.3.2 ULPM Flow 121

9 EPD Controller 123

9.1 Overview 123
9.2 EPDC Pin Description 123
9.3 Register Description EPD Controller Registers 124
9.3.1 Registers Mapping 124
9.4 Registers Description 125
9.4.1 EPDC Control Registers 125
9.4.2 EPDC Configuration Registers 126
9.4.3 EPDC Status Register 127
9.4.4 EPDC ISR Control Register 128
9.4.5 EPDC DISPLAY0 Register 128
9.4.6 EPDC DISPLAY1 Register 129
9.4.7 EPDC SIZE Setting Register 130
9.4.8 EPDC Virtual Display Area Setting Register 130

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.4.9 EPDC Vertical Display Area Setting Register 131
9.4.10 EPDC Horizontal Display Area Setting Register 131
9.4.11 EPDC Vertical Synchronous Start Pulse Setting 131
9.4.12 EPDC Horizontal Synchronous Start Pulse Setting 132
9.4.13 EPDC Gate Driver Clock Setting Register 132
9.4.14 EPDC Gate Output Enable Setting Register 133
9.4.15 EPDC Gate Driver Start Pulse Setting 133
9.4.16 EPDC Source Driver Output Enable Setting Register 133
9.4.17 EPDC Source Driver Start Pulse Setting Register 134
9.4.18 EPDC Power Management Registers 0 134
9.4.19 EPDC Power Management Registers 1 134
9.4.20 EPDC Power Management Registers 2 135
9.4.21 EPDC Power Management Registers 4 135
9.4.22 EPDC LUT Base address Registers 136
9.4.23 EPDC LUT Size Registers 136
9.4.24 EPDC Current Buffer Base address Registers 136
9.4.25 EPDC Current Buffer Size Registers 137
9.4.26 EPDC Working Buffer0 Base address Registers 137
9.4.27 EPDC Working Buffer0 Size Registers 137
9.4.28 EPDC Working Buffer1 Base address Registers 138
9.4.29 EPDC Working Buffer0 Size Registers 138
9.4.30 EPDC Border Registers 0~1 138
9.4.31 EPDC PRIORITY Registers 139
9.5 Application Guide 140
9.5.1 Data Buffer Requirement 140
9.5.2 Power On/Off Sequence 140
9.5.3 Display Timing Setting 141
9.5.4 Auto Stop 143
9.5.5 Partial/Parallel Update 144
9.5.6 Update BD0~15 144
9.5.7 Border Display 144
9.5.8 Update flow 144
10 Image Process Unit 146
10.1 Overview 146
10.1.1 Feature 146
10.2 Block 147
10.3 Data flow 147
10.3.1 Input data 147
10.3.2 Output data 147
10.4 Function description 148
10.4.1 Source picture and buffer 148
10.4.2 Picture resize 149

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4.3 Color convention 149
10.4.4 Result picture and buffer 150
10.4.5 Result picture direct to LCD 150
10.4.6 Picture crop 151
10.4.7 Run the IPU 151
10.4.8 IPU work finish and free 152
10.4.9 STOP, reset the IPU 152
10.4.10 Interrupt 152
10.4.11 Switch between to lcd and to result buffer 152
10.4.12 TLB 153
10.5 Registers Descriptions 155
10.5.1 IPU Frame Control (0x0) 155
10.5.2 IPU Status (0x4) 157
10.5.3 Data Format (0x08) 157
10.5.4 Input Y Data Address (0xC) 159
10.5.5 Input U Data Address (0x10) 160
10.5.6 Input V Data Address (0x14) 160
10.5.7 Input Geometric Size (0x18) 161
10.5.8 Input Y Data Line Stride (0x1C) 161
10.5.9 Input UV Data Line Stride (0x20) 162
10.5.10 Output Frame Start Address (0x24) 162
10.5.11 Output Geometric Size (0x28) 162
10.5.12 Output Data Line Stride (0x2C) 163
10.5.13 CSC C0 Coefficient (0x34) 163
10.5.14 CSC C1 Coefficient (0x38) 164
10.5.15 CSC C2 Coefficient (0x3C) 164
10.5.16 CSC C3 Coefficient (0x40) 165
10.5.17 CSC C4 Coefficient (0x44) 165
10.5.18 CSC Offset Parameter (0x50) 166
10.5.19 Input source TLB base address (0x54) 166
10.5.20 Destination TLB base address (0x58) 166
10.5.21 Register configure enable control (0x64) 167
10.5.22 IPU trigger (0x74) 168
10.5.23 Output Frame offset on screen (0x78) 168
10.5.24 IPU global Control(0x7C) 168
10.5.25 Input Y Data Address of next frame (0x84) 169
10.5.26 Input U Data Address of next frame (0x88) 169
10.5.27 Input V Data Address of next frame (0x8C) 170
10.5.28 Output Data Address of next frame (0x90) 170
10.5.29 Source TLB base address of next frame (0x94) 171
10.5.30 Destination TLB base address of next frame (0x98) 171
10.5.31 IPU OSD Control(0x9C) 171
1.1.1 TLB parameter (0xA0) 172

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.32 Virtual address for source TLB error (0xA4) ... 172
10.5.33 Virtual address for destination TLB error (0xA8) ... 172
10.5.34 Resize Coefficients (0xB0) ... 173
10.6 IPU Operation Flow ... 174
10.6.1 Data out to frame buffer ... 174
10.6.2 Data out to lcdc ... 175
10.6.3 TLB refill ... 176
10.7 Special Instruction ... 177
10.7.1 Resizing size feature ... 177
10.7.2 Color convention feature ... 177
10.7.3 YUV/YCbCr to RGB CSC Equations ... 177
10.7.4 Output data package format (RGB order) ... 178
10.7.5 Input data package format (RGB order) ... 178
10.7.6 Source Data storing format in external memory (separated YUV Frame) ... 179

11 MIPI_CSI Interface ... 180

11.1 Overview ... 180
11.1.1 Features ... 180
11.2 Register Descriptions ... 180
11.2.1 Register Mapping ... 180
11.2.2 VERSION ... 181
11.2.3 N_LANES ... 182
11.2.4 PHY_SHUTDOWNZ ... 182
11.2.5 DPHY_RSTZ ... 183
11.2.6 CSI2_RESETN ... 183
11.2.7 PHY_STATE ... 184
11.2.8 DATA_IDS_1 ... 184
11.2.9 DATA_IDS_2 ... 185
11.2.10 ERR1 ... 185
11.2.11 ERR2 ... 187
11.2.12 MASK1 ... 188
11.2.13 MASK2 ... 189
11.2.14 PHY_TST_CRTL0 ... 190
11.2.15 PHY_TST_CRTL1 ... 191
11.3 Operation ... 192

12 AC97/I2S/SPDIF Controller ... 193

12.1 Overview ... 193
12.1.1 Block Diagram ... 194
12.1.2 Features ... 194
12.1.3 Interface Diagram ... 195
12.1.4 Signal Descriptions ... 197
12.2 Register Descriptions ... 199

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.2.1 AIC Configuration Register (AICFR) 200
12.2.2 AIC Common Control Register (AICCR) 202
12.2.3 AIC AC-link Control Register 1 (ACCR1) 206
12.2.4 AIC AC-link Control Register 2 (ACCR2) 207
12.2.5 AIC I2S/MSB-justified Control Register (I2SCR) 208
12.2.6 AIC Controller FIFO Status Register (AICSR) 209
12.2.7 AIC AC-link Status Register (ACSR) 211
12.2.8 AIC I2S/MSB-justified Status Register (I2SSR) 212
12.2.9 AIC AC97 CODEC Command Address & Data Register (ACCAR, ACCDR) 213
12.2.10 AIC AC97 CODEC Status Address & Data Register (ACSAR, ACSDR) 214
12.2.11 AIC I2S/MSB-justified Clock Divider Register (I2SDIV) 214
12.2.12 AIC FIFO Data Port Register (AICDR) 215
12.2.13 SPDIF Enable Register (SPENA) 216
12.2.14 SPDIF Control Register (SPCTRL) 216
12.2.15 SPDIF State Register (SPSTATE) 217
12.2.16 SPDIF Configure 1 Register (SPCFG1) 218
12.2.17 SPDIF Configure 2 Register (SPCFG2) 219
12.2.18 SPDIF FIFO Register (SPFIFO) 221
12.3 Serial Interface Protocol 221
12.3.1 AC-link serial data format 221
12.3.2 I2S and MSB-justified serial audio format 222
12.3.3 Audio sample data placement in SDATA_IN/SDATA_OUT 224
12.3.4 SPDIF Protocol 225
12.4 AC97/I2S Operation 226
12.4.1 Initialization 227
12.4.2 AC ’97 CODEC Power Down 228
12.4.3 Cold and Warm AC ’97 CODEC Reset 228
12.4.4 External CODEC Registers Access Operation 229
12.4.5 Audio Replay 230
12.4.6 Audio Record 231
12.4.7 FIFOs operation 232
12.4.8 Data Flow Control 234
12.4.9 Audio Samples format 235
12.4.10 Serial Audio Clocks and Sampling Frequencies 237
12.4.11 Interrupts 241
12.5 SPDIF Guide 241
12.5.1 Set SPDIF clock frequency 241
12.5.2 PCM audio mode operation (Reference IEC60958) 242
12.5.3 Non-PCM mode operation (Reference IEC61937) 242
12.5.4 Disable operation 242

13 PCM Interface 243

13.1 Overview 243

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.2 Pin Description 243
13.3 Block Diagram 244
13.4 Registers 244
13.4.1 Registers Memory Map 245
13.4.2 Register Description 245
13.5 PCM Interface Timing 251
13.5.1 Short Frame SYN 251
13.5.2 Long Frame SYN 252
13.5.3 Multi-Slot Operation 252
13.6 PCM Operation 253
13.6.1 PCM Initialization 253
13.6.2 Audio Replay 253
13.6.3 Audio Record 254
13.6.4 FIFOs operation 254
13.6.5 Data Flow Control 255
13.6.6 PCM Serial Clocks and Sampling Frequencies 256
13.6.7 Interrupts 256

14 Internal CODEC Interface 258

14.1 Overview 258
14.2 Features 258
14.2.1 Signal Descriptions 259
14.2.2 Block Diagram 260
14.2.3 Application schematic 261
14.3 Mapped Register Descriptions 262
14.3.1 CODEC internal register access control (RGADW) 262
14.3.2 CODEC internal register data output (RGDATA) 263
14.4 Operation 264
14.4.1 Access to internal registers of the embedded CODEC 264
14.4.2 CODEC controlling and typical operations 264
14.4.3 Power saving 265
14.4.4 Pop noise and the reduction of it 265
14.5 Timing parameters 266
14.6 C parameters 267
14.7 CODEC internal Registers 268
14.7.1 Registers Memory Map 268
14.7.2 Register Description 269
14.8 Programmable gains 297
14.8.1 Programmable boost gain: GIM 297
14.8.2 Programmable input gain amplifier: GID 297
14.8.3 Programmable digital attenuation: GOD 298
14.8.4 Programmable attenuation: GO 298
14.8.5 Programmable Bypass path attenuation: GI 299

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.8.6 Programmable digital mixer gain: GIMIX and GOMIX 300
14.8.7 Gain refresh strategy 300
14.9 Configuration of the headphone output stage 301
14.10 Out-of-band noise filtering 301
14.11 Output short-circuit protection (headphone output) 301
14.11.1 Indication of the short circuit detection 301
14.11.2 Reset of short circuit detection 302
14.11.3 Capacitor-coupled headphone connection 302
14.12 Sampling frequency: FREQ 302
14.13 Programmable data word length 303
14.14 Ramping system note 303
14.15 AGC system guide 304
14.15.1 AGC operating mode 304
14.16 Digital Mixer description 306
14.17 Digital microphone interface 307
14.17.1 Timing Diagram 308
14.17.2 Timings 308
14.17.3 Noise template (TBC) 309
14.18 CODEC I2S interface DAC part slave mode. 309
14.19 CODEC Operating modes 310
14.19.1 Power-On mode and Power-Off mode 310
14.19.2 RESET mode 311
14.19.3 STANDBY mode 311
14.19.4 SLEEP mode 311
14.19.5 Initial all the gain 312
14.19.6 Soft Mute mode 312
14.19.7 Power-Down mode and ACTIVE mode 314
14.19.8 Working modes summary 314
14.20 SYS_CLK turn-off and turn-on 315
14.21 Requirements on outputs and inputs selection and power-down modes 315
14.22 Anti-pop operation sequences 315
14.22.1 Initialization and configuration 316
14.22.2 Start up sequence (DAC) 316
14.22.3 Shutdown sequence (DAC) 318
14.22.4 Start up sequence (Line input) 319
14.22.5 Shutdown sequence (Line input) 320
14.23 Circuits design suggestions 321
14.23.1 Avoid quiet ground common currents 321
14.23.2 Headphone connection (Capacitor-coupled) 322
14.23.3 Microphone connection 322
14.23.4 Description of the connections to the jack 325
14.23.5 PCB considerations 326
14.24 Analog characteristics 327

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.24.1 Line input to audio ADC path 327
14.24.2 Audio DAC to headphone output path 328
14.24.3 Audio DAC to mono line output path 330
14.24.4 Line input to headphone output path (analog bypass) 330
14.24.5 Micbias and reference 332

15 DMIC Controller 333

15.1 Overview 333
15.1.1 Features 333
15.1.2 Pin Description 333
15.1.3 DMIC Controller Block Diagram 334
15.2 Register 334
15.2.1 Registers Memory Map 335
15.2.2 Register Description 336
15.3 DMIC Controller Timing 342
15.4 DMIC Controller Operation 342
15.4.1 Turn on data path flow 342
15.4.2 Turn on trigger path flow 342
15.4.3 Trigger application flow. 343
15.4.4 Wakeup by DMIC Trigger flow 343
15.4.5 FIFO operations 344
15.4.6 Trigger operations: 346
15.4.7 DMIC controller clocks set 349

Section 4 Memory Interface

16 DDR Controller 351

16.1 Overview 351
16.1.1 Supported DDR SDRAM Types 351
16.1.2 Block Diagram 351
16.1.3 Revision History 352
16.2 Register Description 352
16.2.1 Registers Memory Map 353
16.2.2 Registers and Fields Description 355
16.3 Functional Description 385
16.3.1 DDRC and DDR2 Memory Initialization Sequence 385
16.4 Change Clock Frequency 386
16.4.1 Manually SELF-REFRESH Mode 386
16.4.2 CPM driven SELF-REFRESH Mode 386
16.4.3 DLL bypass mode 386
16.5 Data Endian 387
16.6 Retention Flow 387
16.6.1 Enter Retention mode 387

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.6.2 Exit Retention mode 388
16.7 DDR Connection Diagrams 391
16.7.1 Connection to one 512Mb x16 DDR2 device 391
16.7.2 Connection to two 512Mb x16 DDR2 devices 392

17 NAND Flash Interface 393

17.1 Overview 393
17.2 Pin Description 393
17.3 Physical Address Space Map 394
17.4 NAND Register Description 397
17.4.1 Register Description 397
17.4.2 NAND Flash Boot Loader 414
17.4.3 NAND Flash Operation 415
17.5 Static Memory Interface 417
17.5.1 Register Description 417

18 BCH Controller 421

18.1 Overview 421
18.2 Register Description 423
18.2.1 BCH Control Register (BHCR) 423
18.2.2 BCH Control Set Register (BHCSR) 424
18.2.3 BCH Control Clear Register (BHCCR) 425
18.2.4 BCH ENC/DEC Count Register (BHCNT) 426
18.2.5 BCH Data Register (BHDR) 426
18.2.6 BCH Parity Register (BHPARn, n=0, 1, 2, 3, 4, 5…26, 27) 426
18.2.7 BCH Error Report Register (BHERRn, n=0,1,2,3,4,5,6,7,…,61,62,63) 427
18.2.8 BCH Interrupt Status Register (BHINT) 428
18.2.9 BCH Interrupt Enable Set Register (BHINTES) 430
18.2.10 BCH Interrupt Enable Clear Register (BHINTEC) 431
18.2.11 BCH Interrupt Enable Register (BHINTE) 431
18.2.12 BCH User TAG OUTPUT Register (BHTO) 432
18.3 BCH Operation 433
18.3.1 Encoding Sequence 433
18.3.2 Decoding Sequence 434

Section 5 System Functions

19 Clock Reset and Power Controller 438

19.1 Overview 438
19.1.1 CGU Block Diagram 439
19.1.2 CGU Registers 442
19.1.3 PLL Operation 475
19.1.4 Main Clock Division Change Sequence 476

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2 Power Manager 476
19.2.1 Low-Power Modes and Function 476
19.2.2 Register Description 477
19.2.3 Doze Mode 490
19.2.4 IDLE Mode 490
19.2.5 SLEEP Mode 490
19.2.6 Power Down Mode 491
19.3 Reset Control Module 491
19.3.1 Register Description 491
19.3.2 Power On Reset 493
19.3.3 WDT Reset 493

20 Timer/Counter Unit 494

20.1 Overview 494
20.2 Pin Description 494
20.3 Register Description 495
20.3.1 Timer Control Register (TCSR) 497
20.3.2 Timer Data FULL Register (TDFR) 499
20.3.3 Timer Data HALF Register (TDHR) 499
20.3.4 Timer Counter (TCNT) 499
20.3.5 Timer Counter Enable Register (TER) 500
20.3.6 Timer Counter Enable Set Register (TESR) 501
20.3.7 Timer Counter Enable Clear Register (TECR) 502
20.3.8 Timer Flag Register (TFR) 503
20.3.9 Timer Flag Set Register (TFSR) 504
20.3.10 Timer Flag Clear Register (TFCR) 505
20.3.11 Timer Mast Register (TMR) 505
20.3.12 Timer Mask Set Register (TMSR) 506
20.3.13 Timer Mask Clear Register (TMCR) 507
20.3.14 Timer Stop Register (TSR) 508
20.3.15 Timer Stop Set Register (TSSR) 509
20.3.16 Timer Stop Clear Register (TSCR) 510
20.3.17 Timer Status Register (TSTR) 511
20.3.18 Timer Status Set Register (TSTSR) 511
20.3.19 Timer Status Clear Register (TSTCR) 512
20.3.20 Timer control mode Register (TCUMOD) 513
20.3.21 Timer fifo write data (TFWD) 514
20.3.22 Timer fifo state Register (TFIFOSR) 514
20.4 Operation 514
20.4.1 Basic Operation in TCU1 Mode 514
20.4.2 Disable and Shutdown Operation in TCU1 Mode 515
20.4.3 Basic Operation in TCU2 Mode 515
20.4.4 Disable and Shutdown Operation in TCU2 Mode 516

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.4.5 Read Counter in TCU2 Mode 516
20.4.6 Pulse Width Modulator (PWM) 516
20.4.7 Trackball Input Waveform Detect 517
20.4.8 Basic Operation in FIFO Mode 1 517
20.4.9 Basic Operation in FIFO Mode 2 518

21 Operating System Timer 520

21.1 Overview 520
21.2 Register Description 520
21.2.1 Operating System Control Register (OSTCSR) 521
21.2.2 Operating System Timer Data Register (OSTDR) 522
21.2.3 Operating System Timer Counter (OSTCNTH, OSTCNTL) 523
21.2.4 Operating System Timer Counter high 32 bits buffer (OSTCNTHBUF) 523
21.3 Operation 523
21.3.1 Basic Operation 523
21.3.2 Disable and Shutdown Operation 524

22 Interrupt Controller 525

22.1 Overview 525
22.2 Register Description 525
22.2.1 Registers Mapping 526
22.2.2 Interrupt Controller Source Register0 (ICSR0) 526
22.2.3 Interrupt Controller Source Register1 (ICSR1) 527
22.2.4 Interrupt Controller Mask Register0 (ICMR0) 527
22.2.5 Interrupt Controller Mask Register1 (ICMR1) 527
22.2.6 Interrupt Controller Mask Set Register0 (ICMSR0) 528
22.2.7 Interrupt Controller Mask Set Register1 (ICMSR1) 528
22.2.8 Interrupt Controller Mask Clear Register0 (ICMCR0) 529
22.2.9 Interrupt Controller Mask Clear Register1 (ICMCR1) 529
22.2.10 Interrupt Controller Pending Register0 (ICPR0) 529
22.2.11 Interrupt Controller Pending Register1 (ICPR1) 530
22.2.12 Interrupt Source Register0 for PDMA (DSR0) 530
22.2.13 Interrupt Mask Register0 for PDMA (DMR0) 531
22.2.14 Interrupt Pending Register0 for PDMA (DPR0) 531
22.2.15 Interrupt Source Register1 to PDMA (DSR1) 531
22.2.16 Interrupt Mask Register1 for PDMA (DMR1) 532
22.2.17 Interrupt Pending Register1 for PDMA (DPR1) 532
22.3 Software Considerations 533

23 Watchdog Timer 534

23.1 Overview 534
23.2 Register Description 534
23.2.1 Watchdog Control Register (TCSR) 535

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.2.2 Watchdog Enable Register (TCER) 536
23.2.3 Watchdog Timer Data Register (TDR) 536
23.2.4 Watchdog Timer Counter (TCNT) 537
23.3 Watchdog Timer Function 537

24 PDMA Controller 538

24.1 Overview 538
24.2 Features 538
24.3 Block Diagram 538
24.4 Memory Mapped Register Descriptions 539
24.4.1 DMA Channel Registers 539
24.4.2 Global Control Registers 539
24.5 DMA Channel Register Definition 540
24.5.1 DMA Source Address (DSAn, n = 0 ~ 31) 540
24.5.2 DMA Target Address (DTAn, n = 0 ~ 31) 541
24.5.3 DMA Transfer Count (DTCn, n = 0 ~ 31) 541
24.5.4 DMA Request Types (DRTn, n = 0 ~ 31) 541
24.5.5 DMA Channel Control/Status (DCSn, n = 0 ~ 31) 543
24.5.6 DMA Channel Command (DCMn, n = 0 ~ 31) 544
24.5.7 DMA Descriptor Address (DDAn, n = 0 ~ 31) 548
24.5.8 DMA Stride Difference (DSDn, n = 0 ~ 31) 549
24.6 DMA Global Register Definition 549
24.6.1 DMA Control 549
24.6.2 DMA Interrupt Pending (DIRQP) 550
24.6.3 DMA Doorbell (DDB) 551
24.6.4 DMA Doorbell Set (DDS) 551
24.6.5 DMA Channel Programmable (DMACP) 551
24.6.6 DMA Soft IRQ Pending 552
24.6.7 DMA Soft IRQ Mask 552
24.6.8 DMA Channel IRQ Pending to MCU 552
24.6.9 DMA Channel IRQ to MCU Mask 553
24.6.10 Programmable Channel Bound With INTC_IRQ 553
24.6.11 Special Channel 0 and Channel 1 554
24.7 MCU 554
24.7.1 MCU Control & Status 554
24.7.2 MCU Normal MailBox 555
24.7.3 MCU Security MailBox 555
24.7.4 MCU Interrupt 556
24.7.5 Multiple Bank Tightly Coupled Sharing Memory 556
24.7.6 CP0 Registers of MCU 557
24.7.7 Normal Exceptions Accepted by MCU 557
24.7.8 How to Boot MCU Up 558
24.7.9 Security features 558

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.8 DMA manipulation 558
24.8.1 Descriptor Transfer Mode 558
24.8.2 No-Descriptor Transfer Mode 561
24.9 DMA Requests 562
24.9.1 Auto Request 562
24.9.2 On-Chip Peripheral Request 562
24.10 How to Use Programmable DMA Channel 562

25 SAR A/D Controller 563

25.1 Overview 563
25.2 Pin Description 563
25.3 Registers 563
25.3.1 Registers Memory Map 564
25.3.2 Registers and Fields Description 564
25.4 SAR A/D Controller Guide 569
25.4.1 Power Down Mode 569
25.4.2 VBAT Sample Operation 569
25.4.3 AUX Sample Operation 569

26 Real Time Clock 570

26.1 Overview 570
26.1.1 Features 570
26.1.2 Signal Descriptions 570
26.2 Registers 571
26.2.1 Registers Memory Map 571
26.2.2 Registers and Fields Description 573
26.3 Time Regulation 582
26.4 Operate Mode 582
26.4.1 Registers Access 582
26.4.2 Normal Mode 583
26.4.3 HIBERNATE Mode 583
26.5 Clock select 584

27 EFUSE Slave Interface (EFUSE) 586

27.1 Overview 586
27.2 Registers 586
27.2.1 Registers Memory Map 587
27.2.2 Registers and Fields Description 587
27.3 Flow 592
27.3.1 Program EFUSE Flow 592
27.3.2 Program Security Key Flow 592
27.3.3 Read EFUSE Flow 592
27.3.4 Read Security Key/Random Number Flow 593

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 6 Peripherals

28 General-Purpose I/O Ports 595

28.1 Function Overview 595
28.1.1 GPIO Port A Summary 595
28.1.2 GPIO Port B Summary 596
28.1.3 GPIO Port C Summary 597
28.1.4 GPIO Port D Summary 598
28.1.5 GPIO Port E Summary 599
28.1.6 GPIO Port F Summary 599
28.1.7 GPIO Port Z - Shadow Group 600
28.2 Registers Description 600
28.2.1 PORT PIN Level Registers (PxPIN) 602
28.2.2 PORT Interrupt Registers (PxINT) 603
28.2.3 PORT Interrupt Set Registers (PxINTS) 603
28.2.4 PORT Interrupt Clear Registers (PxINTC) 604
28.2.5 PORT Mask Registers (PxMSK) 604
28.2.6 PORT Mask Set Registers (PxMSKS) 604
28.2.7 PORT Mask Clear Registers (PxMSKC) 605
28.2.8 PORT PAT1/Direction Registers (PxPAT1) 605
28.2.9 PORT PAT1/Direction Set Registers (PxPAT1S) 606
28.2.10 PORT PAT1/Direction Clear Registers (PxPAT1C) 606
28.2.11 PORT PAT0/Data Registers (PxPAT0) 606
28.2.12 PORT PAT0/Data Set Registers (PxPAT0S) 607
28.2.13 PORT PAT0/Data Clear Registers (PxPAT0C) 607
28.2.14 PORT FLAG Registers (PxFLG) 608
28.2.15 PORT FLAG Clear Registers (PxFLGC) 608
28.2.16 PORT PULL Disable Registers (PxPE) 608
28.2.17 PORT PULL Set Registers (PxPES) 609
28.2.18 PORT PULL Clear Registers (PxPEC) 609
28.2.19 GPIOZ Group ID to Load (PzGID2LD) 610
28.2.20 PORT GLITCH FILTER Configure (PxCFG0~3) 610
28.3 Program Guide 611
28.3.1 Port Function Guide 611
28.3.2 Configure without 3rd-unexpected state 612

29 SMB Controller 614

29.1 Overview 614
29.1.1 Features 614
29.1.2 Pin Description 614
29.2 Registers 615
29.2.1 Registers Memory Map 615

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2 Registers and Fields Description 616
29.3 Operating Flow 640
29.3.1 SMB Behavior 640
29.3.2 Master Mode Operation 641
29.3.3 Slave Mode Operation 642
29.3.4 Disabling SMB 646
29.3.5 Summary the condition could flush TX FIFO 646
29.3.6 The condition could generate START, STOP and RESTART 646

30 Synchronous Serial Interface 650

30.1 Overview 650
30.2 Pin Description 650
30.3 Register Description 651
30.3.1 Registers Mapping 651
30.3.2 SSI Data Register (SSIDR) 652
30.3.3 SSI Control Register0 (SSICR0) 653
30.3.4 SSI Control Register1 (SSICR1) 655
30.3.5 SSI Status Register (SSISR) 658
30.3.6 SSI Interval Time Control Register (SSIITR) 659
30.3.7 SSI Interval Character-per-frame Control Register (SSIICR) 660
30.3.8 SSI Clock Generator Register (SSIGR) 661
30.3.9 SSI Receive Counter Register (SSIRCNT) 661
30.4 Functional Description 662
30.5 Data Formats 662
30.5.1 Motorola's SPI Format Details 663
30.5.2 TI's SSP Format Details 667
30.5.3 National Microwire Format Details 668
30.6 Interrupt Operation 670
30.7 Software Guid-line 671

31 UART Interface 672

31.1 Overview 672
31.1.1 Features 672
31.1.2 Pin Description 672
31.2 Register Descriptions 673
31.2.1 Register Mapping 673
31.2.2 UART Receive Buffer Register (URBR) 673
31.2.3 UART Transmit Hold Register (UTHR) 674
31.2.4 UART Divisor Latch Low/High Register (UDLLR / UDLHR) 674
31.2.5 UART Interrupt Enable Register (UIER) 675
31.2.6 UART Interrupt Identification Register (UIIR) 676
31.2.7 UART FIFO Control Register (UFCR) 677
31.2.8 UART Line Control Register (ULCR) 678

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.2.9 UART Line Status Register (ULSR) 679
31.2.10 UART Modem Control Register (UMCR) 681
31.2.11 UART Modem Status Register (UMSR) 682
31.2.12 UART Scratchpad Register 683
31.2.13 Infrared Selection Register (ISR) 683
31.2.14 UART M Register (UMR) 684
31.2.15 UART Add Cycle Register (UACR) 684
31.2.16 UART RXFIFO Counter Register (URCR) 685
31.2.17 UART TXFIFO Counter Register (UTCR) 685
31.3 Operation 685
31.3.1 UART Configuration 685
31.3.2 Data Transmission 686
31.3.3 Data Reception 686
31.3.4 Receive Error Handling 687
31.3.5 Modem Transfer 687
31.3.6 DMA Transfer 687
31.3.7 Slow IrDA Asynchronous Interface 687
31.3.8 For any frequency clock to use the UART 688

32 MMC/SD CE-ATA Controller 691

32.1 Overview 691
32.2 Features 691
32.3 Pins Description 691
32.4 Block Diagram 692
32.5 MMC/SD Controller Signal I/O Description 692
32.6 Register Description 693
32.6.1 MSC Control Register (MSC_CTRL) 695
32.6.2 MSC Control 2 Register (MSC_CTRL2) 696
32.6.3 MSC Status Register (MSC_STAT) 697
32.6.4 MSC Clock Rate Register (MSC_CLKRT) 699
32.6.5 MSC Command and Data Control Register (MSC_CMDAT) 700
32.6.6 MSC Response Time Out Register (MSC_RESTO) 702
32.6.7 MSC Read Time Out Register (MSC_RDTO) 703
32.6.8 MSC Block Size Register (MSC_BLKLEN) 703
32.6.9 MSC Block Count Register (MSC_NOB) 703
32.6.10 MSC Successfully-transferred Blocks Count Register (MSC_SNOB) 704
32.6.11 MSC Interrupt Mask Register (MSC_IMASK) 704
32.6.12 MSC Interrupt Flag Register (MSC_IFLG) 706
32.6.13 MSC Command Index Register (MSC_CMD) 708
32.6.14 MSC Command Argument Register (MSC_ARG) 709
32.6.15 MSC Response FIFO Register (MSC_RES) 709
32.6.16 MSC Receive FIFO Port Register (MSC_RXFIFO) 710
32.6.17 MSC Transmit FIFO Port Register (MSC_TXFIFO) 710

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.6.18 MSC Low Power Mode Register (MSC_LPM) 710
32.6.19 MSC DMA Control Register (MSC_DMAC) 711
32.6.20 MSC DMA Descriptor Address Register (MSC_DMANDA) 712
32.6.21 MSC DMA Data Address Register (MSC_DMADA) 712
32.6.22 MSC DMA Data Length Register (MSC_DMALEN) 713
32.6.23 MSC DMA Command Register (MSC_DMACMD) 713
32.6.24 MSC RTFIFO Data Counter Register(MSC_RTCNT) 714
32.7 MMC/SD Functional Description 714
32.7.1 MSC Reset 715
32.7.2 Voltage Validation 715
32.7.3 Card Registry 716
32.7.4 Card Access 717
32.7.5 Protection Management 718
32.7.6 Card Status 722
32.7.7 SD Status 725
32.7.8 SDIO 726
32.7.9 Clock Control 727
32.7.10 Application Specified Command Handling 727
32.8 MMC/SD Controller Operation 728
32.8.1 Data FIFOs 728
32.8.2 DMA and Program I/O 729
32.8.3 Start and Stop clock 732
32.8.4 Software Reset 732
32.8.5 Voltage Validation and Card Registry 733
32.8.6 Single Data Block Write 734
32.8.7 Single Block Read 735
32.8.8 Multiple Block Write 735
32.8.9 Multiple Block Read 736
32.8.10 Stream Write (MMC) 737
32.8.11 Stream Read (MMC) 738
32.8.12 Erase, Select/Deselect and Stop 738
32.8.13 SDIO Suspend/Resume 739
32.8.14 SDIO Read Wait 739
32.8.15 Operation and Interrupt 740

33 OTG Controller 742

33.1 Overview 742
33.2 Pin Description 742
33.3 Control and Status Overview 742
33.4 CSR Memory Map 743
33.4.1 Register Maps 744
33.4.2 Global CSR Map 744
33.4.3 Host Mode CSR Map 745

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.4.4 Device Mode CSR Map 745
33.4.5 Data FIFO (DFIFO) Access Register Map 747
33.4.6 Interrupt Hierarchy 748
33.5 Register Descriptions 749
33.5.1 Application Access to the CSRs 749
33.5.2 Overview of Commonly Used Register Bits 750
33.5.3 Global Registers 755
33.5.4 Host Mode Registers 793
33.5.5 Device Mode Registers 811

Section 7 Boot

34 XBurst Boot ROM Specification 845
34.1 Boot Select 845
34.2 Boot Procedure 845
34.3 SPL Structure 846
34.4 SPL Parameters 847
34.5 NAND Boot Specification 851
34.6 USB Boot Specification 856
34.7 MSC0 Boot Specification 860
34.8 eMMC Boot Specification 862
34.9 msc1 boot Specification 864
34.10 SPI boot Specification 865

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

TABLES

Table 3-1 VPU Features 15
Table 3-2 VPU Internal physical address base definition 17
Table 5-1 ISP Features 20
Table 5-2 ISP Internal ram physical address 22
Table 6-1 LCD Controller Pins Description 25
Table 6-2 Registers Memory Map-Address Base 32
Table 6-3 LCD Controller Registers Description 32
Table 7-1 SLCD Pins Description 70
Table 8-1 Registers Memory Map-Address Base 85
Table 8-2 MIPI DSI Register Description 85
Table 9-1 EPDC Pin Mapping 123
Table 9-2 Registers Memory Map-Address Base 124
Table 9-3 EPD Controller Registers 124
Table 12-1 AIC Pins Description 197
Table 12-2 AIC Registers Description 199
Table 12-3 Sample data bit relate to SDATA_IN/SDATA_OUT bit 225
Table 12-4 Cold AC ’97 CODEC Reset Timing parameters 229
Table 12-5 Warm AC ’97 CODEC Reset Timing Parameters 229
Table 12-6 Audio Sampling rate, BIT_CLK and SYS_CLK frequencies 238
Table 12-7 BIT_CLK divider setting 239
Table 12-8 Approximate common multiple of SYS_CLK for all sample rates 239
Table 12-9 CPM/AIC clock divider setting for various sampling rate if PLL = 270.64MHz 240
Table 12-10 PLL parameters and audio sample errors for EXCLK=12MHz 240
Table 13-1 PCM Interface Pins Description 243
Table 13-2 Registers Memory Map-Address Base 245
Table 13-3 PCM Registers Description 245
Table 14-1 CODEC signal IO pin description 259
Table 14-2 Internal CODEC Mapped Registers Description (AIC Registers) 262
Table 15-1 DMIC Controller Pins Description 333
Table 15-2 Registers Memory Map-Address Base 335
Table 15-3 DMIC Controller Registers Description 335
Table 16-1 Registers Memory Map-Address Base 353
Table 16-2 DDR Controller(DDRC) Registers Description 353
Table 17-1 NFI Pin Description 394
Table 17-2 Physical Address Space Map 396
Table 17-3 Default Configuration of NFI Chip Select Signals 396
Table 17-4 Registers Memory Map-Address Base 397
Table 17-5 NAND Flash Interface command and address Registers 397
Table 17-6 NAND Flash Interface Registers 398
Table 17-7 Registers Memory Map-Address Base 417
Table 17-8 NAND Flash Interface command and address Registers 417

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 18-1 Registers Memory Map-Address Base 421
Table 18-2 BCH Registers 421
Table 19-1 Registers Memory Map-Address Base 442
Table 19-2 CGU Registers Configuration 442
Table 19-3 Power/Reset Management Controller Registers Configuration 477
Table 20-1 PWM Pins Description 494
Table 20-2 Registers Memory Map-Address Base 495
Table 20-3 TCU Registers Configuration 495
Table 21-1 Registers Memory Map-Address Base 521
Table 21-2 OST Registers Configuration 521
Table 22-1 INTC Register 526
Table 23-1 Registers Memory Map-Address Base 534
Table 23-2 WDT Registers Configuration 535
Table 24-1 Registers Memory Map-Address Base 539
Table 24-2 DMA Channel Registers (n=0~31) 539
Table 24-3 Registers Memory Map-Address Base 540
Table 24-4 Global Control Registers 540
Table 24-5 Transfer Request Types 542
Table 24-6 Available RDIL 545
Table 24-7 TCSM space 556
Table 24-8 Descriptor Structure 559
Table 25-1 SADC Pin Description 563
Table 25-2 Registers Memory Map-Address Base 564
Table 25-3 SADC Register Description 564
Table 26-1 Registers Memory Map-Address Base 571
Table 26-2 Registers for real time clock 572
Table 26-3 Registers for hibernating mode 572
Table 26-4 Clock select registers 584
Table 27-1 Registers Memory Map-Address Base 587
Table 27-2 EFUSE Register Description 587
Table 28-1 GPIO Port A summary 595
Table 28-2 GPIO Port B summary 596
Table 28-3 GPIO Port C summary 597
Table 28-4 GPIO Port D summary 598
Table 28-5 GPIO Port E summary 599
Table 28-6 GPIO Port F summary 599
Table 28-7 Registers Memory Map-Address Base 601
Table 28-8 GPIO Registers 601
Table 29-1 SMB Pin Description 614
Table 29-2 Registers Memory Map-Address Base 615
Table 29-3 Registers Memory Map-Address Offset 615
Table 30-1 SSI Controller Pins Description 650
Table 30-2 Registers Memory Map-Address Base 651

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-3 SSI0 Serial Port Registers 652
Table 30-4 SSI Interrupts 670
Table 31-1 UART Pins Description 672
Table 31-2 UART Registers Description 673
Table 31-3 UART Interrupt Identification Register Description 677
Table 32-1 Command Token Format 693
Table 32-2 MMC/SD Data Token Format 693
Table 32-3 Registers Memory Map Base Address 694
Table 32-4 MMC/SD Controller Registers Map 694
Table 32-5 Command Data Block Structure 719
Table 32-6 Card Status Description 722
Table 32-7 SD Status Structure 725
Table 32-8 How to stop multiple block write 736
Table 32-9 How to stop multiple block read 737
Table 32-10 The mapping between Commands and Steps 740
Table 33-1 OTG Pins Description 742
Table 33-2 Core Global CSR Map (000h-3FFh) 744
Table 33-3 Host Mode CSR Map (400h-7FFh) 745
Table 33-4 Device Mode CSR Map (800h-BFFh) 746
Table 33-5 Data FIFO (DFIFO) Access Register Map 747
Table 33-6 List of Commonly Used Register Bits 751
Table 33-7 Control and Status Register: GOTGCTL 755
Table 33-8 Interrupt Register: GOTGINT 759
Table 33-9 AHB Configuration Register: GAHBCFG 760
Table 33-10 USB Configuration Register: GUSBCFG 763
Table 33-11 Reset Register: GRSTCTL 769
Table 33-12 Interrupt Register: GINTSTS 772
Table 33-13 Interrupt Mask Register: GINTMSK 778
Table 33-14 Host Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP 781
Table 33-15 Device Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP 781
Table 33-16 Receive FIFO Size Register: GRXFSIZ 782
Table 33-17 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Host Mode and Device Shared FIFO Mode) 782
Table 33-18 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Device Dedicated FIFO Mode) 783
Table 33-19 Non-Periodic Transmit FIFO/Queue Status Register: GNPTXSTS 783
Table 33-20 User HW Config1 Register: GHWCFG1 784
Table 33-21 User HW Config2 Register: GHWCFG2 784
Table 33-22 User HW Config3 Register: GHWCFG3 787
Table 33-23 User HW Config4 Register: GHWCFG4 789
Table 33-24 Global DFIFO Software Config Register: GDFIFOCFG 791

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-25 Host Periodic Transmit FIFO Size Register: HPTXFSIZ 791
Table 33-26 Device Periodic Transmit FIFO-n Register: DPTXFSIZn 792
Table 33-27 Device In Endpoint Transmit FIFO Size Register: (DIEPTXFn) 793
Table 33-28 Host Configuration Register: HCFG 793
Table 33-29 Host Frame Interval Register: HFIR 796
Table 33-30 Host Frame Number/Frame Time Remaining Register: HFNUM 796
Table 33-31 Host Periodic Transmit FIFO/Queue Status Register: HPTXSTS 797
Table 33-32 Host All Channels Interrupt Register: HAINT 798
Table 33-33 Host All Channels Interrupt Mask Register: HAINTMSK 798
Table 33-34 Host Port Control and Status Register: HPRT 799
Table 33-35 Host Channel-n Characteristics Register: HCCHARn 802
Table 33-36 Host Channel-n Split Control Register: HCSPLTn 803
Table 33-37 Host Channel-n Interrupt Register: HCINTn 804
Table 33-38 Host Channel-n Interrupt Mask Register: HCINTMSKn 806
Table 33-39 Host Channel-n Transfer Size Register: HCTSIZn 807
Table 33-40 Host Channel-n DMA Address Register: HCDMAn 809
Table 33-41 Host Channel-n DMA Buffer Address Register: HCDMABn 811
Table 33-42 Host Frame List Base Address Register: HFLBAddr 811
Table 33-43 Device Configuration Register: DCFG 812
Table 33-44 evice Control Register: DCTL 814
Table 33-45 Minimum Duration for Soft Disconnect 818
Table 33-46 Device Status Register: DSTS 818
Table 33-47 Device IN Endpoint Common Interrupt Mask Register: DIEPMSK 819
Table 33-48 Device OUT Endpoint Common Interrupt Mask Register: DOEPMSK 820
Table 33-49 Device All Endpoints Interrupt Register: DAINT 821
Table 33-50 Device Endpoints Interrupt Mask Register: DAINTMSK 821
Table 33-51 Device IN Token Sequence Learning Queue Read Register 1: DTKNQR1 822
Table 33-52 Device IN Token Sequence Learning Queue Register 2: DTKNQR2 822
Table 33-53 Device IN Token Sequence Learning Queue Register 3: DTKNQR3 823
Table 33-54 Device IN Token Sequence Learning Queue Register 4: DTKNQR4 823
Table 33-55 Device VBUS Discharge Time Register: DVBUSDIS 823
Table 33-56 Device VBUS Pulsing Time Register (DVBUSPULSE) 824
Table 33-57 Device Threshold Control Register (DTHRCTL) 824
Table 33-58 Device IN Endpoint FIFO Empty Interrupt Mask Register: DIEPEMPMSK 826
Table 33-59 Device Each Endpoint Interrupt Register: DEACHINT 826
Table 33-60 Device Each Endpoint Interrupt Register Mask: DEACHINTMSK 827
Table 33-61 Device Each In Endpoint-n Interrupt Register: DIEPEACHMSKn 827
Table 33-62 Device Each Out Endpoint-n Interrupt Register: DOEPEACHMSKn 828
Table 33-63 Device Control IN Endpoint 0 Control Register: DIEPCTL0 829
Table 33-64 Device OUT Endpoint 0 Control Register: DOEPCTL0 831
Table 33-65 Device Endpoint-n Control Register: DIEPCTLn/DOEPCTLn 832
Table 33-66 Device Endpoint-n Interrupt Register: DIEPINTn/DOEPINTn 836
Table 33-67 Device IN Endpoint 0 Transfer Size Register: DIEPTSIZ0 840

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-68 Device OUT Endpoint 0 Transfer Size Register: DOEPTSIZ0 840
Table 33-69 Device Endpoint-n Transfer Size Register: DIEPTSIZn/DOEPTSIZn 841
Table 33-70 Device Endpoint-n DMA Address Register: DIEPDMAn/DOEPDMAn 842
Table 33-71 Device Endpoint-n DMA Buffer Address Register: DIEPDMABn/DOEPDMABn.... 843
Table 33-72 Device IN Endpoint Transmit FIFO Status Register: DTXFSTSn..... 843
Table 34-1 Boot Configuration of M200 845
Table 34-2 SPL parameters structure 848
Table 34-3 The definition of the flag in NAND flash 851
Table 34-4 Transfer Types Used by the Boot Program 856
Table 34-5 Vendor Request 0 Setup Command Data Structure..... 858
Table 34-6 Vendor Request 1 Setup Command Data Structure..... 858
Table 34-7 Vendor Request 2 Setup Command Data Structure..... 859
Table 34-8 Vendor Request 3 Setup Command Data Structure..... 859
Table 34-9 Vendor Request 4 Setup Command Data Structure..... 859
Table 34-10 Vendor Request 5 Setup Command Data Structure..... 860
Table 34-11 SPI NOR flash boot flag informations（in spl signature） 866
Table 34-12 SPI NAND flash boot flag informations（in spl signature） 866

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

FIGURES

Figure 1-1 M200 Diagram 2
Figure 5-1 ISP Block Diagram 19
Figure 6-1 Block Diagram when use OSD mode 26
Figure 6-2 Block Diagram of not use OSD mode (this function is not supported in current version) 27
Figure 6-3 Block Diagram of HDMI interface(no HDMI in current version) 27
Figure 6-4 Display Parameters 28
Figure 6-5 OSD Graphic 29
Figure 6-6 General 16-bit and 18-bit TFT LCD Timing 59
Figure 6-7 8-bit serial TFT LCD Timing (24bpp) 60
Figure 6-8 Delta RGB timing 61
Figure 6-9 RGB Dummy timing 62
Figure 9-1 Powers On/Off Sequence 141
Figure 9-2 Source Drivers Reference Timing for TYPE1 EPD 142
Figure 9-3 Gate Drivers Reference Timing for TYPE1 EPD 142
Figure 9-4 Source Drivers Reference Timing for TYPE2 EPD 143
Figure 9-5 Gate Drivers Reference Timing for TYPE2 EPD 143
Figure 10-1 The Block about the IPUData flow 147
Figure 12-1 AIC Block Diagram 194
Figure 12-2 Interface to an External AC'97 CODEC Diagram 195
Figure 12-3 Interface to an External Master Mode I2S/MSB-Justified CODEC Diagram 196
Figure 12-4 Interface to an External Slave Mode I2S/MSB-Justified CODEC Diagram 196
Figure 12-5 Interface to a HDMI Transmitter via SPDIF Diagram 196
Figure 12-6 Interface to an internal Master Mode I2S CODEC Diagram 197
Figure 12-7 AC-link audio frame format 221
Figure 12-8 AC-link tag phase, slot 0 format 222
Figure 12-9 AC-link data phases, slot 1 ~ slot 12 format 222
Figure 12-10 I2S data format (A: LR mode) 222
Figure 12-11 I2S data format (B: RL mode) 223
Figure 12-12 MSB-justified data format (C: LR mode) 223
Figure 12-13 MSB-justified data format (D: RL mode) 224
Figure 12-14 Block format 226
Figure 12-15 Sub-frame format in PCM mode 226
Figure 12-16 Sub-frame format in non-PCM mode 226
Figure 12-17 Cold AC '97 CODEC Reset Timing 228
Figure 12-18 Warm AC '97 CODEC Reset Timing 229
Figure 12-19 Transmitting/Receiving FIFO access via APB Bus 233
Figure 12-20 One channel (Left) and Two channels (right) mode (16 bits packed mode) 235
Figure 12-21 Four channels (Left) and Six channels (right) mode (16 bits packed mode) 235
Figure 12-22 Eight channels mode (16 bits packed mode) 236
Figure 12-23 One channel (Left) and Two channels (right) mode 236

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-24 Four channels (Left) and Six channels (right) mode 236
Figure 12-25 Eight channels mode 237
Figure 12-26 SYS_CLK, BIT_CLK and SYNC generation scheme 238
Figure 13-1 Short Frame SYN Timing (Shown with 16bit Sample) 251
Figure 13-2 Short Frame SYN Timing (Shown with 16bit Sample) 251
Figure 13-3 Long Frame SYN Timing (Shown with 16bit Sample) 252
Figure 13-4 Long Frame SYN Timing (Shown with 16bit Sample) 252
Figure 13-5 Multi-Slot Frame SYN Timing (Shown with two Slots and 8bit Sample) 253
Figure 13-6 Transmitting/Receiving FIFO access via APB Bus 255
Figure 13-7 PCMCLK and PCMSYN generation scheme 256
Figure 14-1 CODEC block diagram 260
Figure 14-2 Internal CODEC works with AIC 260
Figure 14-3 AGC adjusting waves 305
Figure 14-4 AGC adjust areas 305
Figure 14-5 Digital microphone interface connection 308
Figure 14-6 Digital microphone timing diagram at MCLK = 12 MHz 308
Figure 14-7 Digital microphone modulation noise reference spectrum 309
Figure 14-8 CODEC Power Diagram 310
Figure 14-9 DAC Gain up and gain down sequence 313
Figure 14-10 ADC Gain up and gain down sequence 313
Figure 14-11 Start up sequence 317
Figure 14-12 Shutdown sequence 318
Figure 14-13 Capacitor-coupled connection 322
Figure 14-14 Ground distributing 325
Figure 14-15 the bottom corner of chip PCB Layer 326
Figure 15-1 DMIC Controller block diagram 1 of 2 334
Figure 15-2 DMIC Controller block diagram 2 of 2 334
Figure 15-3 The DMIC signal Timing 342
Figure 15-4 Mono Non-pack mode 344
Figure 15-5 Mono pack mode 344
Figure 15-6 Stereo mode 344
Figure 15-7 APB read when unpack function enable 345
Figure 15-8 APB read when unpack function disable and Big-Endian mode (UNPACK_MSB) enable 345
Figure 15-9 APB read when unpack function disable and non Big-Endian mode (UNPACK_MSB) disable 345
Figure 15-10 One word voice signal 346
Figure 15-11 Magnifying the above figure 15-10 346
Figure 15-12 A bump noise before one word voice signal 347
Figure 15-13 Magnifying the bump part of above figure 15-12 347
Figure 15-14 Magnifying the voice part of above figure 15-12 348
Figure 15-15 A bump noise before one word voice signal 348
Figure 15-16 Magnifying the one word voice signal use by multiple level trigger mode 349

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 15-17 Magnifying the one word voice signal use by multiple Schmitt trigger mode 349
Figure 16-1 DDR system block diagram 352
Figure 17-1 Physical Address Space Map..... 395
Figure 17-2 Basic Write Command to NAND FLASH 398
Figure 17-3 Basic Write Address to NAND FLASH 399
Figure 17-4 Basic Write/Read NAND FLASH 406
Figure 17-5 RB_ posedge Detected by NFI, with tBUSY 408
Figure 17-6 Read Data By Sel=0/1(EDO mode) 408
Figure 17-7 Basic Toggle Nand Write, with CS disable..... 409
Figure 17-8 Basic Toggle Nand Read, with CS disable 410
Figure 17-9 ONFI NAND Clock Timing.....411
Figure 17-10NFI CMD/ADDR Timing 411
Figure 17-11 ONFI NAND Write Data Timing 412
Figure 17-12 ONFI NAND Read Data Timing 412
Figure 17-13 Select Both of FLASH1.RB_ and FLASH2.RB_ to BUSY Detector1..... 413
Figure 17-14 Structure of NAND Flash Boot Loader 415
Figure 17-15 Static Bank 1 Partition When NAND Flash is Used (an example) 416
Figure 17-16 Example of 8-bit NAND Flash Connection..... 416
Figure 17-17 Basic Write/Read NAND FLASH 419
Figure 18-1 Block Diagram for BCH encoding 434
Figure 18-2 Block Diagram for BCH decoding 435
Figure 18-3 BCH decoding data flow 436
Figure 19-1 Block Diagram of PLL 475
Figure 24-1 Block Diagram of PDMA 538
Figure 24-2 Descriptor Transfer Flow..... 560
Figure 24-3 Example for Stride Transfer Mode 561
Figure 26-1 Core Power On 583
Figure 26-2 RTC clock selection path 584
Figure 29-1 Master Transmitter — Tx FIFO Empties/STOP Generation 647
Figure 29-2 Master Receiver — Tx FIFO Empties/STOP Generation 647
Figure 29-3 Master Transmitter — Restart bit of SMB_DC is set 647
Figure 29-4 Master Receiver — Restart bit of SMB_DC is set 648
Figure 29-5 Master Transmitter — Stop Bit of SMB_DC Set/Tx FIFO Not Empty 648
Figure 29-6 Master Transmitter — First Byte Loaded Into Tx FIFO Allowed to Empty, Restart Bit Set..... 648
Figure 29-7 Master Receiver — Stop Bit of IC_DATA_CMD Set/Tx FIFO Not Empty 649
Figure 29-8 Master Receiver — First Command Loaded After Tx FIFO Allowed to Empty/Restart Bit Set..... 649
Figure 30-1 Standard SPI Single Character Transfer Format (PHA = 0) 663
Figure 30-2 Standard SPI Single Character Transfer Format (PHA = 1) 664
Figure 30-3 Quad SPI Transfer Format (PHA = 1 and big endian) 664
Figure 30-4 Dual SPI Transfer Format (PHA = 1 and little endian) 664
Figure 30-5 Standard SPI Back-to-Back Transfer Format..... 665

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-6 Quad SPI Back-to-Back Transfer Format 665
Figure 30-7 Standard SPI Frame Interval Mode Transfer Format (ITFRM = 0, ITFRM= 0) 666
Figure 30-8 Standard SPI Frame Interval Mode Transfer Format (ITFRM = 1, ITFRM = 1) 667
Figure 30-9 TI's SSP Single Transfer Format 667
Figure 30-10 TI's SSP Back-to-back Transfer Format 668
Figure 30-11 National Microwire Format 1 Single Transfer 669
Figure 30-12 National Microwire Format 1 Back-to-back Transfer 669
Figure 30-13 National Microwire Format 2 Read Timing 669
Figure 30-14 National Microwire Format 2 Write Timing 670
Figure 32-1 MMC/SD Controller Block Diagram 692
Figure 33-1 OTG CSR Memory Map 743
Figure 33-2 Interrupt Hierarchy 748
Figure 33-3 Core Interrupt Handler 749
Figure 34-1 Boot sequence diagram of M200 846
Figure 34-2 The basic SPL structure 847
Figure 34-3 SPL parameters location 848
Figure 34-4 SPL configuration change Procedure 850
Figure 34-5 the distribution and structure of the boot code in NAND 852
Figure 34-6 M200 NAND Boot Procedure 855
Figure 34-7 USB Communication Flow 856
Figure 34-8 Typical Procedure of USB Boot 858
Figure 34-9 Typical Procedure of MSC0 Boot 861
Figure 34-10 SPL Structure of MSC0/MSC1 Boot 862
Figure 34-11 M200 eMMC Boot Procedure 863
Figure 34-12 SPL Structure of eMMC Boot 864

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 1

OVERVIEW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1 Overview

M200 is a low power consumption, high performance and high integrated application processor, the application is focus on wearable devices, such as smart watch and smart glass. And it can match the requirements of many other embedded products.

1.1 Block Diagram

Figure 1-1 M200 Diagram

1.2 Features

1.2.1 CPU Core

- Two MIPS-Based XBurst® cores (One core up to 1.2GHz for high performance application, the other is 300MHz for low power application)
- MIPS-Based XBurst CPU
  - XBurst® RISC instruction set
  - XBurst® SIMD instruction set
  - XBurst® FPU instruction set supporting both single and double floating point format which are IEEE754 compatible
  - XBurst® 9-stage pipeline micro-architecture
- MMU

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

  - 32-entry joint-TLB
  - 4 entry Instruction TLB
  - 4 entry data TLB
- L1 Cache
  - 32KB instruction cache for each core
  - 32KB data cache for each core
- Hardware debug support
- 16KB tight coupled memory
- L2 Cache
  - 512KB unify cache

1.2.2 VPU Core

- Video encoder up to 720P@30fps
  —H.264, VP8
- Video decoder up to 720P@30fps
  —H.264, MPEG-4, MPEG-1/2, VC-1, RV9, VP8

1.2.3 GPU Core

- 3D Graphic
  - OpenGL ES 2.0 / 1.1 compliance, including extensions; OpenVG 1.1
  - Low CPU loading
  - Up to 12 programmable elements per vertex
  - Dependent texture operation with high-performance
  - Alpha blending
  - Depth and stencil compare
  - Support for 8 fragment shader simultaneous textures
  - Support for 4 vertex shader simultaneous textures
  - Point sampling, bi-linear sampling, tri-linear filtering, and cubic textures
  - Resolve and fast clear
  - 8k x 8k texture size and 8k x 8k rendering target
  - 4 Vertex DMA streams
  - Supports YUV formats in display output (YUY2 4:2:2)
  - MMU functionality supported
- 2D feature
  - Bit BLT and stretch BLT
  - Rectangle fill and clear
  - Line drawing
  - High-performance stretch and shrink
  - Monochrome expansion for text rendering
  - ROP2, ROP3, ROP4
  - Alpha blending including Java 2 Porter-Duff compositing blending rules
  - 32k x 32k coordinate system

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- 90, 180, and 270 degrees rotation
- Transparency by monochrome mask, chroma key, or pattern mask
- Support 2x2 in 4x4 tile format
- A8/RG16 output with rotation in FilterBlit and BitBlit

- Multi Source Blending
  - Full support for Multi source blending with variable block size to improve BW and reduce SW overhead.
  - Up to 8 sources are supported.
  - Programmable block size guarantees cache efficiency so each source is read once and each destination is written once.
  - Supports 90, 180, 270 degree rotation with different block size for higher cache efficiency.
  - Supports independent source rotation with blending.

1.2.4 ISP Core

- ISP core for image pre-process.
  - Wide Dynamic Range Processing(EDR)
  - Dual-stream processing
  - Video and still image stabilization
  - Image cropping and rescaling
  - Auto exposure and gain control
  - Auto white balancing
  - Auto focus control
  - 50/60Hz flicker cancellation
  - Lens shading correction
  - Defect pixel correcting recorded and on the fly
  - Edge sharpening
  - Advanced noise reduction
  - Enhanced color interpolation(RGB Bayer demosaicing)
  - Color correction
  - Color management
  - Auto contrast enhancement
  - Gamma correction
  - Special digital effects
- Support MIPI-CSI2(v1.0) interface
  - 2-lanes
- Independent SMB interface(ISP_SDA, ISP_SCK)

1.2.5 Display/Camera/Audio

- LCD controller
  - Basic Features

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

        - Support panel(TFT, SLCD, MIPI-DSI)
        - Display size up to 800x480@60Hz(BPP24)
    - Colors Supports
        - Encoded pixel data of 16, 18 in TFT mode
        - Support up to 16,777,216 (16M) colors in TFT mode
    - Panel Supports
        - Support 16-bit parallel TFT panel
        - Support 18-bit parallel TFT panel
        - Support SLCD panel
    - OSD Supports
        - Supports one single color background
        - Supports two foregrounds, and every size can be set for each foreground
        - Supports one transparency for the whole graphic
        - Supports one transparency for each pixel in one graphic
        - Supports color key and mask color key
        - Supports porter-duff blending
    - MIPI-DSI(v1.1) Supports
        - 2-lanes
- EPD Controller
    - Supports multiple types of compatible EPD panels
    - Supports different size up to 480x480@60Hz
    - Supports 2/3/4 bits grayscale
    - Pixel base updating
    - Supports hand-writing mode
    - Supports SW LUT algorithm
    - Supports AUTO-DU, AUTO-GC4 mode
- Image post processor(IPU)
    - Input format
        - Separate frame: YUV /YCbCr (4:2:0, 4:2:2, 4:4:4, 4:1:1), RGB888
        - Packaged data: YUV422, RGB888, RGB565, RGB555, YUV444
        - Separate frame in block format: YUV/YCbCr 420
    - Output data format
        - RGB (565, 555, 888, AAA)
        - Packaged data YUV422
    - Color convention coefficient: configurable (CSC enable)
    - Minimum input image size (pixel): 4x4
    - Maximum input image size (pixel): 8096x8096
    - Maximum output image size (pixel)
        - Width: up to 4095 (without vertical resizing)
        - up to 1280 (with vertical resizing)
        - Height: up to 4095
    - Image resizing
        - Support bilinear

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

        - Up scaling ratios up to 1:31 in fractional steps with 1/32 accuracy
        - Down scaling ratios up to 31:1 in fractional steps with 1/32 accuracy
- Camera interface
    - MIPI-CSI2(v1.0) interface Supports (Depend on ISP feature)
        - 2-lanes
- AC97/I2S controller
    - AC-link (AC97) features
        - Up to 20 bit audio sample data sizes supported
        - DMA transfer mode supported
        - Stop serial clock supported
        - Programmable Interrupt function supported
        - Support mono PCM data to stereo PCM data expansion on audio play back
        - Support endian switch on 16-bits normal audio samples play back
        - Support variable sample rate in AC-link format
        - Multiple channel output and double rated supported for AC-link format
        - Power Down Mode and two Wake-Up modes Supported for AC-link format
    - I2S features
        - 8, 16, 18, 20 and 24 bit audio sample data sizes supported, 16 bits packed sample data is supported
        - Up to 8 channels sample data supported
        - DMA transfer mode supported
        - Stop serial clock supported
        - Programmable Interrupt function supported
        - Support share clock mode and split clock mode.
        - Support mono PCM data to stereo PCM data expansion on audio play back
        - Support endian switch on 16-bits normal audio samples play back
        - Internal programmable or external serial clock and optional system clock supported for I2S or MSB-Justified format
        - Internal I2S CODEC supported
        - Two FIFOs for transmit and receive respectively
- PCM interface
    - Support master mode and slave mode
    - Data starts with the frame PCMSYN or one PCMCLK later
    - Support three modes of operation for PCM
        - Short frame sync mode
        - Long frame sync mode
        - Multi-slot mode
    - Data is transferred and received with the MSB first
    - The PCM serial output data, PCMDOUT, is clocked out using the rising edge of the PCMSCLK
    - The PCM serial input data, PCMDIN, is clocked in on the falling edge of the PCMSCLK
    - 8/16 bit sample data sizes supported
    - DMA transfer mode supported

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

  - Two FIFOs for transmit and receive respectively with 16 samples capacity in every direction
- Internal CODEC
  - 24 bits ADC and DAC
  - Headphone load up to 16 Ohm
  - Sample frequency supported: 8k, 11.025k, 12k, 16k, 22.05k, 24k, 32k, 44.1k, 48k, and 96k
  - Stereo line input
  - DAC to HP path: Power consumption: 17.6mW, THD: -65dB @17.6mW /16Ohm
  - DAC to stereo line output path @10kOhm: SNR: 100dB A-Weighted, THD: -80dB @FS-1dB
  - Line input to ADC path: SNR: 90dB A-Weighted, THD: -80dB @FS-1dB
  - Separate power-down modes for ADC and DAC path with several shutdown modes
  - Reduction of audible glitches systems: Pop Reduction system, Soft Mute mode
  - Output short circuit protection
  - Embedded low noise Linear Regulator
  - 2 MIC in path or 2 line in path Maximum (Total 2 analog input)
  - Support Digital MIC
- Low power DMIC Controller
  - 16 bits data interface and 24bit precision internal controller.
  - SNR: 90dB, THD: -90dB @ FS -20dB
  - Linear high pass filter include. Attenuation: -2.9dB@100Hz, -22dB@27Hz. -36dB@10Hz
  - Low power voice trigger when waiting to start talking.
  - Mono and stereo digital MIC support.
  - Support voice data pre-fetch when trigger enable and the data interface disable, but do not increase the power dissipation.
  - Sample frequency supported: 8k, 16k.
  - Support low power mode, user for decrease DMIC sensor and DMIC Controller power dissipation.

1.2.6 Memory Interface

- DDR Controller
  - Support DDR2, DDR3, DDR3L, mobile DDR (LPDDR), LPDDR2, memory，up to 667Mbps
  - Support x16 and x32 external DDR bus width
  - Asynchornize to system bus and each port.
  - Support clock-stop mode
  - Support auto self-refresh mode
  - Support power-down mode and deep-power-down mode
  - Programmable DDR timing parameters
  - Programmable DDR row and column address width and order
- NAND flash interface

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Support on CS3~CS1#, sharing with static memory bank3~bank1
- Support both of conventional NAND flash memory and Toggle NAND flash memory
- Support most types of NAND flashes, 8/16-bit data access, 512B/2KB/4KB/8KB/16KB page size. For 512B page size, 3 and 4 address cycles are supported. For 2KB/4KB/8KB/16KB page size, 4 and 5 address cycles are supported
- Support read/erase/program NAND flash memory
- Support boot from NAND flash
- Support Toggle1.0/ONFI2.0 NAND

- BCH Controller
  - Support up to 64-bit ECC encoding and decoding for NAND
- The XBurst® processor system supports little endian only

1.2.7 System Functions

- Clock generation and power management
  - On-chip oscillator circuit
  - Two phase-locked loops (PLL) with programmable multiplier
  - CCLK, HHCLK, H2CLK, PCLK, H0CLK, DDR_CLK, VPU_CLK frequency can be changed separately for software by setting registers
  - Functional-unit clock gating
  - Supply block power shut down
- Timer and counter unit with PWM output and/or input edge counter
  - Provide 4 channels, all can generate PWM, two of them have input signal transition edge counter
  - 16-bit A counter and 16-bit B counter with auto-reload function every channel
  - Support interrupt generation when the A counter underflows
  - Three clock sources: RTCLK (real time clock), EXCLK (external clock input), PCLK (APB Bus clock) selected with 1, 4, 16, 64, 256 and 1024 clock dividing selected
- OS timer
  - One channel
  - 32-bit counter and 32-bit compare register
  - Support interrupt generation when the counter matches the compare register
  - Three clock sources: RTCLK (real time clock), EXCLK (external clock input), PCLK (APB Bus clock) selected with 1, 4, 16, 64, 256 and 1024 clock dividing selected
- Interrupt controller
  - Total 64 interrupt sources
  - Each interrupt source can be independently enabled
  - Priority mechanism to indicate highest priority interrupt
  - All the registers are accessed by CPU and PDMA
  - Unmasked interrupts can wake up the chip in sleep mode
  - Another set of source, mask and pending registers to serve for PDMA
- Watchdog timer
  - Generates WDT reset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

  - A 16-bit Data register and a 16-bit counter
  - Counter clock uses the input clock selected by software
    - PCLK, EXTAL and RTCCLK can be used as the clock for counter
    - The division ratio of the clock can be set to 1, 4, 16, 64, 256 and 1024 by software
- PDMA Controller
  - Support up to 32 independent DMA channels
  - Descriptor or No-Descriptor Transfer mode
  - A simple Xburst®-1 CPU supports smart transfer mode controlled by programmable firmware
  - Transfer data units: 1-byte, 2-byte, 4-byte, 16-byte, 32-byte, 64-byte, 128-byte
  - Transfer number of data unit: 1 ~ 2^{24} - 1
  - Independent source and destination port width: 8-bit, 16-bit, 32-bit
  - Fixed three priorities of channel groups: 0~3, highest; 4~11: mid; 12~31: lowest
  - A dedicated bus interface - BIF interconnects with on-chip BCH
  - A dedicated bus interface - NIF interconnects with on-chip NEMC or off-chip NEMC.
  - An extra INTC IRQ can be bound to one programmable DMA channel
- SAR A/D Controller
  - 3 Channels
  - Resolution: 12-bit
  - Integral nonlinearity: ±1 LSB
  - Differential nonlinearity: ±0.5 LSB
  - Resolution/speed: up to 2Msps
  - Max Frequency: 200k
  - Low power dissipation: 1.5mW(worst)
  - Support write control command by software
  - Support voltage measurement (Through pin VBAT)
  - Support two auxiliary input (Through pin AUX1, AUX2)
- RTC (Real Time Clock)
  - Need external 32768Hz oscillator for 32KHz clock generation.
  - RTCLK selectable from the oscillator or from the divided clock of EXCLK, so that 32k crystal can be absent if the hibernating mode is not needed
  - 32-bits second counter
  - Programmable and adjustable counter to generate accurate 1 Hz clock
  - Alarm interrupt, 1Hz interrupt
  - Stand alone power supply, work in hibernating mode
  - Power down controller
  - Alarm wakeup
  - External pin wakeup with up to 2s glitch filter

1.2.8 Peripherals

- General-Purpose I/O ports
  - Each port can be configured as an input, an output or an alternate function port

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Each port can be configured as an interrupt source of low/high level or rising/falling edge triggering. Every interrupt source can be masked independently
- Each port has an internal pull-up or pull-down resistor connected. The pull-up/down resistor can be disabled
- GPIO output 7 interrupts, 1 for every group, to INTC

- Four SMB Controller (SMB0, SMB1, SMB2, SMB3)
  - Two-wire SMB serial interface – consists of a serial data line (SDA) and a serial clock (SCL)
  - Two speeds
    - Standard mode (100 Kb/s)
    - Fast mode (400 Kb/s)
  - Device clock is identical with pclk
  - Programmable SCL generator
  - Master or slave SMB operation
  - 7-bit addressing/10-bit addressing
  - -level transmit and receive FIFOs
  - Interrupt operation
  - The number of devices that you can connect to the same SMB-bus is limited only by the maximum bus capacitance of 400pF
  - APB interface
- Two Synchronous serial interfaces (SSI0,SSI1)
  - 3 protocols support: National's Microwire, TI's SSP, and Motorola's SPI
  - Support 1，2，4 bit data width
  - Full-duplex or transmit-only or receive-only operation
  - Programmable transfer order: MSB first or LSB first
  - 128 entries deep x 32 bits wide transmit and receive data FIFOs
  - Configurable normal transfer mode or Interval transfer mode
  - Programmable clock phase and polarity for Motorola's SSI format
  - Two slave select signal (SSI_CE0_ / SSI_CE1_) supporting up to 2 slave devices
  - Back-to-back character transmission/reception mode
  - Loop back mode for testing
- Five UARTs (UART0, UART1, UART2, UART3, UART4)
  - Full-duplex operation
  - 5-, 6-, 7- or 8-bit characters with optional no parity or even or odd parity and with 1, 1½, or 2 stop bits
  - 64x8 bit transmit FIFO and 64x11bit receive FIFO
  - Independently controlled transmit, receive (data ready or timeout), line status interrupts
  - Internal diagnostic capability Loopback control and break, parity, overrun and framing-error is provided
  - Separate DMA requests for transmit and receive data services in FIFO mode
  - Supports modem flow control by software or hardware
  - Slow infrared asynchronous interface that conforms to IrDA specification
- Two MMC/SD/SDIO controllers (MSC0, MSC1)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Fully compatible with the MMC System Specification version 4.2
- Support SD Specification 3.0
- Support SD I/O Specification 1.0 with 1 command channel and 4 data channels
- Consumer Electronics Advanced Transport Architecture (CE-ATA – version 1.1)
- Maximum data rate is 50MBps
- Support MMC data width 1bit ,4bit and 8bit
- Built-in programmable frequency divider for MMC/SD bus
- Built-in Special Descriptor DMA
- Mask-able hardware interrupt for SDIO interrupt, internal status and FIFO status
- 128 x 32 built-in data FIFO
- Multi-SD function support including multiple I/O and combined I/O and memory
- IRQ supported enable card to interrupt MMC/SD controller
- Single or multi block access to the card including erase operation
- Stream access to the MMC card
- Supports SDIO read wait, interrupt detection during 1-bit or 4-bit access
- Supports CE-ATA digital protocol commands
- Support Command Completion Signal and interrupt to CPU
- Command Completion Signal disable feature
- The maximum block length is 4096bytes

- USB 2.0 OTG interface
  - Complies with the USB 2.0 standard for high-speed (480 Mbps) functions and with the On-The-Go supplement to the USB 2.0 specification
  - Operates either as the function controller of a high- /full-speed USB peripheral or as the host/peripheral in point-to-point or multi-point communications with other USB functions
  - Supports Session Request Protocol (SRP) and Host Negotiation Protocol (HNP)
  - UTMI+ Level 3 Transceiver Interface
  - Soft connect/disconnect
  - 8 endpoints in device mode, 16 channels for host mode.
  - Dedicate FIFO
  - Supports control, interrupt, ISO and bulk transfer

- OTP Slave Interface
  - Total 4KB.

1.2.9 Bootrom

32KB Boot ROM memory

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 2

CORE FUNCTIONS

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 CPU

2.1 Overview

The CPU of this M200-SOC is a binary star implementation with a unified L2 cache. It has following obvious features contrasting with past implementations of XBurst-1.

- one little-core with low-power design orientation
- one big-core with high frequency design orientation
- 32KB L1-Icache, 32KB L1-Dcache for each core
- DCSC (Dual Core Schedule & Control), a dedicated module to control the binary star MPcores.
- 512 KB unified L2 cache supporting WB and WT attribute with enhanced hardware prefetch

2.2 M200-CPU

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3 VPU

3.1 Overview

Video Processing Unit (VPU) core in this chip is dedicated for video decoding and encoding. VPU embeds an XBurst® CPU core (named AUX in VPU) and application specified hardware accelerators for common video compress/decompress algorithms, which includes Stream Parser, Motion Compensation, Motion Estimation, Quant/Inverse Quant, DCT/Inverse DCT and De-block engines. Also VPU scheduler (named SCH in VPU) is embedded so that AUX can be standby and SCH would manage the whole VPU control and data flow for most popular video formats. That is to say, we can get preferable trade-off of high performance and low power consumption of most popular video formats by SCH and flexible programmability for other video formats or even some other arithmetic such as image processing by AUX.

Key standards performance of VPU in the chip:

- MPEG-2 decoding up to 1080P 30fps
- VC-1 decoding up to 1080P 30fps
- H.264 decoding up to 1080P 30fps
- VP8 decoding up to 1080P 30fps
- MPEG-4 decoding up to 1080P 30fps
- RV9 decoding up to 1080P 30fps
- H.264 encoding up to 720P 30fps
- JPEG compressing/decompressing up to 50Mega-pixels per second (baseline)

3.2 Block Diagram

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.3 Features of VPU

Table 3-1 VPU Features

Item	Features
XBurst® core(AUX)	● XBurst-1 CPU – Industry standard RISC instruction set – 32 32-bit general purpose registers, no shadow GPR – Physical address accessing directly ● Media Extension Unit (MXU) – Ingenic SIMD instruction set II – fully pipelined
Tightly Coupled Sharing Memory (TCSM)	● TCSM – Size: 32K bytes – AHB slave interface supports external DMA access **NOTE:** TCSM is coupled with VPU XBurst® core internally.
Scratch RAM (SRAM)	● Size: 16K bytes ● AHB slave interface supports external DMA access
VPU Gerenal Purpose DMA(VDMA)	● Descriptor based DMA ● VPU self automatic configure supported
Parser (SDE)	Parser is a bit-stream engine (named as SDE) in VPU ● H.264 Context adaptive binary arithmetic (CABAC) decoding support, PCM not supported (treated as bitstream error) ● H.264 Context adaptive variable length (CAVLC) decoding support, PCM not supported (treated as bitstream error) ● VC-1 bitstream decoding support ● MPEG2 bitstream decoding support ● VP8 bitstream decoding support ● H.264 Context adaptive binary arithmetic encoding support
Motion (MCE)	Motion serves as a COMBO engine of compensation and estimation (named as MCE) in VPU ● Reference data cache embed ● Descriptor based task fetching ● Programmable processing size from 2x2 to 16x16 (in estimation the size is from 4x4 to 16x16) ● Programmable interplation filter from 2-tap to 8-tap ● Programmable sub-pixel accuracy from 1/2-pixel to 1/8-pixel (in estimation searching accuracy is supported from integer to 1/4-pixel) ● Interlaced mode support ● Intensity compensation support ● Weighted prediction support

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	● Automatic rotation support for rotated referenced pictures ● Automatic expanding support for outside frame's reference ● Configurable searching strategy in estimation
Recover (VMAU)	Recover is a Matrix Arithmetic Unit in VPU (named as VMAU), it serves for pixel's recovery and reconstruction during vedio decoding and encoding. ● Descriptor based task fetching ● Configurable format intra prediction support ● Configurable format Inverse quant support ● Configurable format IDCT support ● Residual add for pixel's recovery ● Estimation subtract and pixel reconstruction in encoding flow
Deblock (DBLK)	● Descriptor based task fetching ● RealVideo(RV9) in loop filter support ● H.264 in loop filter support, FMO/MBAFF not supported ● VC-1 deblock support ● VP8 deblock support ● tile-base write buffer for DMA out
Encoder front-end (EFE)	● tile-based YUV RAW fetching and delievering ● raster YUV420p RAW fetching and delievering ● raster YUV420sp RAW fetching and delievering
JPEG codec (JPGC)	● baseline profile JPEG compressing/decompressing ● quant table configurable ● huffman table configurable
Scheduler (SCH)	Scheduler is a special unit which is used to manage internel functional engines' handshake. ● 4 programable channels ● 2 programable groups
Translation look-aside buffer (TLB)	● 4Kbyte based pagesize mapping ● TLB miss automatic entry filling
Interrupt control center (INTC)	● Task end interrupt (frame/slice end) ● TLB error interrupt ● Bitstream error interrupt ● VDMA ACFG error interrupt

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.4 Internal physical address base definition

Table 3-2 VPU Internal physical address base definition

Module	Physical address base
SCH	0x1320_0000
AUX	0x132A_0000
TCSM	0x132C_0000
SRAM	0x132F_0000
VDMA	0x1321_0000
EFE	0x1324_0000
MCE	0x1325_0000
DBLK	0x1327_0000
VMAU	0x1328_0000
SDE	0x1329_0000
JPGC	0x132E_0000

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 GPU

- 3D Graphic
  - OpenGL ES 2.0 / 1.1 compliance, including extensions; OpenVG 1.1
  - Low CPU loading
  - Up to 12 programmable elements per vertex
  - Dependent texture operation with high-performance
  - Alpha blending
  - Depth and stencil compare
  - Support for 8 fragment shader simultaneous textures
  - Support for 4 vertex shader simultaneous textures
  - Point sampling, bi-linear sampling, tri-linear filtering, and cubic textures
  - Resolve and fast clear
  - 8k x 8k texture size and 8k x 8k rendering target
  - 4 Vertex DMA streams
  - Supports YUV formats in display output (YUY2 4:2:2)
  - MMU functionality supported
- 2D feature
  - Bit BLT and stretch BLT
  - Rectangle fill and clear
  - Line drawing
  - High-performance stretch and shrink
  - Monochrome expansion for text rendering
  - ROP2, ROP3, ROP4
  - Alpha blending including Java 2 Porter-Duff compositing blending rules
  - 32k x 32k coordinate system
  - 90, 180, and 270 degrees rotation
  - Transparency by monochrome mask, chroma key, or pattern mask
  - Support 2x2 in 4x4 tile format
  - A8/RG16 output with rotation in FilterBlit and BitBlit
- Multi Source Blending
  - Full support for Multi source blending with variable block size to improve BW and reduce SW overhead.
  - Up to 8 sources are supported.
  - Programmable block size guarantees cache efficiency so each source is read once and each destination is written once.
  - Supports 90, 180, 270 degree rotation with different block size for higher cache efficiency.
  - Supports independent source rotation with blending.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5 ISP

The ISP is a complete 20 megapixel (MP) video and still picture input unit in this chip is dedicated for image and video process for cameras. The high-performance camera pipeline features image processing, 3D processing, wide dynamic range processing, digital image stabilization and scaling functions. The integrated image signal processor (ISP) supports simple CMOS sensors with no image pre-processing.

The ISP supports resolutions ranging from QCIF up to 20 MP in a single low-cost/low-power design. To effectively deliver high resolutions, the processor features sophisticated bad pixel detection/correction and noise reduction techniques to ensure image quality even when paired with low cost, high-resolution CMOS sensors commonly found in mobile devices

The ISP supports both parallel and serial input interfaces compatible with most CMOS sensors and several CPU and memory system interfaces. The CMOS sensor sends data to the ISP via parallel digital video interfaces. In addition, camera serial interface (CSI-2) MIPI is supported. When communicating with a CPU or memory, 32-bit ARM AMBA AHB master and ARM AMBA AXI are supported.

Key standards performance of ISP in the chip:

- Still capture mode
  5168x3876@30fps or @15fps 3D
- Video mode
  Maximum size: 2584x1938
  2560x1600@30fps
  1080P@60fps or @30fps 3D
  720P@120fps or @60fps 3D

5.1 Block Diagram

Figure 5-1 ISP Block Diagram

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.2 Features of ISP

Table 5-1 ISP Features

Item	Features
System level	● 32-bit AHB slave interface ● 64-bit AXI master interface ● Two MIPI 4-lane IDI input interface (can be used as one 8-lane, still capture only) ● Two serial SCCB interfaces ● Parallel 10-bit DVP input interface _
Format	● Camera input RAW8, RAW10, RAW12, RAW14 YUV422 (only support scale up/down, crop, SDE) RGB565, RGB888, JPEG (bypass mode) ● Read from memory RAW8, RAW10, RAW12 YUV422 (only support scale up/down, crop, SDE) ● ISP output YUV422 (YUYV/UYVY), YUV420 RGB888, RGB565 RAW8, RAW10, RAW12, RAW14
Basic ISP features	● AECAGC, Auto exposure and gain control ● AWB, Auto white balancing (color temperature based) ● AFC, Auto focus control (25 points and single point) ● DPC, Defect pixel correction ● LENC, Lens shading correction (color temperature based) ● CCM, Color correction matrix (color temperature based) ● 50/60Hz flicker detection and cancellation ● Enhanced color interpolation (De-mosaic) ● Gamma correction ● Noise reduction ● Sharpness enhancement ● Brightness, saturation, hue and contrast control ● Digital zoom (zoom in 4x, zoom out 16x) ● ARC, auto dynamic range control (HDR only) ● Flash light control ● Mechanical shutter control
Enhanced ISP features	● 3D video and still capture ● HDR/EDR video and still capture ● Support any size still image capture and process (limited by

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	memory size) ● Low latency still image capture (from preview to still capture) ● Video mode auto-focus ● Video anti-shaking ● Software/hardware/optical combined EDOF ● ROI support ● Support RAW data capture (up to 16 bit/pixel) ● 3D support Multiple synchronization methods (support any sensor) Multiple 3D combinations A. Two cameras B. One array by array camera C. One pixel by pixel camera Multiple output formats A. Left/Right side by side B. Up/Down frame by frame C. Pixel by pixel (full size or half size) ● EDR support RAW data process with EDR ● HDR support Multiple frame HDR still image One sensor HDR video A. Split pixel B. Two lines by two lines Combined software & hardware HDR ● Special effects support Black white Negative Sepia Sketch Watercolor Ink Cartoon ● Color ink
Software features	● Face detection ● Smile detection ● Eye blinking detection ● 1D/2D bar code reading ● YUV data process with EDR ● Special effects Frame

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Mask Superimpose

5.3 Internal ram physical address

Table 5-2 ISP Internal ram physical address

Name	Physical address base	Type Address	Size (bit)
CS_ROM	0x000000~0x01BFFF	Line Buffer	28672x32 = 917504
Data RAM	0x01C000~0x01EFFF	Line Buffer	3072x32 = 98304
I-Cache	0x01F000~0x01F3FF	Line Buffer	256x32 = 8192
D-Cache	0x01F400~0x01F7FF	Line Buffer	256x32 = 8192
I-Tag	0x01F800~0x01FCFF	Line Buffer	64x23 = 1472
D-Tag	0x01FD00~0x01FFFF	Line Buffer	64x23 = 1472

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 3

DISPLAY/CAMERA/AUDIO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6 LCD Controller

6.1 Overview

The LCD controller has the capabilities to driving the latest TFT LCD panels. It also supports some special TFT panels used in consuming electronic products. The controller performs the basic memory based frame buffer to LCD panel data transfer through usage of a dedicated DMA controller. And OSD (two layers pictures overlaying) is also supported for LCD controller.

Features:

- Basic Features
  - Support panel(TFT, SLCD)
  - Display size up to 1280x720@60Hz(BPP24)

- Colors Supports
  - Encoded pixel data of 16, 18 or 24 BPP in TFT mode
  - Support up to 16,777,216 (16M) colors in TFT mode
  - Support 24 BPP packed data

- Panel Supports
  - Support 16-bit parallel TFT panel
  - Support 18-bit parallel TFT panel
  - Support 24-bit serial TFT panel with 8 data output pins
  - Support 24-bit parallel TFT panel
  - Support Delta RGB panel
  - Support SLCD panel

- OSD Supports
  - Supports one single color background
  - Supports two foregrounds, and every size can be set for each foreground
  - Supports one transparency for the whole graphic
  - Supports one transparency for each pixel in one graphic
  - Supports color key and mask color key
  - Supports porter-duff blending

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.2 Pin Description

Table 6-1 LCD Controller Pins Description

Name	I/O	Description
Lcd_pclk	Output	Display device pixel clock
Lcd_vsync	Output	Display device vertical synchronize pulse
Lcd_hsync	Output	Display device horizontal synchronize pulse
Lcd_de	Output	Display device data enable Pin
Lcd_d[17:0]	Output	Display device data pins
lcd_lo6_o[5:0]	Output	Display device data pins use in 24 bit parallel mode.
Lcd_spl*[1]	Output	Programmable special pin for generating control signals
Lcd_cls*[1]	Output	Programmable special pin for generating control signals
Lcd_ps*[1]	Output	Programmable special pin for generating control signals
Lcd_rev*[1]	Output	Programmable special pin for generating control signals

*1 : gray pins only exists in previous LCDC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.3 Block Diagram

Figure 6-1 Block Diagram when use OSD mode

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 6-2 Block Diagram of not use OSD mode (this function is not supported in current version)

Figure 6-3 Block Diagram of HDMI interface(no HDMI in current version)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.4 LCD Display Timing

Figure 6-4 Display Parameters

NOTES:

1 VPS === 0
VSYNC pulse always start at point (0,0)

2 H: Horizontal V: Vertical T: Total
D: Display Area P: Pulse
S: Start point E: End point

In the (H, V) Coordinates:

1 The gray rectangle (0, 0) to (HT, VT) is “Virtual Area”.
2 The blue rectangle (HDS, VDS) to (HDE, VDE) is “Display Area”.
3 VPS, VPE defines the VSYNC signal timing. (VPS always be zero)
4 HPS, HPE defines the HSYNC signal timing.

All timing parameters start with “H” is measured in lcd_pclk ticks.
All timing parameters start with “V” is measured in lcd_hsync ticks.

This diagram describes the general LCD panel parameters, these can be set via the registers that describes in next section.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.5 OSD Graphic

Figure 6-5 OSD Graphic

NOTES:

1 Background is one single color and the size is the full screen.

2 The size of foregrounds can be every size smaller than background.

3 The order of the graphic is as follows:

- a Top layer: Foreground 0.
- b Middle layer: Foreground 1.
- c Bottom layer: Background0 & Background1.

6.5.1 Color Key

This function gives user a method to implement irregular display window. User can make foreground 0 and foreground 1 to different shape. The color key has two implements mode that called color key and mask color key.

Color Key mode is meant to mask a chosen color and show others.

Mask Color Key mode is meant to only show a chosen color and mask others.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Not use color key function

Color key mode

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Mask color key mode

6.6 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 6-2 Registers Memory Map-Address Base

Name	Base	Description
LCDC	0x13050000	Address base of LCDC

Table 6-3 LCD Controller Registers Description

Name	RW	Description	Address offset	Access Size
LCDCFG	RW	configure register	0x0000	32
LCDCTRL	RW	control register	0x0030	32
LCDSTATE	RW	status register	0x0034	32
LCDOSDC	RW	OSD configure register	0x0100	32
LCDOSDCTRL	RW	OSD control register	0x0104	32
LCDOSDS	RW	OSD STATUS register	0x0108	32
LCDBGC0	RW	background0 color register	0x010C	32
LCDBGC1	RW	background1 color register	0x02C4	32
LCDKEY0	RW	Foreground Color Key Register 0	0x0110	32
LCDKEY1	RW	Foreground Color Key Register 1	0x0114	32
LCDALPHA	RW	ALPHA Register	0x0118	32
LCDRGBC	RW	RGB Control	0x0090	32
LCDVAT	RW	Virtual Area Setting	0x000C	32
LCDDAH	RW	Display Area Horizontal Start/End Point	0x0010	32
LCDDAV	RW	Display Area Vertical Start/End Point	0x0014	32
LCDXYP0	RW	Foreground 0 XY Position Register	0x0120	32
LCDXYP1	RW	Foreground 1 XY Position Register	0x0124	32
LCDSIZE0	RW	Foreground 0 Size Register	0x0128	32
LCDSIZE1	RW	Foreground 1 Size Register	0x012C	32
LCDVSYNC	RW	Vertical Synchronize Register	0x0004	32
LCDHSYNC	RW	Horizontal	0x0008	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Synchronize Register		
LCDPS*1	RW		0x0018	32
LCDCLS*1	RW		0x001C	32
LCDSPL*1	RW		0x0020	32
LCDREV*1	RW		0x0024	32
LCDIID	R	Interrupt ID Register	0x0038	32
LCDDA0	RW	Descriptor0 Address Registers	0x0040	32
LCDSA0	R	Source Address Registers0	0x0044	32
LCDFID0	R	Frame ID Registers0	0x0048	32
LCDCMD0	R	DMA0 Command Registers	0x004C	32
LCDOFFS0	R	DMA0 OFFSIZE Registers	0x0060	32
LCDPW0	R	DMA0 Page Width Registers	0x0064	32
LCDCNUM0/LCDPOS0	R	DMA0 Commend Counter Registers	0x0068	32
LCDDESSIZE0	R	Foreground0 x Size in Descriptor	0x006C	32
LCDDA1*2	RW	Descriptor1 Address Registers	0x0050	32
LCDSA1*2	R	Source Address Registers1	0x0054	32
LCDFID1*2	R	Frame ID Registers1	0x0058	32
LCDCMD1*2	R	DMA1 Command Registers	0x005C	32
LCDOFFS1*2	R	DMA1 OFFSIZE Registers	0x0070	32
LCDPW1*2	R	DMA1 Page Width Registers	0x0074	32
LCDCNUM1/LCDPOS1*2	R	DMA1 Commend Counter Registers	0x0078	32
LCDDESSIZE1*2	R	Foreground1 x Size in Descriptor	0x007C	32
LCDPCFG	RW	Priority level threshold configure Register	0x02C0	32

NOTES:

1 *1: These registers are only used for SPECIAL TFT panels.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 *[2]: These registers are only used for DMA channel 1 in OSD mode for TFT panels.

6.6.1 Configure Register (LCDCFG)

	LCDCFG																															BASE+0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCDPIN	TVEPEH	FUHOLD	NEWDES	PALBP	TVEN	RECOVER	MIPIDAEN	PSM	CLSM	SPLM	REVM	HSYNM	PCLKM	INVDAT	SYNDIR	PSP	CLSP	SPLP	REVP	HSP	PCP	DEP	VSP	TFT_BIT		PDW		MODE			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	LCDPIN[*1]	LCD PIN Select bit. It is used to choose the function of LCD PINS or SLCD PINS. The function of pins is as follows.when use slcd in current LCDC with GPIO in function2, you need not to care this bit at all as slcd's pin in connected to GPIO seperatly **LCDPIN** — **PIN SELECT** 0 — LCD PIN 1 — SLCD PIN	RW
30	TVEPEH	TVE PAL enable extra_halfline signal.	
29	FUHOLD	**KEEP THIS BIT TO 0.**	RW
28	NEWDES	indicate use new 8 words descriptor or not. Set this bit to 1	RW
27	PALBP	Indicate bypass pal in BPP8, and in OSD mode, set this bit to 1 is also bypass data format and alpha blending. 0: not use PAL; 1: use PAL. This bit is fixed to 0 in this version	R
26	TVEN	Indicate the terminal is LCD panel or TV.	RW
25	RECOVER	Auto recover when output FIFO under run. 0: disable; 1: enable.	RW
24	MIPIDAEN	MIPI DSI data enable in video mode 1： enable 0：disable， all data fixed to 0	RW
23	PSM	PS signal mode bit. 0: enabled; 1: disabled.	RW
22	CLSM	CLS signal mode bit. 0: enabled; 1: disabled.	RW
21	SPLM	SPL signal mode bit. 0: enabled; 1: disabled.	RW
20	REVM	REV signal mode bit. 0: enabled; 1: disabled. Note: PSM, CLSM, SPLM and REVM are useless now, just keep them 0.	RW
19	HSYNM	H-Sync signal polarity choice function. 0: enabled; 1: disabled.	RW
18	PCLKM	Dot clock signal polarity choice function. 0: enabled; 1: disabled.	RW
17	INVDAT	Inverse output data. 0: normal; 1: inverse.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16	SYNDIR	V-Sync and H-Sync direction. 0: output; 1: input. As vsync and hsync are always outputs in current version , you must keep this bit zero.	RW
15	PSP	PS pin reset state.	RW
14	CLSP	CLS pin reset state.	RW
13	SPLP	SPL pin reset state.	RW
12	REVP	REV pin reset state. Note: PSP, CLSP, SPLP and REVP are also useless now, keep them 0.	RW
11	HSP	H-Sync polarity. 0: active high; 1: active low.	RW
10	PCP	Pix-clock polarity. 0: data translations at rising edge 1: data translations at falling edge	RW
9	DEP	Data Enable polarity. 0: active high; 1: active low.	RW
8	VSP	V-Sync polarity. 0: leading edge is rising edge 1: leading edge is falling edge	RW
7:6	TFT_BIT	this 2 bits decide TFT's data width 2'bx1:24-bit TFT 2'b00:16-bit TFT 2'b10:18-bit TFT	
5:4	PDW	Some parameters for STN, keep it zero in current version	
3:0	MODE	Display Device Mode Select/Output mode.	RW

	LCD Panel
0000	Generic 16-bit/18-bit/24-bit Parallel TFT Panel
0001	Special TFT Panel Mode1
0010	Special TFT Panel Mode2
0011	Special TFT Panel Mode3
0100	Non-Interlaced TV out
0101	Reserved
0110	Interlaced TV out
0111	Reserved
1000	Reserved
1001	Reserved
1010	Reserved
1011	Reserved
1100	8-bit Serial TFT
1101	SLCD
1110	Reserved
1111	Reserved

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

*1.

LCDPIN	PIN25	PIN24	PIN23	PIN22	PIN21	PIN20	PIN19	PIN18	PIN17-0
0	LCD PCLK	LCD VSYNC	LCD HSYNC	LCD DE	LCD REV	LCD PS	LCD CLS	LCD SPL	LCD D [17:0]
1	SLCD CLK	SLCD WR	SLCD DC	--	--	--	--	--	SLCD D [17:0]

1 The direction of PIN25 is set by register LPCDR.LCS in CPM SPEC.

2 The direction of PIN23 and PIN23 are set by register LCDCFG.SYNDIR.

6.6.2 Control Register (LCDCTRL)

	LCDCTRL																															BASE+0x0030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PINMD	BST			OUTRGB	OFUP	STN_FRC		PDD								VGA	DACTEST	EOFM	SOFM	OFUM	IFUM0	IFUM1	LDDM	QDM	BEDN	PEDN	DIS	ENA	BPP0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	PINMD	This register set Pin distribution in 16-bit parallel mode. 0: 16-bit data correspond with LCD_D[15:0] 1: 16-bit data correspond with LCD_D[17:10], LCD_D[8:1]	RW
30:28	BST	Burst Length Selection. 000 — 4 word 001 — 8 word 010 — 16 word 011 — 32 word 100 — 64 word	RW
27	OUTRGB	Bpp16 RGB output mode. 0: RGB565; 1: RGB555. When 16-bit panel is used, this bit is valid	RW
26	OFUP	Output FIFO under run protection. 0: disable; 1: enable. Note: this bit is useless now, not care about it	RW
25:24	STN_FRC	STN FRC algorithm select Note: it's useless now, not care about this bit.	RW
23:16	PDD	keep these bits to 0.	RW
15	VGA	keep this bit to 0.	RW
13	EOFM	Mask end of frame interrupt. 0: INT-disabled; 1: INT-enabled.	RW
12	SOFM	Mask start of frame interrupt. 0: INT-disabled; 1: INT-enabled.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11	OFUM	Mask out FIFO under run interrupt. 0: INT-disabled;1: INT-enabled.	RW
10	IFUM0	Mask in FIFO 0 under run interrupt. 0: INT-disabled; 1: INT-enabled.	RW
9	IFUM1	Mask in FIFO 1 under run interrupt. 0: INT-disabled; 1: INT-enabled.	RW
8	LDDM	Mask LCD disable done interrupt. 0: INT-disabled; 1: INT-enabled.	RW
7	QDM	Mask LCD quick disable done interrupt. 0: INT-disabled; 1: INT-enabled.	RW
6	BEDN	Endian selection. 0: same as system Endian; 1: reverse endian format.	RW
5	PEDN	Endian in byte. 0: msb first; 1: lsb first.	RW
4	DIS	Disable controller indicate bit. 0: enable; 1: in disabling or disabled.	RW
3	ENA	Enable controller. 0: disable; 1: enable.	RW
2:0	BPP0	Bits Per Pixel of foreground0.	R

	Bits Per Pixel
000	Reserved
001	Reserved
010	Reserved
011	Reserved
100	15/16bpp
101	18bpp/24bpp
110	24bpp compressed
111	30bpp

Note:clear LCDCTRL.ENA and wait LCDSTATE.QD to be high is quick disable, and this mode is used when panel is smart lcd; set LCDCTRL.DIS and wait LCDSTATE.LDD to be high is normal disable and this mode is used when panel is TFT

6.6.3 Status Register (LCDSTATE)

	LCDSTATE																															BASE+0x0034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								QD	Reserved	EOF	SOF	OUF	IFU0	IFU1	LDD
RST	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:8	Reserved	Writing has no effect, read as zero.	R
7	QD	LCD Quick disable. 0: not been quick disabled; 1: quick disabled done. It can only be set high by hardware, and software can only write 0	RW
6	Reserved	Writing has no effect, read as zero.	R
5	EOF	End of Frame indicate bit. set 1 by hardware,write 0 by software	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4	SOF	Start of Frame indicate bit. set 1 by hardware,write 0 by software	RW
3	OUF	Out FIFO under run. set 1 by hardware,write 0 by software	RW
2	IFU0	In FIFO 0 under run. set 1 by hardware,write 0 by software	RW
1	IFU1	In FIFO 1 under run. set 1 by hardware,write 0 by software	RW
0	LDD	LCD disable. 0: not been normal disabled; 1: been normal disabled. set 1 by hardware,write 0 by software	RW

Note: EOF and SOF can only be set high in TFT mode by hardware, software can only write 0

6.6.4 OSD Configure Register (LCDOSDC)

	LCDOSDC																															BASE+0x0100
Bit									23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
									PREMULTI1	COEF_SEL1		PREMULTI0	COEF_SEL0		ALPHAMD1	64bit_r	SOFM1	EOFM1	Reserved		SOFM0	EOFM0	DENDM	Reserved				F1EN	F0EN	ALPHAEN	ALPHAMD0	OSDEN
RST									0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
23	PREMULTI1	Premulti enable of foreground1 : 0 : data has been premultied and don't need premulti 1 : data should be premultied by lcd	R
22:21	COEF_SLE1	Select coefficient for foreground1 : 00 : 0 01 : 1 10 : alpha0 11 : 1-alpha0	R
20	PREMULTI0	Premulti enable of foreground0 : 0 : data has been premultied and don't need premulti 1 : data should be premultied by lcd	R
19:18	COEF_SLE0	Select coefficient for foreground0 : 00 : 0 01 : 1 10 : alpha1 11 : 1-alpha1	R
17	ALPHAMD1	Alpha blending mode for foreground1. 0: One transparency for the whole graphic, and the LCDALPHA register is used for transparency 1: One transparency for each pixel in one graphic, and the alpha value is coming from each pixel data	R
16	64bit_r	high means used dual OSD, low means use only one OSD, set high the lcdc will work stronger	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15	SOFM1	Start of frame interrupt mask for foreground 1.	RW
14	EOFM1	End of frame interrupt mask for foreground 1.	RW
13：12	Reserved	Writing has no effect, read as zero.	R
11	SOFM0	Start of frame interrupt mask for foreground 0.	RW
10	EOFM0	End of frame interrupt mask for foreground 0.	RW
9	DENDM	Display end interrupt mask	RW
8:5	Reserved	Writing has no effect, read as zero.	R
4	F1EN	1: Foreground 1 is enabled 0: Foreground 1 is disabled	R
3	F0EN	1: Foreground 0 is enabled 0: Foreground 0 is disabled.	R
2	ALPHAEN	1: Alpha blending is enabled 0: Alpha blending is disabled Keep this bit to 1	RW
1	ALPHAMD0	Alpha blending mode for foreground0. 0: One transparency for the whole graphic, and the LCDALPHA register is used for transparency 1: One transparency for each pixel in one graphic, and the alpha value is coming from each pixel data	R
0	OSDEN	OSD mod enable. 1: enabled. And you can use F0 F1 0: disabled Set this bit to 1	RW

Note: Frame's interrupt is only valid when in TFT mode. we suggest to keep LCDOSDC.EOFM0/EOFM1/SOFM0/SOFM1 to be zero, and if need frame start and end interruput, you set LCDCTRL.SOFM LCDCTRL.EOFM. and relative layer of SOFINT and EOFINT should be set high in LCDCMD. Except frame's interrupt, there are so many interruptions:

two input fifo underrun int(LCDCTRL.IFUM0/IFUM1); output fifo underrun int(LCDCTRL.OFU), quick/normal disable int(LCDCTRL.QDM/LDDM), and display end int(LCDOSDCTRL.DENDM)

6.6.5 OSD Control Register (LCDOSDCTRL)

	LCDOSDCTRL																															BASE+0x0104
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																IPU_CLKEN	Reserved									RGB0	RGB1	Reserve	BPP1		
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
15	IPU_CLKEN	this bit is write and read able but useless	RW
5	RGB0	Bpp16 RGB mode of foreground0. 0: RGB565; 1: RGB555.	R
4	RGB1	Bpp16 RGB mode of foreground1. 0: RGB565; 1: RGB555.	R
2:0	BPP1	Bits Per Pixel of OSD channel 1. 000 Reserved 001 Reserved 010 Reserved 011 Reserved 100 15/16bpp 101 18bpp/24bpp 110 24bpp compressed 111 30bpp	R

6.6.6 OSD State Register (LCDOSDS)

	LCDOSDS																															BASE+0x0108
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SOF1	EOF1	Reserved		SOF0	EOF0	Reserved	DEND	Reserved							
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	SOF1	Start of frame flag for foreground 1.	RW
14	EOF1	End of frame flag for foreground 1.	RW
13:12	Reserved	Writing has no effect, read as zero.	R
11	SOF0	Start of frame flag for foreground 0.	RW
10	EOF0	End of frame flag for foreground 0.	RW
9	Reserved	Writing has no effect, read as zero.	R
8	DEND	Display end, 1: display end; 0 : not end hardare set 1, and software clear0	RW
7:0	Reserved	Writing has no effect, read as zero.	R

6.6.7 Background0 Color Register (LCDBGC0)

	LCDBGC0																															BASE+0x010C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved								Red [7:0]								Green [7:0]								Blue [7:0]							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:27	Reserved	Writing has no effect, read as zero.	R
23:16	Red	Red part or Y part of background0.	RW
15:8	Green	Green part or Cb part of background0.	RW
7:0	Blue	Blue part or Cr part of background0.	RW

6.6.8 Background1 Color Register (LCDBGC1)

	LCDBGC1																															**BASE+0x02C4**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								Red [7:0]								Green [7:0]								Blue [7:0]							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:27	Reserved	Writing has no effect, read as zero.	R
23:16	Red	Red part or Y part of background1.	RW
15:8	Green	Green part or Cb part of background1.	RW
7:0	Blue	Blue part or Cr part of background1.	RW

6.6.9 Foreground Color Key Register 0 (LCDKEY0)

	LCDKEY0																															**BASE+0x0110**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KEYEN	KEYMD	Reserved(but RW)						Red [7:0]								Green [7:0]								Blue [7:0]							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	KEYEN	The enable bit of color key for foreground 0.	RW
30	KEYMD	Color key mod. 0: color key; 1: mask color key.	RW
29:27	Reserved	Read Write able, but useless	RW
23:16	Red	Red part of color key for foreground 0.	RW
15:8	Green	Green part of color key for foreground 0.	RW
7:0	Blue	Blue part of color key for foreground 0.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.6.10 Foreground Color Key Register 1 (LCDKEY1)

	LCDKEY1																															BASE+0x0114
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KEYEN	KEYMD	Reserved(but RW)						Red [7:0]								Green [7:0]								Blue [7:0]							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	KEYEN	The enable bit of color key for foreground 1.	RW
30	KEYMD	Color key mod. 0: color key; 1: mask color key.	RW
29:27	Reserved	Read Write able, but useless	RW
23:16	Red	Red part of color key for foreground 1.	RW
15:8	Green	Green part of color key for foreground 1.	RW
7:0	Blue	Blue part of color key for foreground 1.	RW

6.6.11 ALPHA Register (LCDALPHA)

	LCDALPHA																															BASE+0x0118
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	ALPHA 1[7:0]								ALPHA 0[7:0]							
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:8	ALPHA1	The alpha value of foreground1 for one graphic with one transparency.	R
7:0	ALPHA0	The alpha value of foreground0 for one graphic with one transparency.	R

The formula of alpha blending is as follows:

$$PIXEL_{0/1} = [Alpha_{0/1} * Foreground_{0/1}]$$

$$F_{0/1} = [0,1, Alpha_{1/0}, (1 - Alpha_{1/0})]$$

$$NewPixel = [F_0 * PIXEL0 + F_1 * PIXEL1]$$

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.6.12 RGB Control (LCDRGBC)

	LCDRGBC																															BASE+0x0090
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	RGBDM	DMM	Reserved					YCC	RGBFMT	OddRGB			Reserved	EvenRGB		
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	RGBDM	RGB with dummy data enable. Only useful for RGB serial mode. If this bit set to 1, the one pixel include 4 clock periods, that Red, Green, Blue and Dummy data. Dummy is equal to 0. 0: Disable; 1: Enable.	RW
14	DMM	RGB dummy mode. 0: R-G-B-Dummy 1: Dummy-R-G-B	
13:9	Reserved	Writing has no effect, read as zero.	R
8	YCC	Change 444 to 422. 0: not change; 1: change to YUV. This bit only use in OSD mode. Change 444 data to 422 and sent to TV encoder. Please notice that the data will be translated as 16 bits parallel. And only half of it will be transfer. (YCb or YCr in one pixel). If you not use OSD mode and TV encoder, please set this bit to 0. When use this function with IPU transfer data to an interlaced TV, please set IPU output as RGB 888, and OSDBPP to 24. or IPU output data as PACKAGE(YCbYCr) and OSDBPP to 16.	
7	RGBFMT	RGB format enable : 0: disable; 1: enable.	RW
6:4	OddRGB	Odd line serial RGB data arrangement, useful for both serial mode and parallel mode 000 RGB 001 RBG 010 GRB 011 GBR 100 BRG 101 BGR 110 Reserved 111 Reserved	RW
3	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2:0	EvenRGB	Even line serial RGB data arrangement, useful for both serial mode and parallel mode	RW

	RGB mode
000	RGB
001	RBG
010	GRB
011	GBR
100	BRG
101	BGR
110	Reserved
111	Reserved

Note: when LCDCFG.MODE is 8-bit serial TFT or LCDRGBC. RGBFMT is high, OddRGB and EvenRGB is valid.

6.6.13 Virtual Area Setting (LCDVAT)

	LCDVAT																															BASE+0x000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				HT												Reserved				VT											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	HT	Horizontal Total size. (in dot clock, sum of display area and blank space)	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	VT	Vertical Total size. (in line clock, sum of display area and blank space)	RW

6.6.14 Display Area Horizontal Start/End Point (LCDDAH)

	LCDDAH																															BASE+0x0010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				HDS												Reserved				HDE											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	HDS	Horizontal display area start. (in dot clock)	RW
15:12	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11:0	HDE	Horizontal display area end. (in dot clock)	RW

6.6.15 Display Area Vertical Start/End Point (LCDDAV)

	LCDDAV																															BASE+0x0014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				VDS												Reserved				VDE											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	VDS	Vertical display area start position. (in line clock)	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	VDE	Vertical display area end position. (in line clock)	RW

6.6.16 Foreground 0 XY Position Register (LCDXYP0)

	LCDXYP0																															BASE+0x0120
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				YPOS												Reserved				XPOS											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	YPOS	The Y position of top-left part for foreground 0.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	XPOS	The X position of top-left part for foreground 0.	R

6.6.17 Foreground 1 XY Position Register (LCDXYP1)

	LCDXYP1																															BASE+0x0124
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				YPOS												Reserved				XPOS											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	YPOS	The Y position of top-left part for foreground 1.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	XPOS	The X position of top-left part for foreground 1.	R

6.6.18 Foreground 0 Size Register (LCDSIZE0)

	LCDSIZE0																															BASE+0x0128
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				Height												Reserved				Width											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	Height	The height-1 of foreground 0.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	Width	The width-1 of foreground 0.	R

6.6.19 Foreground 1 Size Register (LCDSIZE1)

	LCDSIZE1																															BASE+0x012c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				Height												Reserved				Width											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	Height	The height-1 of foreground 1.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	Width	The width-1 of foreground 1.	R

6.6.20 Vertical Synchronize Register (LCDVSYNC)

	LCDVSYNC																															BASE+0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved			VPS													Reserved			VPE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	VPS	V-Sync Pulse start position, fixed to 0. (in line clock)	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	VPE	V-Sync Pulse end position. (in line clock)	RW

6.6.21 Horizontal Synchronize Register (LCDHSYNC)

	LCDHSYNC																BASE+0x0008															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			HPS													Reserved			HPE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	HPS	H-Sync pulse start position. (in dot clock)	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	HPE	H-Sync pulse end position. (in dot clock)	RW

6.6.22 PS Signal Setting (LCDPS)

	LCDPS																BASE+0x0018															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			PSS													Reserved			PSE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	PSS	PS signal start position. (in dot clock) In STN mode, PS signal is ignored. But this register is used to define the AC BIAS signal. AC BIAS signal will toggle very N lines per frame. PSS defines the Toggle position.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:13	Reserved	Writing has no effect, read as zero.	R
12:0	PSE	PS signal end position. (in dot clock) In STN mode, PSE defines N, which described in PSS.	RW

6.6.23 CLS Signal Setting (LCDCLS)

	LCDCLS																															BASE+0x001C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			CLSS													Reserved			CLSE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	CLSS	CLS signal start position. (in dot clock)	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	CLSE	CLS signal end position. (in dot clock)	RW

6.6.24 SPL Signal Setting (LCDSPL)

	LCDSPL																															BASE+0x0020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			SPLS													Reserved			SPLE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	SPLS	SPL signal start position. (in dot clock)	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	SPLE	SPL signal end position. (in dot clock)	RW

* In test mode this register use to keep TV encoder module's output data: comp_luma([25:16]) and chroma([9:0]).

6.6.25 REV Signal Setting (LCDREV)

	LCDREV																															BASE+0x0024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved			REVS													Reserved			REVE												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:16	REVS	REV signal start position. (in dot clock)	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	REVE	REV signal end position. (in dot clock)	RW

Note: LCDPS, LCDCLS, LCDSPL are useless, LCDREV is useless in RTL, but when run case in FPGA, LCDREV.REVE is used to be pixclk devider and devider = REVE + 1;

6.6.26 Interrupt ID Register (LCDIID)

LCDIID is a read-only register that contains a copy of the Frame ID register (LCDFID) from the descriptor currently being processed when a start of frame (SOF) or end of frame (EOF) interrupt is generated. LCDIID is written to only when an unmasked interrupt of the above type is signaled and there are no other unmasked interrupts in the LCD controller pending. As such, the register is considered to be sticky and will be overwritten only when the signaled interrupt is cleared by writing the LCD controller status register.

LCDIID is written with the last channel to reach that state. (i.e. LCDFID of the last channel to reach SOF would be written in LCDIID if SOF interrupts are enabled). Reserved bits must be written with zeros and reads from them must be ignored.

	LCDIID																															BASE+0x0038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IID																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	IID	A copy of Frame ID register, which transferred from Descriptor.	R

6.6.27 Descriptor Address Registers (LCDDAx)

Now the lcdc's descriptor is a 8-word block, aligned on 2-word (8-byte, as axi bus is 64bit wide) boundary, in external memory:

WORD [0] contains the physical address for next LCDDAx.

WORD [1] contains the physical address for LCDSAx.

WORD [2] contains the value for LCDFIDx.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

WORD [3] contains the value for LCDCMDx.

WORD[4] contains the value for LCDOFFSx

WORD[5] contialns the value for LCDPWx

WORD[6] contians the value for LCDCNUMx when LCDCMD.CMD=1 or the value for LCDCPOSx when LCDCMD.CMD=0.

WORD[7] contains the value for LCDDESSIZEx

Software must write the physical address of the first descriptor to LCDDAx before enabling the LCD Controller. Once the LCD Controller is enabled, the first descriptor is read, and all 8 registers are written by the DMAC. The next frame descriptor pointed to by LCDDAx is loaded into the registers for the associated DMA channel after all data for the current descriptor has been transferred.

NOTE: If only one frame buffer is used in external memory, the LCDDAx field (word [0] of the frame descriptor) must point back to itself. That is to say, the value of LCDDAx is the physical address of itself.

Read/write registers LCDDA0 and LCDDA1, corresponding to DMA channels 0 and 1, contain the physical address of the next descriptor in external memory. The DMAC fetches the descriptor at this location after finishing the current descriptor. On reset, the bits in this register are zero. The target address must be aligned to 8-byte boundary. Bits [2:0] of the address must be zero.

	LCDDA0, LCDDA1																															
																	BASE+0x0040, BASE+0x0050														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA0, 1																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	DA0, 1	Next descriptor physical address. And descriptor structure as following: WORD [0]: next descriptor physical address WORD [1]: the buffer physical address WORD [2]: the buffer ID value (Only for debug) WORD [3] contains the value for LCDCMDx. WORD[4] contains the value for LCDOFFSx WORD[5] contialns the value for LCDPWx WORD[6] contians the value for LCDCNUMx when LCDCMD.CMD=1 or the value for LCDCPOSx when LCDCMD.CMD=0. WORD[7] contains the value for LCDDESSIZEx	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.6.28 Source Address Registers (LCDSA)

Registers LCDSA0 and LCDSA1, corresponding to DMA channels 0 and 1, contain the physical address of frame buffer in external memory. The address must be aligned on a 4, 8, or 16 word boundary according to register LCDCTRL.BST in old lcdc. But after rewrite of input DMA, the address just need to be aligned by word, If this descriptor is for frame data, LCDSAx points to the memory location of the frame buffer. This address is incremented by hardware as the DMAC fetches data from memory. If desired, the Frame ID Register can be used to hold the initial frame source address.

	LCDSA0, LCDSA1																															BASE+0x0044, BASE+0x0054
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA0, 1																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	SA0, 1	Buffer start address. (Only for driver debug)	R

6.6.29 Frame ID Registers (LCDFIDx)

Registers LCDFID0 and LCDFID1, corresponding to DMA channels 0 and 1, contain an ID field that describes the current frame. The particular use of this field is up to the software. This ID register is copied to the LCD Controller Interrupt ID Register when an interrupt occurs.

	LCDFID0, LCDFID1																															BASE+0x0048, BASE+0x0058
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FID0, 1																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	FID0, 1	Frame ID. (Only for debug)	R

6.6.30 DMA Command Registers (LCDCMDx)

	LCDCMD0, LCDCMD1																															BASE+0x004C, BASE+0x005C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SOFINT	EOFINT	CMD	PAL	DECOMPEN	FRM_EN	FIELD_SEL	16x16BLOCK	LEN																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	SOFINT	Enable start of frame interrupt. When SOFINT =1, the DMAC sets the start of frame bit (LCDSTATE.SOF) when starting a new frame. The SOF bit is set after a new descriptor is loaded from memory and before the frame data is fetched. In dual-panel mode, LCDSTATE.SOF is set only when both channels reach the start of frame and both frame descriptors have SOFINT set.	R
30	EOFINT	Enable end of frame interrupt. When EOFINT =1, the DMAC sets the end of frame bit (LCDSTATE.EOF) after fetching the last word in the frame buffer. In dual-panel mode, LCDSTATE.EOF is set only when both channels reach the end of frame and both frame descriptors have EOFINT set.	R
29	CMD	It is used to distinguish command and data in lcm mode. And it is only loaded via DMA channel 0. 1: The data is command 0: The data is data	R
28	PAL	As palette is not supported, keep this bit to 0.	
27	DECOMPEN	as display_compress is not used, Keep this bit 0	
26	FRM_EN	Indicat this frm is enable. 0:disable; 1:enable.	R
25	FIELD_SEL	Field select for interlace : 0:odd field or no interlace; 1: even field As tve is not used, keep this bit to 0	
24	16x16BLOCK	Fetch data by 16x16 block,0:disable;1:enable.	R
23:0	LEN	The buffer length value. (in WORD) The LEN bit field determines the number of bytes of the buffer size pointed by LCDSAx. LEN = 0 is not valid. DMAC fetch data according to LEN. Each time one or more word(s) been fetched, LEN is decreased automatically. Software can read LEN. *When you use 16x16block mode, the LEN should be the line number, not word number.	R

6.6.31 DMA OFFSIZE Registers (LCDOFFSx)

	LCDOFFS0, LCDOFFS1																BASE+0x0060, BASE+0x0070															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								OFFSIZE																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
23:0	OFFSIZE0, 1	OFFSIZE value for DMA 0,1. Indicate the offset in word. *please notice that when you need OFFSIZE function, to set this reg to an un-zero value and also need to set LCDPW0, 1 to indicate how much word in one line of this frame. *When you use 16x16block mode, you must use this to indicate how many word between each of block	R

6.6.32 DMA Page Width Registers (LCDPWx)

	LCDPW0, LCDPW1																BASE+0x0064, BASE+0x0074															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								PAGEWIDTH																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
23:0	PAGEWIDTH0, 1	Page width for DMA 0,1. * When you set LCDOFFS.OFFSIZE0/1 to 0, you need keep the PAGEWIDTH0/1 0. *When you use 16x16block mode, you use this to indicate how many words per line.	R

Note: When not in 16x16 block mode and offsize is set to 0, PAGEWIDTH must be set 0, if not in 16x16 mode and offsize is not 0, PAGEWIDTH shoule be words in block line that dma need to read in frame buffer. for example,surppose there is a 90 line picture in frame buffer, if you set PAGEWIDTH

as large as 30lines, the picture will be divided to 3 blocks of 30lines. but usually we will set PAGEWIDTH as large as 1line, and divide the picture to 90 blocks, that is to say one block equals one line.

As h2d is not used now, we will not use 16x16block mode.

6.6.33 DMA Commend Counter Registers (LCDCNUMx)

When LCDCMD.CMD = 1, **BASE+0x0068, BASE+0x0078** is use as LCDCNUM0, 1 are not used now, set it to 0.

	LCDCNUM0, LCDCNUM1																BASE+0x0068, BASE+0x0078															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								CNUM							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
7:0	CNUM0,1	Commands' number in this frame transfer by DMA. (only use in Smart LCD mode)	R

6.6.34 DMA Commend Counter Registers (LCDCPOSx)

When LCDCMD.CMD = 0, **BASE+0x0068, BASE+0x0078** is use as LCDPOS0, 1.

	LCDPOS0, LCDPOS1																**BASE+0x0068, BASE+0x0078**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPHAMD1	RGB0/1	BPP0/1			PREMULTI0/1	COEF_SLE0/1		YPOS								XPOS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	ALPHAMD1	Alpha blending mode for foreground0/1. 0: One transparency for the whole graphic, and the LCDALPHA register is used for transparency 1: One transparency for each pixel in one graphic, and the alpha value is coming from each pixel data	R
30	RGB0/1	Bpp16 RGB mode of foreground0/1. 0: RGB565; 1: RGB555.	R
29:27	BPP0/1	Bits Per Pixel of OSD channel 1. **Bits Per Pixel** 000 Reserved 001 Reserved 010 Reserved 011 Reserved 100 15/16bpp 101 18bpp/24bpp 110 24bpp compressed 111 30bpp	R
26	PREMULTI0/1	Premulti enable of foreground0/1 : 0 : data has been premultied and don't need premulti 1 : data should be premultied by lcd	R
25:24	COEF_SLE0/1	Select coefficient for foreground0/1 : 00 : 0 01 : 1 10 : alpha1 11 : 1-alpha1	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

23:12	YPOS0,1	The Y position of top-left part for foreground 0/1	R
11:0	XPOS0,1	The X position of top-left part for foreground 0/1	R

Note: only low 8 bits of LCDCPOS is readable when LCDCMD.CMD = 0.

6.6.35 Foreground x Size in Descriptor (LCDDESSIZEx)

When LCDCMD.CMD = 0, **BASE+0x006C, BASE+0x007C** is use as LCDDESSIZE0, 1, to indicator the next frame foreground0, 1's size.

	LCDDESSIZE0, 1																															**BASE+0x006C, BASE+0x007C**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPHA								Height								Width															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	ALPHA	The alpha value of foreground0/1 for one graphic with one transparency.	R
23:12	Height	The height-1 of foreground 0/1.	R
11:0	Width	The width-1 of foreground 0/1.	R

6.6.36 Priority level threshold configure Register (LCDPCFG)

	LCDPCFG																															**BASE+0x02C0**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Lcd_pri_md	HP_BST			Reserved	Pcfg2									Pcfg1									Pcfg0								
RST	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

The 3 thresholds cut the output fifo into 4 space,. When the entries remained triggers one of them, the priority level will be altered by hardware. And the 3 thresholds must follow the order: pcfg2 ≥ pcfg1 ≥ pcfg0.

Priority level

Priority_lv	Space
11	Empty ~ threshold0
10	Threshold0 ~ threshold1
01	Threshold1 ~ threshold2
00	Threshold2 ~ full

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	Lcd_pri_md	Lcd priority mode. 0: use lcd dynamitic priority level; 1: use arbiter priority level. this bit is useless now	RW
30:28	HP_BST	Highest priority Burst Length Selection. 000: 4 word 001: 8 word 010: 16 word 011: 32 word 101: Contiue16 100: 64 word 111: Disable	RW
27	Reserved	Writing has no effect, read as zero.	R
26:18	Pcfg2	Threshold2: 0~511.	RW
17:9	Pcfg1	Threshold1: 0~511.	RW
8:0	Pcfg0	Threshold0: 0~511	RW

6.7 LCD Controller Pin Mapping

There are several mapping schemes for different LCD panels.

6.7.1 TFT and CCIR Pin Mapping

Pin	Generic 8-bit Serial TFT	Generic 18/16-bit Parallel TFT	Special TFT 1 18/16-bit Parallel	Special TFT 2 18/16-bit Parallel	Special TFT 3 18/16-bit Parallel	CCIR656 8-bit	CCIR601 16-bit
Lcd_pclk/ Slcd_clk	CLK	CLK	DCLK	CLK	HCLK	CLK	CLK
Lcd_vsync/ Slcd_wr	VSYNC	VSYNC	SPS	GSRT	STV	VSYNC	VSYNC
Lcd_hsync/ Slcd_dc	HSYNC	HSYNC	LP	GPCK	STH	HSYNC	HSYNC
Lcd_de	DE	DE	-	-	-	-	-
Lcd_ps	-	-	Pulse	Toggle	Toggle	-	-
Lcd_cls	-	-	Pulse	Pulse	Pulse	-	-
Lcd_rev	-	-	Toggle	Toggle	Toggle	-	-
Lcd_spl	-	-	Pulse	Pulse	Toggle	-	-
Lcd_dat17	-	R5 -	R5 -	R5 -	R5 -	-	-

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Lcd_dat16	-	R4	-	R4	-	R4	-	R4	-	-	-
Lcd_dat15	-	R3	R5	R3	R5	R3	R5	R3	R5	-	D15
Lcd_dat14	-	R2	R4	R2	R4	R2	R4	R2	R4	-	D14
Lcd_dat13	-	R1	R3	R1	R3	R1	R3	R1	R3	-	D13
Lcd_dat12	-	R0	R2	R0	R2	R0	R2	R0	R2	-	D12
Lcd_dat11	-	G5	R1	G5	R1	G5	R1	G5	R1	-	D11
Lcd_dat10	-	G4	G5	G4	G5	G4	G5	G4	G5	-	D10
Lcd_dat9	-	G3	G4	G3	G4	G3	G4	G3	G4	-	D9
Lcd_dat8	-	G2	G3	G2	G3	G2	G3	G2	G3	-	D8
Lcd_dat7	R7/G7/B7	G1	G2	G1	G2	G1	G2	G1	G2	D7	D7
Lcd_dat6	R6/G6/B6	G0	G1	G0	G1	G0	G1	G0	G1	D6	D6
Lcd_dat5	R5/G5/B5	B5	G0	B5	G0	B5	G0	B5	G0	D5	D5
Lcd_dat4	R4/G4/B4	B4	B5	B4	B5	B4	B5	B4	B5	D4	D4
Lcd_dat3	R3/G3/B3	B3	B4	B3	B4	B3	B4	B3	B4	D3	D3
Lcd_dat2	R2/G2/B2	B2	B3	B2	B3	B2	B3	B2	B3	D2	D2
Lcd_dat1	R1/G1/B1	B1	B2	B1	B2	B1	B2	B1	B2	D1	D1
Lcd_dat0	R0/G0/B0	B0	B1	B0	B1	B0	B1	B0	B1	D0	D0

Note : the current version of LCDC does not support Special TFT and CCIR

TFT 24 bit parallel mode/16 bit parallel mode2:

Pin	16 bit Parallel mode2	24 bit Parallel
Lcd_pclk/ Slcd_clk	CLK	CLK
Lcd_vsync/Slcd_wr	VSYNC	VSYNC
Lcd_hsync/Slcd_dc	HSYNC	HSYNC
Lcd_de	DE	DE
Lcd_ps	-	-
Lcd_cls	-	-
Lcd_rev	-	-
Lcd_spl	-	-
Lcd_dat17	R7	R7
Lcd_dat16	R6	R6
Lcd_dat15	R5	R5
Lcd_dat14	R4	R4
Lcd_dat13	R3	R3
Lcd_dat12	G7	R2
Lcd_dat11	G6	G7
Lcd_dat10	G5	G6

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Lcd_dat9	0 (NC for panel)	G5
Lcd_dat8	G4	G4
Lcd_dat7	G3	G3
Lcd_dat6	G2	G2
Lcd_dat5	B7	B7
Lcd_dat4	B6	B6
Lcd_dat3	B5	B5
Lcd_dat2	B4	B4
Lcd_dat1	B3	B3
Lcd_dat0	0 (NC for panel)	B2
Lcd_lo6_o[5]	0	R1
Lcd_lo6_o[4]	0	R0
Lcd_lo6_o[3]	0	G1
Lcd_lo6_o[2]	0	G0
Lcd_lo6_o[1]	0	B1
Lcd_lo6_o[0]	0	B0

6.7.2 Data mapping to GPIO function.

pin name in LCD	mapping to GPIO function
Lcd_dat17/Slcd_dat17	lcd_r7
Lcd_dat16/Slcd_dat16	lcd_r6
Lcd_dat15/Slcd_dat15	lcd_r5
Lcd_dat14/Slcd_dat14	lcd_r4
Lcd_dat13/Slcd_dat13	lcd_r3
Lcd_dat12/Slcd_dat12	lcd_r2
Lcd_dat11/Slcd_dat11	lcd_g7
Lcd_dat10/Slcd_dat10	lcd_g6
Lcd_dat9/Slcd_dat9	lcd_g5
Lcd_dat8/Slcd_dat8	lcd_g4
Lcd_dat7/Slcd_dat7	lcd_g3
Lcd_dat6/Slcd_dat6	lcd_g2
Lcd_dat5/Slcd_dat5	lcd_b7
Lcd_dat4/Slcd_dat4	lcd_b6
Lcd_dat3/Slcd_dat3	lcd_b5
Lcd_dat2/Slcd_dat2	lcd_b4
Lcd_dat1/Slcd_dat1	lcd_b3
Lcd_dat0/Slcd_dat0	lcd_b2
Lcd_lo6_o[5] /Slcd_dat23	lcd_r1
Lcd_lo6_o[4] /Slcd_dat22	lcd_r0
Lcd_lo6_o[3] /Slcd_dat21	lcd_g1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Lcd_lo6_o[2] /Slcd_dat20	lcd_g0
Lcd_lo6_o[1] /Slcd_dat19	lcd_b1
Lcd_lo6_o[0] /Slcd_dat18	lcd_b0

6.8 Display Timing

6.8.1 General 16-bit and 18-bit TFT Timing

This section shows the general 16-bit and 18-bit TFT LCD timing diagram, the polarity of signal "Vsync", "Hsync", and "PCLK" can be programmed correspond to the LCD panel specification.

Figure 6-6 General 16-bit and 18-bit TFT LCD Timing

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.8.2 8-bit Serial TFT Timing

This section shows the 8-bit serial TFT LCD timing diagram, the polarity of signal “Vsync”, “Hsync”, and “PCLK” can be programmed correspond to the LCD panel specification.

Figure 6-7 8-bit serial TFT LCD Timing (24bpp)

6.8.3 Delta RGB panel timing

This section shows the Delta RGB timing diagram, the polarity of signal “Vsync”, “Hsync”, and “PCLK” can be programmed. And the odd/even line RGB order also can be programmed correspond to the LCD panel specification.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 6-8 Delta RGB timing

6.8.4 RGB Dummy mode timing

This section shows the RGB Dummy diagram, the polarity of signal “Vsync”, “Hsync”, and “PCLK” can be programmed.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

***Dummy = 0**

Figure 6-9 RGB Dummy timing

6.9 Format of Frame Buffer

6.9.1 16bpp

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.9.2 18bpp

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	R5	R4	R3	R2	R1	R0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G5	G4	G3	G2	G1	G0	0	0	B5	B4	B3	B2	B1	B0	0	0

6.9.3 24bpp

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	R7	R6	R5	R4	R3	R2	R1	R0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0

6.9.4 30bpp

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0	G9	G8	G7	G6
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G5	G4	G3	G2	G1	G0	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0

6.9.5 16bpp with alpha

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
A7	A6	A5	A4	A3	A2	A1	A0	0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R5	R4	R3	R2	R1	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1

6.9.6 18bpp with alpha

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
A7	A6	A5	A4	A3	A2	A1	A0	R5	R4	R3	R2	R1	R0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G5	G4	G3	G2	G1	G0	0	0	B5	B4	B3	B2	B1	B0	0	0

6.9.7 24bpp with alpha

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
A7	A6	A5	A4	A3	A2	A1	A0	R7	R6	R5	R4	R3	R2	R1	R0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0

6.9.8 24bpp compressed

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

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RED 3 [7:0]								GLEEN 3 [7:0]							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BLUE3 [7:0]								RED2 [7:0]							

6.10 Format of Data Pin Utilization

6.10.1 18-bit Parallel TFT

Col0 (RGB)																	
D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

6.10.2 16-bit Parallel TFT

Col0 (RGB)															
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

6.10.3 8-bit Serial TFT (24bpp)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Col0 (R)							
D7	D6	D5	D4	D3	D2	D1	D0
Col0 (G)							
D7	D6	D5	D4	D3	D2	D1	D0
Col0 (B)							
D7	D6	D5	D4	D3	D2	D1	D0

6.11 LCD Controller Operation

6.11.1 Set LCD Controller AHB Clock and Pixel Clock

The LCD Controller has 2 clock input: AHB clock and pixel clock. The both clocks are generated by CPM (Clock and Power Manager). The frequency of the 2 clocks can be set by CPM registers. lcdc ‘s AHB clock is equal to AHB0 clock (HCLK in CPM spec), and CPM.LPCDR set LCD pixel clock division ratio. Please refer to CPM spec for detail.

LCD AHB clock is the LCD controller’s internal clock while LCD pixel clock is output to drive LCD panel. There have 2 rules for LCD clocks in order to make sure output fifo not underrun:

1. For TFT Panel, the frequency of LCD AHB clock must be at least 1.5 times of LCD pixel clock.
2. For STN Panel, the frequency of LCD AHB clock must be at least 3 times of LCD pixel clock.

That is to make sure LCD will not underrun.
LCD panel determines the frequency of LCD pixel clock.

6.11.2 Enabling the Controller

If the LCD controller is being enabled for the first time after system reset or sleep reset, all of the LCD registers must be programmed as follows:

1. Write the frame descriptors to memory.
2. Program the entire LCD configuration registers except the Frame Descriptor Address Registers (LCDDAx) and the LCD Controller enable bit (LCDCTRL.ENA).
3. Program LCDDAx with the memory address of the frame descriptor.
4. Enable the LCD controller by writing to LCDCTRL.ENA.

If the LCD controller is being re-enabled, there has not been a reset since the last programming; only the registers LCDDAx and LCDCTRL.ENA need to be reprogrammed. The LCD Controller Status Register (LCDSTATE) must also be written to clear any old status flags before re-enabling the LCD controller.

Once the LCD controller has been enabled, do not write new values to LCD registers except LCDCTRL.ENA or DIS.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.11.3 Disabling the Controller

The LCD controller can be disabled in two ways: regular and quick.

1 Regular disabling.

Regular disabling is accomplished by setting the disable bit, LCDCTRL.DIS. The other bits in LCDCTRL must not be changed — read the register, set the DIS bit, and rewrite the register. This method causes the LCD controller to stop cleanly at the end of a frame. The LCD Disable Done bit, LCDSTATE.LDD, is set when the LCD controller finishes displaying the last frame, and the enable bit, LCDCTRL.ENA, is cleared automatically by hardware. LCDCTRL.DIS must be set zero when enabling the controller.

2 Quick disabling.

Quick disabling is accomplished by clearing the enable bit, LCDCTRL.ENA. The LCD controller will finish any current DMA transfer, stop driving the panel, setting the LCD Quick Disable bit (LCDSTATE.QD) and shut down immediately. This method is intended for situations such as a battery fault, where system bus traffic has to be minimized immediately so the processor can have enough time to store critical data to memory before the loss of power. The LCD controller must not be re-enabled until the QD bit is set, indicating that the quick shutdown is complete. Do not set the DIS bit when a quick disabling command has been issued.

NOTE: It is strongly recommended that software set the “LCD Module Stop Bit” in CPM to shut down LCDC clock supply to save power consumption after disable LCDC. Please refer to CPM for detailed information.

6.11.4 Resetting the Controller

At reset, the LCD Controller is disabled. All LCD Controller Registers are reset to the conditions shown in the register descriptions.

6.11.5 Frame Buffer

The starting address of frame buffer stored in external memory must be aligned to 4, 8 or 16 words boundary according to register LCDCTRL.BST. The length of buffer must be multiple of word (32-bit).

If LCDCTRL .BST = 0, align frame buffer to 16 word boundary
If LCDCTRL .BST = 1, align frame buffer to 8 word boundary
If LCDCTRL .BST = 2, align frame buffer to 4 word boundary
but as DMA is rewrited, Frame buffer just need to be aligned to word.

One frame buffer contains encoded pixel data of multiple of screen lines; each line of encoded pixel data must be aligned to word boundary. If the length of a line is not the multiple of word, extra bits must be applied to reach a word boundary. It is suggested that the extra bits to be set zero.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.11.6 OSD Operation

1 Normal process.

a Configuration.

* LCDRGBC

b Set Color.

* LCDBGC, LCDKEY0, LCDKEY1, LCDALHPA

c Set Display.

* LCDVAT, LCDDAH, LCDDAV
* LCDVSYNC, LCDHSYNC

d Set Descriptors.

* LCDIID
* LCDDA0, LCDSA0, LCDFID0, LCDCMD0, LCDOFFS0, LCDPW0, LCDCNUM0/LCDPOS0, LCDDESSIZE0
* LCDDA1, LCDSA1, LCDFID1, LCDCMD1, LCDOFFS1, LCDPW1, LCDCNUM1/LCDPOS1, LCDDESSIZE1

e Enable LCDC.

f Check the state from register LCDSTATE and LCDOSDS.

2 Reconfigure OSD.

If foreground0 and foreground1 (enable, position, size, bpp, alphamd, etc.) need to reconfigure during display process, just prepare a new descriptor and lcd will reconfigure automaticlly.

6.11.7 Descriptor Operation

1 TFT panel

you can use only one descriptor or several connected descriptor. As which shown below.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 SLCD

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7 Smart LCD Controller

7.1 Overview

The Smart LCD Controller affords an interface to transfer data from the LCD controller to the SLCD interface. It supports DMA operation and register operation.

Features:

- Supports a large variety of LCD Module from different vendors
- Supports parallel and serial interfaces
- Supports different size of display panel
- Supports different width of pixel data
- Supports internal DMA operation and register operation
- Supports Write Operation. Read Operation is not supported

7.2 Structure

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.3 Pin Description

Table 7-1 SLCD Pins Description

Name	I/O	Description	Interface
SLCD_DC	Output	Command/Data Select Signal. The polarity of the signal can be programmable.	Serial: DC Parallel: DC
SLCD_WR	Output	Data Sample Signal. The polarity of the signal can be programmable.	Serial: WR Parallel: Sample Data with the edge of WR
SLCD_DAT*1 [17:0]	Output	The data of SLCD. Relate to 1.9.4 Data mapping to GPIO function.	Serial: SLCD_DAT [15] Parallel: 18bit SLCD_DAT [17:0] 16bit SLCD_DAT [15:0] 8bit SLCD_DAT [7:0]
SLCD_DAT[23:18]	Output	24 bit parallel SLCD RGB (or 24 bit command) low bit ([17:16],[9:8],[1:0]) output Relate to 1.7.2 Data mapping to GPIO function.	SLCD_DAT[23:22]->R[1:0] SLCD_DAT[21:20]->G[1:0] SLCD_DAT[19:18]->B[1:0]

NOTE:

***1:** SLCD_DAT [15] is also use as data pin for serial. The SLCD pins are shared with LCDC. You can see the set of register LCDCFG.LCDPIN in LCDC spec.

7.4 Register Description

In this section, we will describe the registers in Smart LCD controller. Following table lists all the registers definition. All register's 32bit address is physical address. And detailed function of each register will be described below.

Name	Description	RW	Reset Value	Address	Access Size
MCFG	Old SLCD Configure Register	RW	0x00000000	0x00A0	32
MCFG_NEW	new SLCD Configure Register	RW	0x00000000	0x00B8	32
MCTRL	SLCD Control Register	RW	0x00000138	0x00A4	32
MSTATE	SLCD Status Register	R	0x00010000	0x00A8	32
MDATA	SLCD Data Register	RW	0x00000000	0x00AC	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

WTIME	SLCD Wait Time Register	RW	0x00000000	0x00B0	32
TASH	Address Setup and Hold Time Register	RW	0x00000000	0x00B4	32
SMWT	Slow Mode Wait Time Register	RW	0x00000000	0x00BC	32

.

7.4.1 SLCD Configure Register (MCFG)

The register MCFG is used to configure SLCD.

	MCFG																															BASE+0x00A0
Bit																16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			DWIDTH			CWIDTH		Reserved			WRPLY	DCPLY	Reserved	CLKPLY	TYPE
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:13	Reserved	Writing has no effect, read as zero.	R
12:10	DWIDTH*1	Data Width. **DWIDTH** / **Data Width** 000 / 18-bit once Parallel/Serial 001 / 16-bit once Parallel/Serial 010 / 8-bit third time Parallel 011 / 8-bit twice Parallel 100 / 8-bit once Parallel/Serial 101 / 24-bit once Parallel 111 / 9-bit twice Parallel 110 / Reserved *Please notice that you can only use 24-bit parallel command when use 24-bit parallel data. (The command may be not 24-bit but need put them as 24-bit in memory(one command use one word))	RW
9:8	CWIDTH*1	Command Width. **CWIDTH** / **Command Width** 00 / 16-bit once / 9bit once 01 / 8-bit once 10 / 18-bit once 11 / 24-bit once	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		*Please notice that you can only use 24-bit parallel command when use 24-bit parallel data. (The command may be not 24-bit but need put them as 24-bit in memory (one command use one word))	
7:5	Reserved	Writing has no effect, read as zero.	R
4	WRPLY	WR Polarity. (WR initial level will be different from WR Polarity) 0: Active Level is Low 1: Active Level is High	RW
3	DCPLY	DC Polarity. 0: Command DC = 0, Data DC = 1 1: Command DC = 1, Data DC = 0	RW
2	Reserved	Writing has no effect, read as zero.	R
1	CLKPLY	LCD_CLK Polarity. (for cpm） 0: Active edge is Falling 1: Active edge is Rising	RW
0	TTYPE	Transfer Type. 0: Parallel 1: Serial	RW

NOTE:

1: The set of DWIDTH and CWIDTH should keep to the rules as follows:

Interface Mode	Command Width	Data Width	Color
Parallel	18-bit	18-bit once	R6G6B6
	16-bit	16-bit once	R5G6B5
		9-bit twice	--
	9-bit	9-bit twice	--
	8-bit	8-bit once	--
		8-bit twice	
		8-bit third times	--
Serial	18-bit	18-bit once	--
	16-bit	16-bit once	--
	9-bit	9bit twice	--
	8-bit	8-bit once	--
		8-bit twice	--
		8-bit third times	--

2: When use new slcd, keep MCFG.DWIDTH to be 3’b101.MCFG.CWIDTH together with MCFG_NEW.CMD_9BIT decide cmd width. other bits of MCFG are useless.

7.4.2 SLCD Configure Register (MCFG_NEW)

The register MCFG_NEW is used to configure new SLCD.it works together with MCFG.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MCFG_NEW																																BASE+0x00B8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																DWIDTH_NEW			Reserved	6800_md	Cmd_9bit	DTIMES_NEW		Reserved		CSPLY_NEW	RSPLY_NEW	CLKPLY_NEW	DTYPE_NEW	CTYPE_NEW	FMT_CONV
RST														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:13	DWIDTH–NEW	Data Width.it's also useful in MIPI DSI command mode DWIDTH: 3'b000 – 8 bit; 3'b001 – 9bit; 3'b010 – 16bit; 3'b011 – 18bit; 3'b100 – 24bit; Others – Reserved	RW
12	Reserved		RW
11	6800_md	When high, wr will be inversed and that's 6800's timing Keep low when not in 6800 mode	RW
10	cmd_9bit	Useful when cwidth in old slcd is 2'b00 1'b1:9 bit mode 1'b0:16 bit mode	RW
9:8	DTIMES_NEW	Data times when in 8BPP parallel mode 2'b00 : once 2'b01 : twice 2'b10 : 3 times 2'b11 : reserved	RW
7:6	Reserved		
5	CSPLY_NEW	1'b0:WR will be high when in IDLE(8080 or serial) 1'b1:WR will be low when in IDLE(6800s)	RW
4	RSPLY_NEW	DC Polarity. 0: Command DC = 0, Data DC = 1 1: Command DC = 1, Data DC = 0	RW
3	CLKPLY_NE	LCD_CLK Polarity.	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	W	0: Active edge is Falling 1: Active edge is Rising	
2	DTYPE_NEW	Data Transfer Type. 0: Parallel 1: Serial	RW
1	CTYPE_NEW	Cmd Transfer Type. 0: Parallel 1: Serial	RW
0	FMT_CONV	Format conversion or not 1'b1:use format conversion 1'b0:not use format conversion	RW

NOTE1:

When WIDTH is 8bit and TYPE is serial, TIMES should be zero.

NOTE2:

MCFG and MCFG_NEW is for old slcd and new slcd seperately.when use new slcd, DWIDTH in MCFG should be set as 24bpp, and DWIDTH_NEW in MCFG_NEW decide the output

NOTE3:

when send data and FMT_CONV is low,8/16/18bpp will directly give out the low 8/16/18 bits from output fifo directly; if this bit is set high, new slcd will convert the 24bpp data in output fifo to relative bpps as follows

24bpp data in output fifo:

31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
X	X	X	X	X	X	X	X	R7	R6	R5	R4	R3	R2	R1	R0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0

8bpp output data(R3G3B20:

								23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
								0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	0	0	0	0	0	0	0	R7	R6	R5	G7	G6	G5	B7	B6

15bpp output data(R5G5B5)

								23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
								0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	R7	R6	R5	R4	R3	G7	G6	G5	G4	G3	B7	B6	B5	B4	B3

16bpp output data(R5G6B5)

								23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
								0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R7	R6	R5	R4	R3	G7	G6	G5	G4	G3	G2	B7	B6	B5	B4	B3

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18bpp output data(R6G6B6)

								23	22	21	20	19	18	17	16
								0	0	0	0	0	0	R7	R6
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R5	R4	R3	R2	G7	G6	G5	G4	G3	G2	B7	B6	B5	B4	B3	B2

9bit twice:

it's data format is the same with 18bpp, but the data will be given out in twice,the first time is

								23	22	21	20	19	18	17	16
								0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	0	0	0	0	0	0	R7	R6	R5	R4	R3	R2	G7	G6	G5

the second time is

								23	22	21	20	19	18	17	16
								0	0	0	0	0	0	0	0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	0	0	0	0	0	0	G4	G3	G2	B7	B6	B5	B4	B3	B2

Note4:

when send cmd, MCFG_NEW.FMT_CONV is useless, the output cmd is only decided by cmd width. if cmd is 8bits width,only the low 8 bits of slcd data is useful, and one word in buffer will be divided into 4 cmd, the 1st cmd will be data[7:0], second data[15:8], third data[23:16], and the last is data[31:24]; if cmd is 9bits width, only the low 9bit of slcd data is useful, and one word in buffer will be divide into 2 cmd, the 1st cmd will be data[8:0], and the second will be data[24:16], other bits will be lost.; if cmd is 16bits width, only the low 16bit of slcd data is useful, and one word in buffer will be divide into 2 cmd, the 1st cmd will be data[15:0], and the second will be data[31:16]; if cmd width is 18bits or 24bits width, one word in buffer will only send one cmd, only low 18 bits or the total 24bits of slcd data is useful, and the cmd will be data[17:0] or data[23:0].

7.4.3 SLCD Control Register (MCTRL)

MCTRL is SLCD Control Register.

	MCTRL																															**BASE+0x00A4**
Bit																							11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																						SD_SLCD	NARROW_TE	TE_INV	NOT_USE_TE	DCSI_SEL	MIPI_SLCD	NEW_SLCD	FAST_MODE	gate_mask	DMAMODE	DMASTART	Dmatxen
RST																							0	0	0	1	0	0	1	1	1	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
11	SD_SLCD	This bit is used to gate slcd's(include old and new slcd) pixclk when use MIPI lcd. 1'b1:shutdown slcd's clk 1'b0:not shutdown slcd's clk	RW
10	NARROW_TE	1'b1: support narrow TE 1'b0:only support TE as wide as 3 pixclk	RW
9	TE_INV	1'b0: te is active high 1'b1:te is active low	RW
8	NOT_USE_TE	1'b1:not wait for TE's relative edge 1'b0: wait for TE's relative edge	RW
7	DCSI_SEL	DSI or CSI sel, it's only used in fpga for debug	RW
6	MIPI_SLCD	1'b1:use mipi slcd (for mipi dsi command mode) 1'b0:not use mipi slcd	RW
5	NEW_SLCD	1'b1:use new slcd 1'b0:use old slcd This bit is valid only when MIPI_SLCD is low	RW
4	FAST_MODE	1'b1:MCU's frame frequency is faster than smart lcd panel 1'b0: MCU's frame frequency is lower than smart lcd panel	RW
3	GATE_MASK	1'b0:slcd's clk will never be gated 1'b1: slcd's clk (include hclk and pixclk) will be gated when in dma singel mode and not busy.	RW
2	DMAMODE	SLCD descriptor DMA mode select. 0: DMA will continually transfer data follow descriptor chain 1: DMA will stop when one descriptor finished	RW
1	DMASTART	Only use when DMAMODE = 1, set 1 to restart DMA transfer.	RW
0	DMATXEN	DMA mode enable.	RW

7.4.4 SLCD Status Register (MSTATE)

The register of MSTATE is SLCD status register.

	MSTATE																															BASE+0x00A8
Bit																													7	6	5	4	3	2	1	0
	LCD_ID																												Reserved							BUSY
RST																													0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	**LCD_ID**	**It's used to point the version of LCDC,in current version, it's 1**	
7:1	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	BUSY	Transfer is working or not. This bit will be set to 1 when transfer is working. It will be cleared by hardware when transfer is finished. 0: not busy 1: busy	R

7.4.5 SLCD Data Register (MDATA)

The register MDATA is used to send command or data to SLCD. When PTR=2'b00, the low 24-bit is used as data. When PTR=2'b01, the low 24-bit is used as command. When PTR=2'b10, the low 24-bit is used as wait time.

	MDATA																															BASE+0x00AC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PTR		Reserved						DATA / CMD																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	PTR	The PTR bit of data register is used to decide the meanings of the low 24-bit. 2'b00: data 2'b01: command 2'b10: wait time 2'b11: reserved	RW
29:24	Reserved	Writing has no effect, read as zero.	R
23:0	DATA/CMD	Data or Command Register or wait time	RW

Note:when use old slcd, MDATA[31] is useless, and MDATA[30] high means command, low means data;when use new slcd,MDATA[31:30] is as described in the table above.

7.4.6 SLCD Wait Time Register (WTIME)

	WTIME																															BASE+0x00B0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DHTIME								DLTIME								CHTIME								CLTIME							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DHTIME:high time period when writing data
DLTIME:low time period when writing data
CHTIME:high time period when writing cmd

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

CLTIME:low time period when writing cmds

7.4.7 Address Setup and Hold Time Register (TASH)

	WTIME																															
																																BASE+0x00B4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MIPI_WAIT																TAH								TAS							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TAS：address setup time

TAH：address hold time

MIPI_WAIT:DBI wait time between two frames.

7.4.8 Slow Mode Wait Time Register(SMWT)

	SLOW_TIME																															
																																BASE+0x00BC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																SLOW_TIME															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SLOW_TIME: if use the function of slcd to wait for TE's relative edge, slcd will wait for SLOW_TIME's period of pixclk before it begin to flush out data.

7.5 System Memory Format

7.5.1 Data format

You can configure these registers according to LCDC module.

7.5.2 Command Format

1 18-bit command

31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
X	X	X	X	X	X	X	X	X	X	X	X	X	X	C17	C16
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0

2 16-bit command

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0

3 9-bit command once

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
X	X	X	X	X	X	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
X	X	X	X	X	X	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0

4 8-bit command once

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
C7	C6	C5	C4	C3	C2	C1	C0	C7	C6	C5	C4	C3	C2	C1	C0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
C7	C6	C5	C4	C3	C2	C1	C0	C7	C6	C5	C4	C3	C2	C1	C0

5 8-bit command twice (Command = command part + data part)
*Please notice that when you use this kind command, set CWIDTH as 8bit once and set the LCDCNUM.CNUM as doubled the real command number.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
D7	D6	D5	D4	D3	D2	D1	D0	C7	C6	C5	C4	C3	C2	C1	C0
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
D7	D6	D5	D4	D3	D2	D1	D0	C7	C6	C5	C4	C3	C2	C1	C0

7.6 Transfer Mode

Two transfer modes can be used: DMA Transfer Mode and Data Register Transfer Mode. In DMA mode, always transfer commands by DMA 0.

7.6.1 DMA Transfer Mode

Command and data can be recognized by DC bit coming from memory. The format of DMA transfer can be as follows:

1 Command and Data

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

C	D	D	C	C	D	D	D	C	C	C	D	D	D	D

*Please notice that the command only can insert between two complete frame and the number of command is 0~255.

2 Only Data

D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

*You can also not use command but you still need to use a command descriptor and set the CNUM = 0.

Because DMA transfer mode only can work in OSD mode, you need to configure the panel according OSD mode:

1 Set Color.
* LCDBGC0,1, LCDKEY0, LCDKEY1, LCDALHPA

2 Set Display.
* LCDVAT, LCDDAH, LCDDAV
* LCDVSYNC, LCDHSYNC

3 Set Descriptors.
* LCDIID
* LCDDA0, LCDSA0, LCDFID0, LCDCMD0, LCDOFFS0, LCDPW0, LCDCNUM0/LCDPOS0, LCDDESSIZE0
* LCDDA1, LCDSA1, LCDFID1, LCDCMD1, LCDOFFS1, LCDPW1, LCDCNUM1/LCDPOS1, LCDDESSIZE1

4 Enable slcd DMA.

5 Enable LCDC.

7.6.2 Register Transfer Mode

Each time you can write a command or a data to the register, then it will transfer the DC signal and data or command to SLCD. Command and data can be recognized by DC bit coming from data register. The format of data register transfer can be as follows:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.7 Timing

7.7.1 Parallel Timing

7.7.2 Serial Timing

7.8.1 DMA Operation

1 Start DMA transfer.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

a Set LCDCFG.MODE to 1101 to choose LCM.

b Set LCDCTRL.BST to choose burst length for transferring.

c Set register LCDIID0, LCDDA0, LCDSA0, LCDFID0, LCDCMD0, LCDOFFS0, LCDPW0, LCDCNUM0, LCDDESSIZE0 to initial internal DMA.

d Also set register LCDIID1, LCDDA1, LCDSA1, LCDFID1, LCDCMD1, LCDOFFS1, LCDPW1, LCDCNUM1, LCDDESSIZE1 when use DMA channel 1 in OSD mode.

e Set MCFG to configure SLCDC.and if use new slcd, you need also configure MCFG_NEW, MCTRL, WTIME,TAS and SMWT

f Before starting DMA, Wait for MSTATE.BUSY == 0.

g Set MCTRL.DMATXEN to 1 to prepare DMA transfer.
Note that if you don't want to stop DMA transfer, you need not to check MSTATE.BUSY.

h Set LCDCCTRL.ENA to 1 to start LCDC internal DMA.

i The LCDC internal DMA will transfer data to SLCDC, and SLCDC transfer data to LCM. Repeat this step till you want to close the SLCDC to transfer data to LCM Panel.

*please notice that use and only use DMA0 to transfer command no matter use DMA0 to transfer frame data or not.

One recommend descriptor chain (CMD0 with CNUM>0 and CMD1 with CNUM=0):

2 Stop DMA transfer.

a Set LCDCCTRL.ENA to 0 to stop LCDC internal DMA at once.

b Wait till MSTATE.BUSY is set to 0 by hardware.
MSTATE.BUSY == 1: there is data in the FIFO waited for transferring to LCM.
MSTATE.BUSY == 0: all data in the FIFO have finished transferring to LCM.

c Set MCTRL.DMATXEN to 0 to stop DMA transfer.

3 Restart DMA transfer.

When MCTRL.DMATXEN is set to 0, and then you want to restart DMA transfer at once, you should ensure that MCTRL.DMATXEN must keep 0 at least three cycles of PIXCLK.

7.8.2 Register Operation

1 Set MCFG, MCFG_NEW, MCTRL, WTIME, TAS, SMWT to configure SLCD.

2 Wait for MSTATE.BUSY == 0.

3 Set MDATA register.

4 Wait for MSTATE.BUSY == 0.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5 Set MDATA register.

6 Wait for MSTATE.BUSY == 0.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8 MIPI DSI Controller

8.1 Overview

The Display Serial Interface (DSI) is part of a group of communication protocols defined by the MIPI Alliance. The MIPI DSI Controller is a host digital core that implements all protocol functions defined in the MIPI DSI Specification. The MIPI DSI controller provides an interface between the system and the MIPI D-PHY, allowing the communication with a DSI-compliant display.

Features:

- Compliant with MIPI Alliance standards
- Supports all commands defined in the MIPI Alliance specification for Display Command Set (DCS)
- Supports up to two D-PHY data lanes
- Bidirectional communication and escape mode support through data lane 0
- Supports non-continuous clock in D-PHY clock lane for additional power saving
- Supports Ultra Low-Power mode with PLL disabled
- ECC and Checksum capabilities
- Support for End of Transmission Packet (EoTp)
- Fault recovery schemes
- 3D transmission support
- Enhanced DPI (eDPI) interface features
- Support for all DPI features (Standard Video Mode)
- Additional support for sending large amounts of data through the memory_write_start (WMS) and memory_write_continue (WMC) DCS command
- Vertical and horizontal color bar generation without DPI stimuli
- BER pattern without DPI stimuli

8.2 Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute

   R - Read only

   W - Write only

   RW - read and write

   RCW - read and write, but clear to 0 by read

   RSW - read and write, but set to 1 by read

   RWC - read and write, clear to 0 by write 1, write 0 has no effect

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

RWS - read and write, set to 1 by write 1, write 0 has no effect
RC - read only, and clear to 0 by read
RS - read only, and set to 1 by read
SPEC - special access method, relate to its description

3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

8.2.1 Registers Memory Map

Table 8-1 Registers Memory Map-Address Base

Name	Base	Description
MIPI_DSI	0x103014000	Address base of MIPI DSI controller

Table 8-2 MIPI DSI Register Description

Name	Description	Reset Value	Address Offset	Access Size
VERSION	Version of the DSI host controller	0x3133302A	0x000	32
PWR_UP	Core power-up	0x00000000	0x004	32
CLKMGR_CFG	Configuration of the internal clock dividers	0x00000000	0x008	32
DPI_VCID	DPI virtual channel id	0x00000000	0x00C	32
DPI_COLOR_CODING	DPI color coding	0x00000000	0x010	32
DPI_CFG_POL	DPI polarity configuration	0x00000000	0x014	32
DPI_LP_CMD_TIM	Low-power command timing configuration	0x00000000	0x018	32
PCKHDL_CFG	Packet handler configuration	0x00000000	0x02C	32
GEN_VCID	Generic interface virtual channel id	0x00000000	0x030	32
MODE_CFG	Video or Command mode selection	0x00000001	0x034	32
VID_MODE_CFG	Video mode configuration	0x00000000	0x038	32
VID_PKT_SIZE	Video packet size	0x00000000	0x03C	32
VID_NUM_CHUNKS	Number of chunks	0x00000000	0x040	32
VID_NULL_SIZE	Null packet size	0x00000000	0x044	32
VID_HSA_TIME	Horizontal Sync Active time	0x00000000	0x048	32
VID_HBP_TIME	Horizontal Back Porch time	0x00000000	0x04C	32
VID_HLINE_TIME	Line time	0x00000000	0x050	32
VID_VSA_LINES	Vertical Sync Active period	0x00000000	0x054	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

VID_VBP_LINES	Vertical Back Porch period	0x00000000	0x058	32
VID_VFP_LINES	Vertical Front Porch period	0x00000000	0x05C	32
VID_VACTIVE_LINES	Vertical resolution	0x00000000	0x060	32
EDPI_CMD_SIZE	Size for eDPI packets	0x00000000	0x064	32
CMD_MODE_CFG	Command mode configuration	0x00000000	0x068	32
GEN_HDR	Generic packet header configuration	0x00000000	0x06C	32
GEN_PLD_DATA	Generic payload data in and out	0x00000000	0x070	32
CMD_PKT_STATUS	Command packet status	0x00001515	0x074	32
TO_CNT_CFG	Timeout timers configuration	0x00000000	0x078	32
HS_RD_TO_CNT	Peripheral response timeout definition after high-speed read operations	0x00000000	0x07C	32
LP_RD_TO_CNT	Peripheral response timeout definition after low-power read operations	0x00000000	0x080	32
HS_WR_TO_CNT	Peripheral response timeout definition after high-speed write operations	0x00000000	0x084	32
LP_WR_TO_CNT	Peripheral response timeout definition after low-power write operations	0x00000000	0x088	32
BTA_TO_CNT	Peripheral response timeout definition after bus turnaround	0x00000000	0x08C	32
SDF_3D	3D control configuration	0x00000000	0x090	32
LPCLK_CTRL	Low-power in clock lane	0x00000000	0x094	32
PHY_TMR_LPCLK_CFG	D-PHY timing configuration for the clock lane	0x00000000	0x098	32
PHY_TMR_CFG	D-PHY timing configuration for data lanes	0x00000000	0x09C	32
PHY_RSTZ	D-PHY reset control	0x00000000	0x0A0	32
PHY_IF_CFG	D-PHY interface configuration	0x00000000	0x0A4	32
PHY_ULPS_CTRL	D-PHY Ultra Low-Power control	0x00000000	0x0A8	32
PHY_TX_TRIGGERS	D-PHY transmit triggers	0x00000000	0x0AC	32
PHY_STATUS	D-PHY PPI status interface	0x00001528	0x0B0	32
PHY_TST_CTRL0	D-PHY test interface control 0	0x00000001	0x0B4	32
PHY_TST_CTRL1	D-PHY test interface control 1	0x00000000	0x0B8	32
INT_ST0	Interrupt status register 0	0x00000000	0x0BC	32
INT_ST0	Interrupt status register 1	0x00000000	0x0C0	32
INT_MSK0	Masks the interrupt generation triggered by the INT_ST0 register	0x00000000	0x0C4	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

INT_MSK1	Masks the interrupt generation triggered by the INT_ST1 register	0x00000000	0x0C8	32
INT_FORCE0	Force interrupts that affect the INT_ST0 register	0x00000000	0x0D8	32
INT_FORCE1	Force interrupts that affect the INT_ST1 register	0x00000000	0x0DC	32
VID_SHADOW_CTRL	Controls DPI shadow register mechanism feature	0x00000000	0x100	32
DPI_VCID_ACT	Configures the virtual channel id for the DPI traffic	0x00000000	0x10C	32

8.2.2 Registers and Fields Description

8.2.2.1 VERSION

This register contains the version of the MIPI DSI Host Controller

																																BASE + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION																															
RST	0	0	0	1	1	0	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	1	0	1	0	1	0

Bits	Name	Description	RW
31:0	VERSION	This field indicates the version of the MIPI DSI Host Controller	R

8.2.2.2 PWR_UP

This register controls the power-up of the MIPI DSI Controller.

																																BASE + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															shutdownz
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:1	Reserved	Reserved	R
0	shutdownz	This bit configures the core either to power up or to reset. shutdownz is	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		the soft reset register. Its default value is 0. After the core configuration, to enable the MIPI DSI Controller, set this register to 1. 0: Reset 1: Power-up	

8.2.2.3 CLKMGR_CFG

This register configures the factor for the internal dividers to divide lanebyteclk for timeout purpose.

																																	BASE + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																to_clk_division								tx_esc_clk_division							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Reserved	R
15:8	to_clk_division	This field indicates the division factor for the Time Out clock used as the timing unit in the configuration of HS to LP and LP to HS transition error.	RW
7:0	tx_esc_clk_division	This field indicates the division factor for the TX Escape clock source (lanebyteclk). The values 0 and 1 stop the TX_ESC clock generation.	RW

8.2.2.4 DPI_VCID

This register configures the virtual channel id for DPI traffic.

																																	BASE + 0x00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														dpi_vcid	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:2	Reserved	Reserved	R
1:0	dpi_vcid	This field configures the DPI virtual channel id that is indexed to the Video mode packets.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.5 DPI_COLOR_CODING

This register configures the DPI color coding.

																																BASE + 0x010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																							loose18_en	Reserved				dpi_color_coding			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:9	Reserved	Reserved	R
8	loosely18_en	When set to 1, this bit activates loosely packed variant to 18-bit configurations.	RW
7:4	Reserved	Reserved	R
3:0	dpi_color_coding	This field configures the DPI color coding as follows: 0000: 16-bit configuration 1 0001: 16-bit configuration 2 0010: 16-bit configuration 3 0011: 18-bit configuration 1 0100: 18-bit configuration 2 0101: 24-bit 0110-1111: Reserved	RW

8.2.2.6 DPI_CFG_POL

This register configures the polarity of the DPI signals.

																																BASE + 0x014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											colorm_act	shutd_act	hsync_act	vsync_act	dataen_act
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:5	Reserved	Reserved	R
4	colorm_active_low	When set to 1, this bit configures the color mode pin as active low.	RW
3	shutd_active_low	When set to 1, this bit configures the shutdown pin as active low.	RW
2	hsync_active_low	When set to 1, this bit configures the horizontal synchronism pin as active low.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	vsync_active_low	When set to 1, this bit configures the vertical synchronism pin as active low.	RW
0	dataen_active_low	When set to 1, this bit configures the data enable pin as active low.	RW

8.2.2.7 DPI_LP_CMD_TIM

This register configures the timing for the low-power commands sent while in video mode.

																																	BASE + 0x018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								outvact_lpcmd_line								Reserved								invact_lpcmd_time							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Reserved	R
23:16	outvact_lpcmd_time	This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VSA, VBP, and VFP regions.	RW
15:8	Reserved	Reserved	R
7:0	invact_lpcmd_time	This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VACT region.	RW

8.2.2.8 PCKHDL_CFG

This register configures how EoTp, BTA, CRC, and ECC are to be used, to meet the peripheral's characteristics.

																																	BASE + 0x02C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											crc_rx_en	ecc_rx_en	bta_en	eotp_rx_en	eotp_tx_en
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:5	Reserved	Reserved	R
4	crc_rx_en	When set to 1, this bit enables the CRC reception and error reporting.	RW
3	ecc_rx_en	When set to 1, this bit enables the ECC reception, error correction, and reporting.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2	bat_en	When set to 1, this bit enables the Bus Turn-Around (BTA) request.	RW
1	etop_rx_en	When set to 1, this bit enables the EoTp reception.	RW
0	etop_tx_en	When set to 1, this bit enables the EoTp transmission.	RW

8.2.2.9 GEN_VCID

This register configures the virtual channel id of the read responses to store and return to the Generic interface.

																																	BASE + 0x030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														gen_vcid	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:2	Reserved	Reserved	R
1:0	gen_vcid_rx	This field indicates the Generic interface read-back virtual channel identification.	RW

8.2.2.10 MODE_CFG

This register configures the mode of operation - Video mode or Command mode (Commands can be sent even in video mode too).

																																	BASE + 0x034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															cmd_video
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1

Bits	Name	Description	RW
31:1	Reserved	Reserved	R
0	cmd_video_mode	This bit configures the operation mode: 0: Video mode 1: Command mode	RW

8.2.2.11 VID_MODE_CFG

This register configures several aspects of the Video mode operation such as the transmission mode, switching to low-power in the middle of a frame, enabling acknowledge,

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

and whether to send the commands in low-power.

																																	BASE + 0x038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							vpg_orient	Reserved			vpg_mode	Reserved			vpg_en	lp_cmd_en	frame_bta	lp_hfp_en	lp_hpb_en	lp_vact_en	lp_vfp_en	lp_vbp_en	lp_vsa_en	Reserved						vid_mode	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Reserved	R
24	vpg_orientation	This field indicates the color bar orientation as follows: 0: Vertical mode 1: Horizontal mode	RW
23:21	Reserved	Reserved	R
20	vpg_mode	This field is to select the pattern: 0: Color bar (horizontal or vertical) 1: BER pattern (vertical only)	RW
19:17	Reserved	Reserved	R
16	vpg_en	When set to 1, this bit enables the video mode pattern generator.	RW
15	lp_cmd_en	When set to 1, this bit enables the command transmission only in low-power mode.	RW
14	frame_bta_ack_en	When set to 1, this bit enables the request for an acknowledge response at the end of a frame.	RW
13	lp_hfp_en	When set to 1, this bit enables the return to low-power inside the Horizontal Front Porch (HFP) period when timing allows.	RW
12	lp_hbp_en	When set to 1, this bit enables the return to low-power inside the Horizontal Back Porch (HBP) period when timing allows.	RW
11	lp_vact_en	When set to 1, this bit enables the return to low-power inside the Vertical Active (VACT) period when timing allows.	RW
10	lp_vfp_en	When set to 1, this bit enables the return to low-power inside the Vertical Front Porch (VFP) period when timing allows.	RW
9	lp_vbp_en	When set to 1, this bit enables the return to low-power inside the Vertical Back Porch (VBP) period when timing allows.	RW
8	lp_vsa_en	When set to 1, this bit enables the return to low-power inside the Vertical Sync Time (VSA) period when timing allows.	RW
7:2	Reserved	Reserved	R
1:0	vid_mode_type	This field indicates the video mode transmission type as follows: 00: Non-burst with sync pulses 01: Non-burst with sync events 10 and 11: Burst mode	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.12 VID_PKT_SIZE

This register configures the video packet size.

																																BASE + 0x03C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																		vid_pkt_size													
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:14	Reserved	Reserved	R
13:0	vid_pkt_size	This field configures the number of pixels in a single video packet. For 18-bit not loosely packed data types, this number must be a multiple of 4.	RW

8.2.2.13 VID_NUM_CHUNKS

This register configures the number of chunks to use. The data in each chunk has the size provided by VID_PKT_SIZE.

																																BASE + 0x040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			vid_num_chunks												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:13	Reserved	Reserved	R
12:0	vid_num_chunks	This register configures the number of chunks to be transmitted during a Line period (a chunk consists of a video packet and a null packet). If set to 0 or 1, the video line is transmitted in a single packet. If set to 1, the packet is part of a chunk, so a null packet follows it if vid_null_size > 0. Otherwise, multiple chunks are used to transmit each video line.	RW

8.2.2.14 VID_NULL_SIZE

This register configures the size of null packets.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

																																BASE + 0x044
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			vid_null_size												
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:13	Reserved	Reserved	R
12:0	vid_null_size	This register configures the number of bytes inside a null packet. Setting it to 0 disables the null packets.	RW

8.2.2.15 VID_HSA_TIME

This register configures the video Horizontal Synchronism Active (HSA) time.

																																BASE + 0x048
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				vid_hsa_time											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:12	Reserved	Reserved	R
11:0	vid_hsa_time	This field configures the Horizontal Synchronism Active period in lane byte clock cycles.	RW

8.2.2.16 VID_HBP_TIME

This register configures the video Horizontal Back Porch (HBP) time.

																																BASE + 0x04C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				vid_hbp_time											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:12	Reserved	Reserved	R
11:0	vid_hbp_time	This field configures the Horizontal Back Porch period in lane byte clock cycles.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.17 VID_HLINE_TIME

This register configures the overall time for each video line

																																BASE + 0x050
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																vid_hline_time															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:15	Reserved	Reserved	R
14:0	vid_hline_time	This field configures the size of the total line time (HSA+HBP+HACT+HFP) counted in lane byte clock cycles.	RW

8.2.2.18 VID_HSA_LINES

This register configures the Vertical Synchronism Active (VSA) period.

																																BASE + 0x054
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																						vsa_lines									
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:10	Reserved	Reserved	R
9:0	vsa_lines	This field configures the Vertical Synchronism Active period measured in number of horizontal lines.	RW

8.2.2.19 VID_VBP_HLINES

This register configures the Vertical Back Porch (VBP) period.

																																BASE + 0x058
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																						vbp_lines									
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:10	Reserved	Reserved	R
9:0	vbp_lines	This field configures the Vertical Back Porch period measured in number of horizontal lines.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.20 VID_VFP_LINES

This register configures the Vertical Front Porch (VFP) period.

																																BASE + 0x05C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																						vfp_lines									
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:10	Reserved	Reserved	R
9:0	vfp_lines	This field configures the Vertical Front Porch period measured in number of horizontal lines.	RW

8.2.2.21 VID_VACTIVE_LINES

This register configures the vertical resolution of the video.

																																BASE + 0x060
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																		v_active_lines													
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:14	Reserved	Reserved	R
13:0	v_active_lines	This field configures the Vertical Active period measured in number of horizontal lines.	RW

8.2.2.22 EDPI_CMD_SIZE

This register configures the size of the eDPI packets.

																																BASE + 0x064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																edpi_allowed_cmd_size															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	Reserved	Reserved	R
15:0	edpi_allowed_cmd_size	This field configures the maximum allowed size for an eDPI write memory command, measured in pixels. Automatic partitioning of data obtained from eDPI is permanently enabled.	RW

8.2.2.23 CMD_MODE_CFG

This register configures several aspects of the command mode operation, tearing effect, acknowledge for each packet, and the speed mode to transmit each data type related to commands.

																																BASE + 0x068
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							max_rd_pk	Reserved				dcs_lw_tx	dcs_sr_0p	dcs_sr_1p	dcs_sw_0p	Reserved	gen_lw_tx	gen_sr_2p	gen_sr_1p	gen_sr_0p	gen_sw_2	gen_sw_1	gen_sw_0	Reserved						ack_rqst	tear_fx_en
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Reserved	R
24	max_rd_pkt_size	This bit configures the maximum read packet size command transmission type: 0: High-speed 1: Low-power	RW
23:20	Reserved	Reserved	R
19	dcs_lw_tx	This bit configures the DCS long write packet command transmission type: 0: High-speed 1: Low-power	RW
18	dcs_sr_0p_tx	This bit configures the DCS short read packet with zero parameter command transmission type: 0: High-speed 1: Low-power	RW
17	dcs_sw_1p_tx	This bit configures the DCS short write packet with one parameter command transmission type: 0: High-speed 1: Low-power	RW
16	dcs_sw_0p_tx	This bit configures the DCS short write packet with zero parameter command transmission type: 0: High-speed 1: Low-power	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15	Reserved	Reserved	R
14	gen_lw_tx	This bit configures the Generic long write packet command transmission type: 0: High-speed 1: Low-power	RW
13	gen_sr_2p_tx	This bit configures the Generic short read packet with two parameters command transmission type: 0: High-speed 1: Low-power	RW
12	gen_sr_lp_tx	This bit configures the Generic short read packet with one parameter command transmission type: 0: High-speed 1: Low-power	RW
11	gen_sr_0p_tx	This bit configures the Generic short read packet with zero parameter command transmission type: 0: High-speed 1: Low-power	RW
10	gen_sw_2p_tx	This bit configures the Generic short write packet with two parameters command transmission type: 0: High-speed 1: Low-power	RW
9	gen_sw_1p_tx	This bit configures the Generic short write packet with one parameter command transmission type: 0: High-speed 1: Low-power	RW
8	gen_sw_0p_tx	This bit configures the Generic short write packet with zero parameter command transmission type: 0: High-speed 1: Low-power	RW
7:2	Reserved	Reserved	R
1	ack_rqst_en	When set to 1, this bit enables the acknowledge request after each packet transmission.	RW
0	tear_fx_en	When set to 1, this bit enables the tearing effect acknowledge request.	RW

8.2.2.24 GEN_HDR

This register sets the header for new packets sent using the Generic interface.

																																BASE + 0x06C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved								gen_wc_msbyte								gen_wc_lsbyte								gen_vc		gen_dt					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Reserved	R
23:16	gen_wc_msbyte	This field configures the most significant byte of the header packet's word count for long packets or data 1 for short packets.	RW
15:8	gen_wc_lsbyte	This field configures the least significant byte of the header packet's Word count for long packets or data 0 for short packets.	RW
7:6	gen_vc	This field configures the virtual channel id of the header packet.	RW
5:0	gen_dt	This field configures the packet data type of the header packet.	RW

8.2.2.25 GEN_PLD_DATA

This register sets the payload for the packets sent using the Generic interface and, when read, returns the contents of the read responses from the peripheral.

																																BASE + 0x070
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	gen_pld_b4								gen_pld_b3								gen_pld_b2								gen_pld_b1							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	gen_pld_b4	This field indicates byte 4 of the packet payload.	RW
23:16	gen_pld_b3	This field indicates byte 3 of the packet payload.	RW
15:8	gen_pld_b2	This field indicates byte 2 of the packet payload.	RW
7:0	gen_pld_b1	This field indicates byte 1 of the packet payload.	RW

8.2.2.26 CMD_PKT_STATUS

This register contains information about the status of the FIFOs related to the DBI and the Generic interfaces.

																																BASE + 0x074
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																									gen_rd_cm	gen_pld_r	gen_pld_r	gen_pld_f	gen_pld_e	gen_cmd_f	gen_cd_e
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:7	Reserved	Reserved	R
6	gen_rd_cmd_busy	This bit is set when a read command is issued and cleared when the entire response is stored in the FIFO.	
5	gen_pld_r_full	This bit indicates the full status of the generic read payload FIFO.	R
4	gen_pld_r_empty	This bit indicates the empty status of the generic read payload FIFO.	R
3	gen_pld_w_ful	This bit indicates the full status of the generic write payload FIFO.	R
2	gen_pld_w_empty	This bit indicates the empty status of the generic write payload FIFO.	R
1	gen_cmd_full	This bit indicates the full status of the generic command FIFO.	R
0	gen_cmd_empty	This bit indicates the empty status of the generic command FIFO.	R

8.2.2.27 TO_CNT_CFG

This register configures the counters that trigger the timeout errors. These are used to warn the system about a failure, through an interrupt, and restart the core in case of unexpected situations that cause deadlock conditions.

																																		BASE + 0x078
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	hstx_to_cnt																lprx_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	hstx_to_cnt	This field configures the timeout counter that triggers a high-speed transmission timeout contention detection (measured in TO_CLK_DIVISION cycles). If using the non-burst mode and there is no sufficient time to switch from HS to LP and back in the period which is from one line data finishing to the next line sync start, the DSI link returns the LP state once per frame, then you should configure the TO_CLK_DIVISION and hstx_to_cnt to be in accordance with: hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION >= the time of one FRAME data transmission * (1 + 10%) In burst mode, RGB pixel packets are time-compressed, leaving more time during a scan line. Therefore, if in burst mode and there is sufficient time to switch from HS to LP and back in the period of time from one line data finishing to the next line sync start, the DSI link can return LP mode and back in this	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		time interval to save power. For this, configure the TO_CLK_DIVISION and hstx_to_cnt to be in accordance with: hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION >= the time of one LINE data transmission * (1 + 10%)	
15:0	lprx_to_cnt	This field configures the timeout counter that triggers a low-power reception timeout contention detection (measured in TO_CLK_DIVISION cycles).	RW

8.2.2.28 HS_RD_TO_CNT

This register configures the Peripheral Response timeout after high-speed read operations.

																																BASE + 0x07C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																hs_rd_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Reserved	R
15:0	hs_rd_to_cnt	This field sets a period for which the MIPI DSI Controller keeps the link still, after sending a high-speed read operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.	RW

8.2.2.29 LP_RD_TO_CNT

This register configures the Peripheral Response timeout after the low-power read operations..

																																BASE + 0x080
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																lp_rd_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:1	Reserved	Reserved	R
0	lp_rd_to_cnt	This field sets a period for which the MIPI DSI Controller keeps the link still, after sending a low-power read operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.30 HS_WR_TO_CNT

This register configures the Peripheral Response timeout after the high-speed write operations.

																																	BASE + 0x084
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							presp_to_	Reserved								hs_wr_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Reserved	R
24	presp_to_mode	When set to 1, this bit ensures that the peripheral response timeout caused by hs_wr_to_cnt is used only once per eDPI frame, when both the following conditions are met: dpivsync_edpiwms has risen and fallen. Packets originated from eDPI have been transmitted and its FIFO is empty again. In this scenario no non-eDPI requests are sent to the D-PHY, even if there is traffic from generic or DBI ready to be sent, making it return to stop state. When it does so, PRESP_TO counter is activated and only when it finishes does the controller send any other traffic that is ready.	RW
23:16	Reserved	Reserved	R
15:0	hs_wr_to_cnt	This field sets a period for which the MIPI DSI Controller keeps the link inactive after sending a high-speed write operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.	RW

8.2.2.31 LP_WR_TO_CNT

This register configures the Peripheral Response timeout after the low-power write operations.

																																	BASE + 0x088
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																lp_wr_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	Reserved	Reserved	R
15:0	lp_wr_to_cnt	This field sets a period for which the MIPI DSI Controller keeps the link still, after sending a low-power write operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.	RW

8.2.2.32 BTA_TO_CNT

This register configures the Peripheral Response timeout after the Bus Turnaround completion.

																		BASE + 0x08C														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																bta_to_cnt															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Reserved	R
15:0	bta_to_cnt	This field sets a period for which the MIPI DSI Controller keeps the link still, after completing a Bus Turn-Around. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.	RW

8.2.2.33 SOF_3D

This register stores the 3D control information for the VSS packets in video mode.

																																BASE + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															send_3d	Reserved										right_fist	second_vs	format_3d		mode_3d	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:17	Reserved	Reserved	R
16	send_3d_cfg	When set, causes the next VSS packet to include 3D control payload in every VSS packet.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:6	Reserved	Reserved	R
5	right_first	This bit defines the left or right order: 0: Left eye data is sent first, and then the right eye data is sent. 1: Right eye data is sent first, and then the left eye data is sent.	RW
4	second_vsync	This field defines whether there is a second VSYNC pulse between Left and Right Images, when 3D Image Format is Frame-based: 0: No sync pulses between left and right data 1: Sync pulse (HSYNC, VSYNC, blanking) between left and right data	RW
3:2	format_3d	This field defines the 3D image format: 00: Line (alternating lines of left and right data) 01: Frame (alternating frames of left and right data) 10: Pixel (alternating pixels of left and right data) 11: Reserved	RW
1:0	mode_3d	This field defines the 3D mode on/off and display orientation: 00: 3D mode off (2D mode on) 01: 3D mode on, portrait orientation 10: 3D mode on, landscape orientation 11: Reserved	RW

8.2.2.34 LPCLK_CTRL

This register configures using the non-continuous clock in the clock lane.

																																	BASE + 0x004	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														auto_clkla	phy_txreq
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:2	Reserved	Reserved	R
1	auto_clklane_ctrl	This bit enables the automatic mechanism to stop providing clock in the clock lane when time allows.	RW
0	phy_txrequestclkhs	This bit controls the D-PHY PPI txrequestclkhs signal.	RW

8.2.2.35 PHY_TMR_LPCLK_CFG

This register sets the time that the MIPI DSI Controller assumes in calculations for the clock lane to switch between high-speed and low-power.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

																																BASE + 0x098
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved						phy_clkhs2lp_time										Reserved						phy_clklp2hs_time									
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:26	Reserved	Reserved	R
25:16	phy_clkhs2lp_time	This field configures the maximum time that the D-PHY clock lane takes to go from high-speed to low-power transmission measured in lane byte clock cycles.	RW
15:10	Reserved	Reserved	R
9:0	phy_clklp2hs_time	This field configures the maximum time that the D-PHY clock lane takes to go from low-power to high-speed transmission measured in lane byte clock cycles.	RW

8.2.2.36 PHY_TMR_CFG

This register sets the time that the MIPI DSI Controller assumes in calculations for the data lanes to switch between high-speed and low-power.

																																BASE + 0x09C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	phy_hs2lp_time								phy_lp2hs_time								Reserved	max_rd_time														
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	phy_hs2lp_time	This field configures the maximum time that the D-PHY data lanes take to go from high-speed to low-power transmission measured in lane byte clock cycles.	R
23:16	phy_lp2hs_time	This field configures the maximum time that the D-PHY data lanes take to go from low-power to high-speed transmission measured in lane byte clock cycles.	RW
15	Reserved	Reserved	R
14:0	max_rd_time	This field configures the maximum time required to perform a read command in lane byte clock cycles. This register can only be modified when no read command is in progress.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.37 PHY_RSTZ

This register controls the resets and the PLL of the D-PHY.

																																BASE + 0x0A0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																												phy_forcep	phy_enab	phy_rstz	phy_shutd
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:4	Reserved	Reserved	R
3	phy_forcepll	When the D-PHY is in ULPS, this bit enables the D-PHY PLL..	RW
2	phy_enableclk	When set to1, this bit enables the D-PHY Clock Lane module.	RW
1	phy_rstz	When set to 0, this bit places the digital section of the D-PHY in the reset state.	RW
0	phy_shutdownz	When set to 0, this bit places the D-PHY macro in power-down state.	RW

8.2.2.38 PHY_IF_CFG

This register configures the number of active lanes and the minimum time to remain in the Stop state.

																																BASE + 0x0A4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																phy_stop_wait_time								Reserved						n_lanes	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:1	Reserved	Reserved	R
15:8	phy_stop_wait_time	This field configures the minimum wait period to request a high-speed transmission after the Stop state.	RW
7:2	Reserved	Reserved	R
1:0	n_lanes	This field configures the number of active data lanes: 00: One data lane (lane 0) 01: Two data lanes (lanes 0 and 1) 10-11: Reserved	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.39 PHY_ULPS_CTRL

This register configures entering and leaving the ULPS in the D-PHY.

																																BASE + 0x0A8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																												phy_txexit	phy_txreq	phy_txexit	phy_txreq
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:4	Reserved	Reserved	R
3	phy_txexitulpslan	ULPS mode Exit on all active data lanes.	RW
2	phy_txrequlpslan	ULPS mode Request on all active data lanes.	RW
1	phy_txexitulpsclk	ULPS mode Exit on clock lane.	RW
0	phy_txrequlpsclk	ULPS mode Request on clock lane.	RW

8.2.2.40 PHY_TX_TRIGGERS

This register configures the signals that activate the triggers in the D-PHY.

																																BASE + 0x0AC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																												phy_tx_tri			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:4	Reserved	Reserved	R
3:0	phy_tx_triggers	This field controls the trigger transmissions.	RW

8.2.2.41 PHY_STATUS

This register contains information about the status of the D-PHY.

																																BASE + 0x0B0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																							phy_ulpsa	phy_stop	phy_rxulps	phy_ulpsa	phy_stop	phy_ulpsa	phy_stopst	phy_direct	phy_lock
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	0	1	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:9	Reserved	Reserved	R
8	phy_ulpsactivenot1lane	This bit indicates the status of ulpsactivenot1lane D-PHY signal.	RW
7	phy_stopstate1lane	This bit indicates the status of phystopstate1lane D-PHY signal.	RW
6	phy_rxulpsesc0lane	This bit indicates the status of rxulpsesc0lane D-PHY signal.	RW
5	phy_ulpsactivenot0lane	This bit indicates the status of ulpsactivenot0lane D-PHY signal.	RW
4	phy_stopstate0lane	This bit indicates the status of phystopstate0lane D-PHY signal.	RW
3	phy_ulpsactivenotclk	This bit indicates the status of phyulpsactivenotclk D-PHY signal.	RW
2	phy_stopstateclklane	This bit indicates the status of phystopstateclklane D-PHY signal.	RW
1	phy_direction	This bit indicates the status of phydirection D-PHY signal.	RW
0	phy_lock	This bit indicates the status of phylock D-PHY signal.	RW

8.2.2.42 PHY_TST_CTRL0

This register controls the clock and the clear signals of the D-PHY vendor specific interface.

																																	BASE + 0x0B4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														phy_testclk	phy_testclr
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bits	Name	Description	RW
31:2	Reserved	Reserved	R
1	phy_testclk	This bit is used to clock the TESTDIN bus into the D-P	
0	phy_testclr	PHY test interface clear (active high	RW

8.2.2.43 PHY_TST_CTRL1

This register controls the data and the enable pins of the D-PHY vendor specific interface.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

																																BASE + 0x0B8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															phy_testen	phy_testdout								phy_testdin							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:17	Reserved	Reserved	R
16	phy_testen	PHY test interface operation selector: 1: The address write operation is set on the falling edge of the testclk signal. 0: The data write operation is set on the rising edge of the testclk signal.	RW
15:8	pht_testdout	PHY output 8-bit data bus for read-back and internal probing functionalities.	R
7:0	phy_testdin	PHY test interface input 8-bit data bus for internal register programming and test functionalities access	RW

8.2.2.44 INT_ST0

This register contains the status of the interrupt sources from the acknowledge reports and the D-PHY.

																																BASE + 0x0BC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved											errors_4	errors_3	errors_2	errors_1	errors_0	ack_err_15	ack_err_14	ack_err_13	ack_err_12	ack_err_11	ack_err_10	ack_err_9	ack_err_8	ack_err_7	ack_err_6	ack_err_5	ack_err_4	ack_err_3	ack_err_2	ack_err_1	ack_err_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:21	Reserved	Reserved	R
20	dphy_errors_4	This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0.	R
19	dphy_errors_3	This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0.	R
18	dphy_errors_2	This bit indicates the ErrControl error from Lane 0.	R
17	dphy_errors_1	This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0.	R
16	dphy_errors_0	This bit indicates ErrEsc escape entry error from Lane 0.	R
15	ack_with_err_15	This bit retrieves the DSI protocol violation from the Acknowledge	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		error report.	
14	ack_with_err_14	This bit retrieves the reserved (specific to device) from the Acknowledge error report.	R
13	ack_with_err_13	This bit retrieves the invalid transmission length from the Acknowledge error report.	R
12	ack_with_err_12	This bit retrieves the DSI VC ID Invalid from the Acknowledge error report.	R
11	ack_with_err_11	This bit retrieves the not recognized DSI data type from the Acknowledge error report.	R
10	ack_with_err_10	This bit retrieves the checksum error (long packet only) from the Acknowledge error report.	R
9	ack_with_err_9	This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error report.	R
8	ack_with_err_8	This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error report.	R
7	ack_with_err_7	This bit retrieves the reserved (specific to device) from the Acknowledge error report.	R
6	ack_with_err_6	This bit retrieves the False Control error from the Acknowledge error report.	R
5	ack_with_err_5	This bit retrieves the Peripheral Timeout error from the Acknowledge Error report.	R
4	ack_with_err_4	This bit retrieves the LP Transmit Sync error from the Acknowledge error report.	R
3	ack_with_err_3	This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report.	R
2	ack_with_err_2	This bit retrieves the EoT Sync error from the Acknowledge error report.	R
1	ack_with_err_1	This bit retrieves the SoT Sync error from the Acknowledge error report.	R
0	ack_with_err_0	This bit retrieves the SoT error from the Acknowledge error report	R

8.2.2.45 INT_ST1

This register contains the status of the interrupt sources related to timeouts, ECC, CRC, packet size, EoTp, Generic, and DBI interfaces.

																																BASE + 0x0C0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			gen_pld_re	gen_pld_r	gen_pld_s	gen_pld_w	gen_cmd_	dpi_pld_w	eopt_err	pkt_size_e	crc_err	ecc_multi	ecc_si_e	to_lp_rx	to_hs_tx
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:13	Reserved	Reserved	R
12	gen_pld_recev_err	This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted.	R
11	gen_pld_rd_err	This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted.	R
10	gen_pld_send_err	This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent.	R
9	gen_pld_wr_err	This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written.	R
8	gen_cmd_wr_err	This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written.	R
7	dpi_pld_wr_err	This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted.	R
6	eopt_err	This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission.	R
5	pkt_size_err	This bit indicates that the packet size error is detected during the packet reception.	R
4	crc_err	This bit indicates that the CRC error is detected in the received packet payload.	R
3	ecc_multi_err	This bit indicates that the ECC multiple error is detected in a received packet.	R
2	ecc_single_err	This bit indicates that the ECC single error is detected and corrected in a received packet.	R
1	to_lp_rx	This bit indicates that the low-power reception timeout counter reached the end and contention is detected.	R
0	to_hs_tx	This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected.	R

8.2.2.46 INT_MSK0

This register configures the masks for the sources of interrupts that affect the INT_ST0 register.

																																BASE + 0x0C4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved											errors_4	errors_3	errors_2	errors_1	errors_0	ack_err_15	ack_err_14	ack_err_13	ack_err_12	ack_err_11	ack_err_10	ack_err_9	ack_err_8	ack_err_7	ack_err_6	ack_err_5	ack_err_4	ack_err_3	ack_err_2	ack_err_1	ack_err_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:21	Reserved	Reserved	R
20	dphy_errors_4	This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0.	R
19	dphy_errors_3	This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0.	R
18	dphy_errors_2	This bit indicates the ErrControl error from Lane 0.	R
17	dphy_errors_1	This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0.	R
16	dphy_errors_0	This bit indicates ErrEsc escape entry error from Lane 0.	R
15	ack_with_err_15	This bit retrieves the DSI protocol violation from the Acknowledge error report.	R
14	ack_with_err_14	This bit retrieves the reserved (specific to device) from the Acknowledge error report.	R
13	ack_with_err_13	This bit retrieves the invalid transmission length from the Acknowledge error report.	R
12	ack_with_err_12	This bit retrieves the DSI VC ID Invalid from the Acknowledge error report.	R
11	ack_with_err_11	This bit retrieves the not recognized DSI data type from the Acknowledge error report.	R
10	ack_with_err_10	This bit retrieves the checksum error (long packet only) from the Acknowledge error report.	R
9	ack_with_err_9	This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error report.	R
8	ack_with_err_8	This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error report.	R
7	ack_with_err_7	This bit retrieves the reserved (specific to device) from the Acknowledge error report.	R
6	ack_with_err_6	This bit retrieves the False Control error from the Acknowledge error report.	R
5	ack_with_err_5	This bit retrieves the Peripheral Timeout error from the Acknowledge Error report.	R
4	ack_with_err_4	This bit retrieves the LP Transmit Sync error from the Acknowledge error report.	R
3	ack_with_err_3	This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report.	R
2	ack_with_err_2	This bit retrieves the EoT Sync error from the Acknowledge error report.	R
1	ack_with_err_1	This bit retrieves the SoT Sync error from the Acknowledge error report.	R
0	ack_with_err_0	This bit retrieves the SoT error from the Acknowledge error report	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.47 INT_MSK1

This register configures the masks for the sources of the interrupts that affect the INT_ST1 register.

																																BASE + 0x0C8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			gen_pld_re	gen_pld_r	gen_pld_s	gen_pld_w	gen_cmd_	dpi_pld_w	eopt_err	pkt_size_e	crc_err	ecc_multi	ecc_si_e	to_lp_rx	to_hs_tx
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:13	Reserved	Reserved	R
12	gen_pld_recev_err	This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted.	R
11	gen_pld_rd_err	This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted.	R
10	gen_pld_send_err	This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent.	R
9	gen_pld_wr_err	This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written.	R
8	gen_cmd_wr_err	This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written.	R
7	dpi_pld_wr_err	This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted.	R
6	eopt_err	This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission.	R
5	pkt_size_err	This bit indicates that the packet size error is detected during the packet reception.	R
4	crc_err	This bit indicates that the CRC error is detected in the received packet payload.	R
3	ecc_multi_err	This bit indicates that the ECC multiple error is detected in a received packet.	R
2	ecc_single_err	This bit indicates that the ECC single error is detected and corrected in a received packet.	R
1	to_lp_rx	This bit indicates that the low-power reception timeout counter reached the end and contention is detected.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	to_hs_tx	This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected.	R

8.2.2.48 INT_FORCE0

This register configures the masks for the sources of interrupts that affect the INT_ST0 register.

																																	BASE + 0x0D8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved											errors_4	errors_3	errors_2	errors_1	errors_0	ack_err_15	ack_err_14	ack_err_13	ack_err_12	ack_err_11	ack_err_10	ack_err_9	ack_err_8	ack_err_7	ack_err_6	ack_err_5	ack_err_4	ack_err_3	ack_err_2	ack_err_1	ack_err_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:21	Reserved	Reserved	R
20	dphy_errors_4	This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0.	R
19	dphy_errors_3	This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0.	R
18	dphy_errors_2	This bit indicates the ErrControl error from Lane 0.	RW
17	dphy_errors_1	This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0.	R
16	dphy_errors_0	This bit indicates ErrEsc escape entry error from Lane 0.	R
15	ack_with_err_15	This bit retrieves the DSI protocol violation from the Acknowledge error report.	R
14	ack_with_err_14	This bit retrieves the reserved (specific to device) from the Acknowledge error report.	R
13	ack_with_err_13	This bit retrieves the invalid transmission length from the Acknowledge error report.	R
12	ack_with_err_12	This bit retrieves the DSI VC ID Invalid from the Acknowledge error report.	R
11	ack_with_err_11	This bit retrieves the not recognized DSI data type from the Acknowledge error report.	R
10	ack_with_err_10	This bit retrieves the checksum error (long packet only) from the Acknowledge error report.	R
9	ack_with_err_9	This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error report.	R
8	ack_with_err_8	This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error report.	R
7	ack_with_err_7	This bit retrieves the reserved (specific to device) from the	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Acknowledge error report.	
6	ack_with_err_6	This bit retrieves the False Control error from the Acknowledge error report.	R
5	ack_with_err_5	This bit retrieves the Peripheral Timeout error from the Acknowledge Error report.	R
4	ack_with_err_4	This bit retrieves the LP Transmit Sync error from the Acknowledge error report.	R
3	ack_with_err_3	This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report.	R
2	ack_with_err_2	This bit retrieves the EoT Sync error from the Acknowledge error report.	R
1	ack_with_err_1	This bit retrieves the SoT Sync error from the Acknowledge error report.	R
0	ack_with_err_0	This bit retrieves the SoT error from the Acknowledge error report	R

8.2.2.49 INT_FORCE1

This register configures the masks for the sources of the interrupts that affect the INT_ST1 register.

																																	BASE + 0x0DC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			gen_pld_re	gen_pld_r	gen_pld_s	gen_pld_w	gen_cmd_	dpi_pld_w	eopt_err	pkt_size_e	crc_err	ecc_multi	ecc_si_e	to_lp_rx	to_hs_tx
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:13	Reserved	Reserved	R
12	gen_pld_recev_err	This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted.	R
11	gen_pld_rd_err	This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted.	R
10	gen_pld_send_err	This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent.	R
9	gen_pld_wr_err	This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written.	R
8	gen_cmd_wr_err	This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7	dpi_pld_wr_err	This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted.	R
6	eopt_err	This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission.	R
5	pkt_size_err	This bit indicates that the packet size error is detected during the packet reception.	R
4	crc_err	This bit indicates that the CRC error is detected in the received packet payload.	R
3	ecc_multi_err	This bit indicates that the ECC multiple error is detected in a received packet.	R
2	ecc_single_err	This bit indicates that the ECC single error is detected and corrected in a received packet.	R
1	to_lp_rx	This bit indicates that the low-power reception timeout counter reached the end and contention is detected.	R
0	to_hs_tx	This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected.	R

8.2.2.50 VID_SHADOW_CTRL

This register controls the DPI shadow register mechanism feature.

																																	BASE + 0x100
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															vid_sh_pin	Reserved							vid_sh_req	Reserved							vid_sh_en
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:17	Reserved	Reserved	R
16	vid_shadow_pin_req	When set to 1, the video request is done by external pin. In this mode, vid_shadow_req is ignored.	RW
15:9	Reserved	Reserved	R
8	vid_shadow_req	When set to 1, the DPI registers are copied to the auxiliary registers. After copying, this bit is auto cleared.	RW
7:1	Reserved	Reserved	R
0	vid_shadow_en	When set to 1, DPI receives the active configuration from the auxiliary registers. When this bit is set along with the vid_shadow_req bit, the auxiliary registers are automatically updated.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.2.2.51 DPI_VCID_ACT

This register configures the virtual channel id for the DPI traffic.

																																	BASE + 0x10C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														dpi_vcid	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:2	Reserved	Reserved	R
1:0	dpi_vcid	This field configures the DPI virtual channel id that is indexed to the Video mode packets.	RW

8.2.3 Static and Dynamic Registers

Some of the registers of the MIPI DSI Controller are static and can be configured only when the MIPI DSI Controller is idle and not transmitting. The dynamic registers can be configured when needed and do not require the MIPI DSI Controller to be in reset or idle. Below table lists the static and dynamic registers.

Static Read/Write Registers	Dynamic Read/Write Registers	Static Read-Only Registers	Dynamic Read-Only/ Write-Only Registers
PWR_UP CLKMGR_CFG DPI_VCID PCKHDL_CFG VID_PKT_SIZE VID_HBP_TIME VID_VFP_LINES HS_RD_TO_CNT BTA_TO_CNT PHY_TMR_CFG DPI_COLOR_CODING DPI_CFG_POL DPI_LP_CMD_TIM GEN_VCID VID_MODE_CFG VID_NUM_CHUNKS VID_NULL_SIZE VID_HSA_TIME	MODE_CFG GEN_HDR GEN_PLD_DATA SDF_3D PHY_RSTZ PHY_ULPS_CTRL PHY_TX_TRIGGERS PHY_TST_CTRL0 PHY_TST_CTRL1 INT_MSK0 INT_MSK1 VID_SHADOW_CTRL	Read-Only: VERSION	Read-Only: CMD_PKT_STATUS PHY_STATUS INT_ST0 INT_ST1 DPI_VCID_ACT Write-Only: INT_FORCE0 INT_FORCE1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

VID_HLINE_TIME VID_VSA_LINES VID_VBP_LINES VID_VACTIVE_LINES EDPI_CMD_SIZE CMD_MODE_CFG TO_CNT_CFG LP_RD_TO_CNT HS_WR_TO_CNT LP_WR_TO_CNT LPCLK_CTRL			

8.3 MIPI DSI Host Controller Guide

To operate MIPI DSI Controller, you must be familiar with the MIPI DSI specification.

8.3.1 Initializing MIPI DSI Host Controller and D-PHY

8.3.1.1 Configuring the DPI Resolution for Transmission

1. Global configuration: Configure n_lanes (PHY_IF_CFG) to define the number of lanes with which the controller can perform high-speed transmissions.
2. Configure the DPI interface to define how the DPI interface interacts with the controller.
   1) Configure dpi_vcid (DPI_VCID): This field configures the virtual channel that the packets generated by the DPI interface is indexed to.
   2) Configure dpi_color_coding (DPI_COLOR_CODING): This field configures the bits per pixel that the interface transmits and also the variant configuration of each bpp. If you select 18 bpp, and multiple of four.
   3) Configure dataen_active_low (DPI_CFG_POL): This bit configures the polarity of the dpidataen signal and sets it to be active low.
   4) Configure vsync_active_low (DPI_CFG_POL): This bit configures the polarity of the dpivsync signal and sets it to be active low.
   5) Configure hsync_active_low (DPI_CFG_POL): This bit configures the polarity of the dpihsync signal and sets it to be active low.
   6) Configure shutd_active_low (DPI_CFG_POL): This bit configures the polarity of the dpishutdn signal and sets it to be active low.
   7) Configure colorm_active_low (DPI_CFG_POL): This bit configures the polarity of the dpicolorm signal and sets it to be active low.
   8) Configure loosely18_en (DPI_COLOR_CODING): This bit configures if the pixel packing is done loosely or packed when dpi_color_coding is 18 bpp. This bit enables loosely packing.
3. Select the Video transmission mode to define how the processor requires the video

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

line to be transported through the DSI link.

1) Configure the low-power transitions (VID_MODE_CFG): This defines the video periods which are permitted to go to low-power if there is time available to do so.
2) Configure frame_bta_ack_en (VID_MODE_CFG): This specifies if the controller should request the peripheral acknowledge message at the end of frames. Configure lp_cmd_en (VID_MODE_CFG): This specifies that commands are to be transmitted in low-power.
3) Burst mode:
   a) Configure the register field vid_mode_type (VID_MODE_CFG) with value 2'b1x. Configure vid_pkt_size (VID_PKT_SIZE) with the size of the active line period, measured in pixels.
   b) The fields vid_num_chunks and vid_null_size are ignored by the MIPI DSI Controller.
4) Non-Burst mode:
   a) Configure the vid_mode_type field (VID_MODE_CFG) with 2'b0x.
   b) Configure the vid_mode_type field (VID_MODE_CFG) with 2'b00 to enable the transmission of sync pulses.
   c) Configure the vid_mode_type field (VID_MODE_CFG) with 2'b01 to enable the transmission of sync events.
   d) Configure the vid_pkt_size field (VID_MODE_CFG) with the number of pixels to be transmitted in a single packet. Selecting this value depends on the available memory of the attached peripheral, if the data is first stored, or on the memory you want to select for the FIFOs in MIPI DSI Controller.
   e) Configure the vid_num_chunks field (VID_NUM_CHUNKS) with the number of packets to be transmitted per video line. The value of vid_pkt_size * vid_num_chunks is the number of pixels per line of video, except if vid_num_chunks is 0, which disables the multi-packets. If you set it to 1, there is still only one packet per line, but it can be part of a chunk, followed by a null packet.
   f) Configure the vid_null_size field (VID_NULL_SIZE) with the size of null packets to be inserted as part of the chunks. Setting it to 0 disables null packets.

4. Define the DPI Horizontal timing configuration as follows:
   1) Configure the vid_hline_time field (VID_HLINE_TIME) with the time taken by a DPI video line measured in cycles of lane byte clock (for a clock lane at 500 MHz the lane byte clock period is 8 ns). When the periods of DPI clock and lane byte clock are not multiples, the value to program the vid_hline_time needs to be rounded. A timing mismatch is introduced between the lines due to the rounding of configuration values. If the MIPI DSI Controller is configured not to go to low-power, this timing divergence accumulates on every line, introducing a significant amount of mismatch towards the end of the frame. The reason for this is that the MIPI DSI Controller cannot re-synchronize on

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

every new line because it transmits the blanking packets when the Horizontal Sync event occurs on the DPI interface. However, the accumulated mismatch should become extinct on the last line of a frame, where, according to the DSI specification, the link should always return to low-power regaining synchronization, when a new frame starts on a vertical sync event. If the accumulated timing mismatch is greater than the time in low-power on the last line, a malfunction occurs. This phenomenon can be avoided by configuring the MIPI DSI Controller to go to low-power once per line.

2) Configure the vid_hsa_time field (VID_HSA_TIME) with the time taken by a DPI Horizontal Sync Active period measured in cycles of lane byte clock (normally a period of 8 ns).
3) Configure the vid_hbp_time field (VID_HBP_TIME) with the time taken by the DPI Horizontal Back Porch period measured in cycles of lane byte clock (normally a period of 8 ns). Special attention should be given to the calculation of this parameter.

5. Define the Vertical line configuration:
   1) Configure the vsa_lines field (VID_VSA_LINES) with the number of lines existing in the DPI Vertical Sync Active period.
   2) Configure the vbp_lines field (VID_VBP_LINES]) with the number of lines existing in the DPI Vertical Back Porch period.
   3) Configure the vfp_lines field (VID_VFP_LINES) with the number of lines existing in the DPI Vertical Front Porch period.
   4) Configure the v_active_lines field (VID_VACTIVE_LINES) with the number of lines existing in the DPI Vertical Active Period.
6. Illustrates the steps for configuring the DPI packet transmission.

8.3.1.2 Initializing D-PHY

1. By default, the PHY_RSTZ register activates the D-PHY reset signals physhutdownz and phyrstz, and disables the enableclk signal. The PHY_TST_CTRL0 register, by default, asserts the testclr signal. All the D-PHY reset pins are activated by default.
2. Configure the PHY_IF_CFG register with the number of lanes to be used by the MIPI DSI Controller.
3. Configure the TX_ESC clock frequency to a frequency lower than 20 MHz, the maximum allowed frequency for the D-PHY escape mode. This is done by writing into tx_esc_clk_division field of the CLKMGR_CFG register. The tx_esc_clk_division field divides the byte clock, and generates a TX_ESC clock for the D-PHY.
4. Configure the D-PHY PLL clock frequency through the Test interface.
5. Configure bit 0 of the register LPCLK_CTRL to generate high-speed clock TXREQUESTHSCLK.
6. Only after the PLL is locked and clock lane is in the Stop state, D-PHY drives the correct LP sequence to configure the receiver end for HS.
7. The D-PHY starts transmitting HS clock on the clock lane.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.1.3 Video Mode Pattern Generator

1. Configure the MODE_CFG register to enable Video mode. Configure the Video mode type using VID_MODE_CFG[1:0] configuration pins.
2. Configure the DPI_COLOR_CODING register.
3. Configure the frame using registers.
4. Configure the pattern generation mode (VID_MODE_CFG [20]) and pattern orientation (VID_MODE_CFG[24]) and enable it (VID_MODE_CFG[16]).

8.3.2 ULPM Flow

8.3.2.1 Process Flow to Enter the ULPM

1. Verify the initial status of the MIPI DSI Controller:
   1) PHY_RSTZ[3:0] = 4'hF
   2) PHY_ULPS_CTRL[3:0] = 4'h0
   3) PHY_TX_TRIGGERS[3:0] = 4'h0
   4) Verify that all active lanes are in Stop state and the D-PHY PLL is locked:
      a) One-lane configuration:
         PHY_STATUS[6:4] = 3'h3
         PHY_STATUS[1:0] = 2'h1
      b) Two -lanes configuration:
         PHY_STATUS[8:4] = 5'h1B
         PHY_STATUS[1:0] = 2'h1
2. Set PHY_ULPS_CTRL [3:0] = 4'hA to enter ULPM in the data and the clock lanes.
3. Wait until the D-PHY active lanes enter into ULPM:
   1) One-lane configuration: PHY_STATUS[6:0] = 7'h001
   2) Two-lanes configuration: PHY_STATUS[8:0] = 9'h001

   The MIPI DSI Controller is now in ULPM. To turn off the D-PHY PLL, perform the next two steps.
4. Turn off the D-PHY PLL by setting PHY_RSTZ[3] = 1'b0
5. Wait until the D-PHY PLL is unlocked: PHY_STATUS [0] = 1'b0

8.3.2.2 Process Flow to Exit the ULPM

1. Verify that all active lanes are in ULPM:
   1) One-lane configuration: PHY_STATUS[6:1] = 6'h00
   2) Two-lanes configuration: PHY_STATUS[8:1] = 8'h00
2. If the D-PHY PLL is already locked (PHY_STATUS [0] = 1'b1), go to step 5, else continue.
3. Turn on the D-PHY PLL by setting PHY_RSTZ [3] = 1'b1.
4. Wait until D-PHY PLL locked ： PHY_STATUS[0] = 1'b1
5. Without deasserting the ULPM request bits, assert the Exit ULPM bits by setting PHY_ULPS_CTRL[3:0] = 4'hF.
6. Wait until all active lanes exit ULPM:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1) One-lane configuration:
   PHY_STATUS [5] = 1'b1
   PHY_STATUS [3] = 1'b1
2) Two-lanes configuration:
   PHY_STATUS [8] = 1'b1
   PHY_STATUS [5] = 1'b1
   PHY_STATUS [3] = 1'b1

7. Wait for 1 ms.
8. Deassert the ULPM requests and the ULPM exit bits by setting PHY_ULPS_CTRL [3:0] = 4'h0.
9. The MIPI DSI Controller is now in Stop state and the D-PHY PLL is locked:
   1) One-lane configuration:
      PHY_STATUS [6:4] = 3'h3
      PHY_STATUS [1:0] = 2'h1
   2) Two-lanes configuration:
      PHY_STATUS [8:4] = 5'h1B
      PHY_STATUS [1:0] = 2'h1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9 EPD Controller

9.1 Overview

The controller provides a low cost SOC solution for EPD applications.

Features:

- Supports multiple types of compatible EPD panels
- Supports different size up to 4096x4096@20Hz
- Supports 2/3/4 bits grayscale and color display
- Pixel base updating
- Supports hand-writing mode
- Supports SW LUT algorithm
- Supports AUTO-DU, AUTO-GC4 mode

9.2 EPDC Pin Description

Table 9-1 EPDC Pin Mapping

Name	TYPE1	TYPE2	I/O	PIN
gdclk	GDCLK	YCLK	Output	PC12
gdsp	GDSP	YDIOU	Output	PC13
gdoe	GDOE	YOE	Output	PC14
sdclk	SDCLK	XCLK	Output	PC15
sdoe	SDOE		Output	PC16
sdle	SDLE	LD	Output	PC17
sdce[1:0]	SDCE [1]	XDIOR	Output	PC19
	SDCE [0]	XDIOL	Output	PC18
sdat[7:0]	SDDO [7]	DATA [7]	Output	PC09
	SDDO [6]	DATA [6]	Output	PC08
	SDDO [5]	DATA [5]	Output	PC07
	SDDO [4]	DATA [4]	Output	PC06
	SDDO [3]	DATA [3]	Output	PC05
	SDDO [2]	DATA [2]	Output	PC04
	SDDO [1]	DATA [1]	Output	PC03
	SDDO [0]	DATA [0]	Output	PC02
pwr[3:0]	PWR3	PWR3	Output	PB25
	PWR2	PWR2	Output	PB24
	PWR1	PWR1	Output	PB23
	PWR0	PWR0	Output	PB22
	BD[1]	BD[1]	Output	PC27
	BD[0]	BD[0]	Output	PC26

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.3 Register Description EPD Controller Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

9.3.1 Registers Mapping

Table 9-2 Registers Memory Map-Address Base

Name	Base	Description
EPDC	0x130C0000	Address base of EPDC

Table 9-3 EPD Controller Registers

Name	Address Offset	Reset Value	Access Size
EPDC_CTRL	0x0000	0x00000000	32
EPDC_CFG	0x0004	0x00000000	32
EPDC_STA	0x0008	0x00000000	32
EPDC_ISRC	0x000C	0x00000000	32
EPDC_DIS0	0x0010	0x00000000	32
EPDC_DIS1	0x0014	0x00000000	32
EPDC_SIZE	0x0018	0x00000000	32
EPDC_VAT	0x0020	0x00000000	32
EPDC_DAV	0x0024	0x00000000	32
EPDC_DAH	0x0028	0x00000000	32
EPDC_VSYN	0x002C	0x00000000	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

EPDC_HSYN	0x0030	0x00000000	32
EPDC_GDCLK	0x0034	0x00000000	32
EPDC_GDOE	0x0038	0x00000000	32
EPDC_GDSP	0x003C	0x00000000	32
EPDC_SDOE	0x0040	0x00000000	32
EPDC_SDSP	0x0044	0x00000000	32
EPDC_PMGR0	0x0050	0x00000000	32
EPDC_PMGR1	0x0054	0x00000000	32
EPDC_PMGR2	0x0058	0x00000000	32
EPDC_PMGR3	0x005C	0x00000000	32
EPDC_PMGR4	0x0060	0x00000000	32
EPDC_LUTBF	0x0070	0x00000000	32
EPDC_LUTSIZE	0x0074	0x00000000	32
EPDC_CURBF	0x0080	0x00000000	32
EPDC_CURSIZE	0x0084	0x00000000	32
EPDC_WORK0BF	0x0090	0x00000000	32
EPDC_WORK0SIZE	0x0094	0x00000000	32
EPDC_WORK1BF	0x0098	0x00000000	32
EPDC_WORK1SIZE	0x009C	0x00000000	32
EPDC_VCOMBD0	0x0100	0x00000000	32
EPDC_VCOMBD1	0x0104	0x00000000	32
EPDC_VCOMBD2	0x0108	0x00000000	32
EPDC_VCOMBD3	0x010C	0x00000000	32
EPDC_VCOMBD4	0x0110	0x00000000	32
EPDC_VCOMBD5	0x0114	0x00000000	32
EPDC_VCOMBD6	0x0118	0x00000000	32
EPDC_VCOMBD7	0x011C	0x00000000	32
EPDC_VCOMBD8	0x0120	0x00000000	32
EPDC_VCOMBD9	0x0124	0x00000000	32
EPDC_VCOMBD10	0x0128	0x00000000	32
EPDC_VCOMBD11	0x012C	0x00000000	32
EPDC_VCOMBD12	0x0130	0x00000000	32
EPDC_VCOMBD13	0x0134	0x00000000	32
EPDC_VCOMBD14	0x0138	0x00000000	32
EPDC_VCOMBD15	0x013C	0x00000000	32
EPDC_PRI	0x0500	0x00000000	32

9.4 Registers Description

9.4.1 EPDC Control Registers

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_CTRL																															BASE + 0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								BDR_STRT	Reserved	PWROFF	PWRON	REF_STOP	REF_STRT	LUT_STRT	EPD_ENA
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7	BDR_STRT	Start refreshing border, cleared by HW when it stopped.	RWS
5	PWROFF	Start the power off sequence, cleared by HW when it finished.	RWS
4	PWRON	Start the power on sequence, cleared by HW when it finished.	RWS
3	REF_STOP	Stop refreshing process, cleared by HW when it stopped. HW will stop after all pixels are updated completely This bit will be disabled when AUTO_STOP is 1.	RWS
2	REF_STRT	Start refreshing process, cleared by HW when it stopped. This bit will be disabled when AUTO_STOP is 1.	RWS
1	LUT_STRT	Start LUT loading, cleared by HW when it finished.	RWS
0	EPD_ENA	Enable EPD controller output.	RW

9.4.2 EPDC Configuration Registers

	EPDC_CFG																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved					STPE2						STEP1					STEP								Reserved	LUT_MODE		AUTO_ATED	AUTO_STOP	IBPP		REF_MODE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
26:21	STEP2	Steps for GC4 mode , these bits is invalid when LUT_MODE is 00 or 01	RW
20:16	STEP1	Steps for DU mode , these bits is invalid when LUT_MODE is 00	RW
15:8	STEP	Steps of per pixel.	RW
6:5	LUT_MODE	00:1 mode : GC16 or GC4 or DU: LUT start from 0x0, step up to 256; 01: 2 mode : DU: LUT start from 0x0, step up to 32 GC16: LUT start from 0x800, step up to 128; 10: 3 mode:	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		DU: LUT start from 0x0, step up to 32 GC4: LUT start from 0x800, step up to 64 GC16: LUT start from 0x1800, step up to 128;	
4	AUTO_GATED	1: When epdc stop , the dma clock and core clock are gated. 0: When epdc stop , the dma clock and core clock are not gated.	RW
3	AUTO_STOP	1: Enable. 0: Disable	RW
2:1	IBPP	00: 4bit. 01: 3bit. 10: 2bit. 11: reserve	RW
0	REF_MODE	1: Partial update.0: Parallel update	RW

9.4.3 EPDC Status Register

	EPDC_STA																															BASE + 0x0008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPD_IDLE	PANEL_IDLE	BDR_IDLE	REF_IDLE	LUT_IDLE	Reserved													IFF2U	IFF1U	IFF0U	WFF1O	WFF0O	OFFU	BDR_DONE	Reserved	PWROFF	PWRON	REF_STOP	REF_STRT	LUT_DONE	FRM_END
RST	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	EPD_IDLE	This bit will be asserted when epdc is idle.	R
30	PANEL_IDLE	This bit will be asserted when epdc panel controller is idle.	R
29	BDR_IDLE	This bit will be asserted when epdc border refresh process is idle.	R
28	REF_IDLE	This bit will be asserted when epdc frame refresh process is idle.	R
27	LUT_IDLE	This bit will be asserted when epdc lut loading process is idle.	R
13	IFF2U	This bit will be asserted when epdc input fifo2 underflow.	RW
12	IFF1U	This bit will be asserted when epdc input fifo1 underflow.	RW
11	IFF0U	This bit will be asserted when epdc input fifo0 underflow.	RW
10	WFF1O	This bit will be asserted when epdc write-back fifo1 overflow.	RW
9	WFF0O	This bit will be asserted when epdc write-back fifo0 overflow.	RW
8	OFFU	This bit will be asserted when epdc output fifo underflow.	RW
7	BDR_DONE	This bit will be asserted when border refreshed completely.	RW
5	PWROFF	This bit will be asserted when epdc power off sequence done.	RW
4	PWRON	This bit will be asserted when epdc power on sequence done.	RW
3	REF_STOP	This bit will be asserted when epdc stopped.	RW
2	REF_STRT	This bit will be asserted when epdc begin to refresh.	RW
1	LUT_DONE	This bit will be asserted when LUT loaded completely.	RW
0	FRM_END	This bit will be asserted when one frame output to panel completely.	RW

NOTE: The bits of EPDE_STA, which were RW, were written 0 to be cleared, and written 1 has no effect.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.4.4 EPDC ISR Control Register

	EPDC_ISRC																															BASE + 0x000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																		IFF2U_MASK	IFF1U_MASK	IFF0U_MASK	WFF1O_MASK	WFF0O_MASK	OFFU_MASK	BDR_DONE_MASK	Reserved	PWROFF_MASK	PWRON_MASK	REF_STOP_MASK	REF_STRT_MASK	LUT_STRT_MASK	FRM_END_MASK
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
13	IFF2U_MASK	1: IFF2U interrupt is open. 0: IFF2U interrupt is close.	RW
12	IFF1U_MASK	1: IFF1U interrupt is open. 0: IFF1U interrupt is close.	RW
11	IFF0U _MASK	1: IFF0U interrupt is open. 0: IFF0U interrupt is close.	RW
10	WFF1O_MASK	1: WFF0O interrupt is open. 0: WFF1O interrupt is close.	RW
9	WFF0O_MASK	1: WFF0O interrupt is open. 0: WFF1O interrupt is close.	RW
8	OFFU_MASK	1: OFFU interrupt is open. 0: OFFU interrupt is close.	RW
7	BDR_DONE_MASK	1: BDR_DONE interrupt is open. 0 : BDR_DONE interrupt is close.	RW
5	PWROFF_MASK	1: PWROFF interrupt is open. 0: PWROFF interrupt is close.	RW
4	PWRON_MASK	1: PWRON interrupt is open. 0: PWRON interrupt is close.	RW
3	REF_STOP_MASK	1: REF_STOP interrupt is open. 0: REF_STOP interrupt is close.	RW
2	REF_STRT_MASK	1: REF_STRT interrupt is open. 0: REF_STRT interrupt is close.	RW
1	LUT_DONE_MASK	1: LUT_DONE interrupt is open. 0 : LUT_DONE interrupt is close.	RW
0	FRM_END_MASK	1: FRM_END interrupt is open. 0: FRM_END interrupt is close.	RW

9.4.5 EPDC DISPLAY0 Register

	EPDC_DIS0																															BASE + 0x0010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved					SDSP_CAS	SDSP_MODE	GDCLK_MODE	Reserved				GDCLK_POL	GDOE_POL	GDSP_POL	SDCLK_POL	SDOE_POL	SDSP_POL	SDCE_POL	SDLE_POL	Reserved		GDSP_CAS	Reserved								EPD_OMODE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
26	SDSP_CAS	It means output start pulse for Source Drivers cascading from XDIOR or XDIOL depending on data shift direction.	RW
25	SDSP_MODE	It means output start pulse for Source Drivers to start line a line data, or using chip enable for data sampling.	RW
24	GDCLK_MODE	0: GDCLK will be terminated when the last line data output, 1: GDCLK will last another line when outputted the last line data. For TYPE1 EPD, it always should be set to 1; for TYPE2 EPD, set it to 0 for normal mode and set it to 1 for fast mode.	RW
23:22	Reserved	Writing has no effect, read as zero.	R
21		Keep this bit to 0	RW
20		Keep this bit to 0	RW
19	GDCLK_POL	Polarity of Gate Driver clock.	RW
18	GDOE_POL	Polarity of Gate Driver output enable.	RW
17	GDSP_POL	Polarity of Gate Driver start pulse.	RW
16	SDCLK_POL	Source Driver clock sample edge selection. 0: sample data at rising edge 1: sample data at falling edge	RW
15	SDOE_POL	Polarity of Source Driver output enable.	RW
14	SDSP_POL	Polarity of Source Driver start pulse.	RW
13	SDCE_POL	Polarity of Source Drivers chip enable.	RW
12	SDLE_POL	Polarity of Source Driver data latch signal.	RW
11:10	Reserved	Writing has no effect, read as zero.	R
9	GDSP_CAS	It means output start pulse for Gate Drivers cascading from YDIOU or depending on data shift direction.	RW
8:1	Reserved	Writing has no effect, read as zero.	R
0	EPD_OMODE	The Source Driver data bus width. 0: 8-bit	RW

9.4.6 EPDC DISPLAY1 Register

	EPDC_DIS1																															BASE + 0x0014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved	SDDO_REV	PDAT_SWAP	SDCE_REV	SDOS												PDAT								SDCE_STN				SDCE_NUM			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	Reserved	Writing has no effect, read as zero.	R
30	SDDO_REV	Set it to reverse pixels arrangement output to Source Driver. For instance, if SDDO_REV = 0, EPD_OMODE = 0, controller will output 4 pixels in each SDCLK in the format: [P3, P2, P1, P0]. But if SDDO_REV = 1, then the output looks like [P0, P1, P2, P3].	RW
29	PDAT_SWAP	Swap padding data or not, using it with SDRL in following combinations: PDAT_SWAP = 0padding after the display data PDAT_SWAP = 1padding before the display data	RW
28	SDCE_REV	If Source Driver using chip enable, set it will reverse chips enable sequence, using it with SDCE_STN and SDCE_NUM. But if SDCE_REV = 1, outputs will be [SDCE1, SDCE0].	RW
27:16	SDOS	It is available when Source Driver using chip enable. SDOS = (Single Source Driver output size) / (Pixels per Clock)	RW
15:8	PDAT	Source Driver padding data, only available when Source Driver use chip enable. PDAT = (Source Driver Output Size * Number – Line Display Size) / (Pixels per Clock)	RW
7:4	SDCE_STN	Source Driver start number, only available when Source Driver use chip enables.	RW
3:0	SDCE_NUM	Source Driver total number, only available when Source Driver use chip enables.	RW

9.4.7 EPDC SIZE Setting Register

	EPDC_SIZE																BASE + 0x0018															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSIZE																HSIZE															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	VSIZE	The number of each frame in line.	RW
15:0	HSIZE	The number of each line in pixel.	RW

9.4.8 EPDC Virtual Display Area Setting Register

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_VAT																															BASE + 0x0020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VT																HT															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	VT	The period of each frame in lines.	RW
15:0	HT	The period of each line in SDCLK.	RW

9.4.9 EPDC Vertical Display Area Setting Register

	EPDC_DAV																															BASE + 0x0024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDE																VDS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	VDE	The line number at which each frame displays data ends.	RW
15:0	VDS	The line number at which each frame displays data starts.	RW

9.4.10 EPDC Horizontal Display Area Setting Register

	EPDC_DAH																															BASE + 0x0028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HDE																HDS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	HDE	The position at which each line displays data ends.	RW
15:0	HDS	The position at which each line displays data starts.	RW

9.4.11 EPDC Vertical Synchronous Start Pulse Setting

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_VSYN																															BASE + 0x002C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPE																VPS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	VPE	The last line number for Gate Driver generating start pulse.	RW
15:0	VPS	The first line number for Gate Driver generating start pulse.	RW

9.4.12 EPDC Horizontal Synchronous Start Pulse Setting

	EPDC_HSYN																															BASE + 0x0030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HPE																HPS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	HPE	The position at which Source Driver data latch signal de-asserts.	RW
15:0	HPS	The position at which Source Driver data latch signal asserts.	RW

9.4.13 EPDC Gate Driver Clock Setting Register

	EPDC_GDCLK																															BASE + 0x0034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GDCLK_DIS																GDCLK_ENA															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	GDCLK_DIS	The position at which Gate Driver clock signal de-asserts.	RW
15:0	GDCLK_ENA	The position at which Gate Driver clock signal asserts.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.4.14 EPDC Gate Output Enable Setting Register

	EPDC_GDOE																															BASE + 0x0038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GDOE_DIS																GDOE_ENA															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	GDOE_DIS	The position at which Gate Driver output enable signal de-asserts.	RW
15:0	GDOE_ENA	The position at which Gate Driver output enable signal asserts.	RW

9.4.15 EPDC Gate Driver Start Pulse Setting

	EPDC_GDSP																															BASE + 0x003C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GDSP_DIS																GDSP_ENA															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	GDSP_DIS	The position at which Gate Driver start pulse signal de-asserts.	RW
15:0	GDSP_ENA	The position at which Gate Driver start pulse signal asserts.	RW

9.4.16 EPDC Source Driver Output Enable Setting Register

	EPD_SDOE																															BASE + 0x0040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDOE_DIS																SDOE_ENA															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	SDOE_DIS	The position at which Source Driver output enable signal de-asserts.	RW
15:0	SDOE_ENA	The position at which Source Driver output enable signal asserts.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.4.17 EPDC Source Driver Start Pulse Setting Register

	EPDC_SDSP																															BASE + 0x0044
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSP_DIS																SDSP_ENA															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	SDSP_DIS	The position at which Source Driver start pulse signal asserts.	RW
15:0	SDSP_ENA	The position at which Source Driver start pulse signal de-asserts.	RW

9.4.18 EPDC Power Management Registers 0

	EPDC_PMGR0																															BASE + 0x0050
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				PWR_DLY12												Reserved				PWR_DLY01											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	PWR_DLY12	The delay time in line between PWR [1] and PWR [2].	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PWR_DLY01	The delay time in line between PWR [0] and PWR [1].	RW

9.4.19 EPDC Power Management Registers 1

	EPDC_PMGR1																															BASE + 0x0054
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				PWR_DLY34												Reserved				PWR_DLY23											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

27:16		keep these bits to zero.	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PWR_DLY23	The delay time in lines between PWR [2] and PWR [3].	RW

9.4.20 EPDC Power Management Registers 2

	EPDC_PMGR3																															BASE + 0x005C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCOM_IDLE		PWRCOM_POL	UNI_POL	BDR_ENA	Reserved	BDR_IDLE		PWR_POL								Reserved				PWR_DLY67											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	VCOM_IDLE	RW, but useless	RW
29	PWRCOM_POL	RW, but useless	RW
28	UNIPOL	RW, but useless	RW
27	BDR_ENA	Border updating or not.	RW
25:24	BDR_IDLE	Border voltage control signals default value when idle.	RW
23:16	PWR_POL	Polarity of PWR3~0.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PWR_DLY67	RW, but useless	RW

9.4.21 EPDC Power Management Registers 4

	EPDC_PMGR4																															BASE + 0x0060
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								PWR_ENA								VCOMBD_STEP								PWR_VAL							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Writing has no effect, read as zero.	R
23:16	PWR_ENA	These bits enable PWR7~0 individually.	RW
15:8	BD_STEP	BORDER steps	RW
7:0	PWR_VAL	The PWR [x] pin value individually if it is not enabled.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.4.22 EPDC LUT Base address Registers

	EPDC_LUTBF																															
																	BASE + 0x0070															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_LUTBF																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	EPDC_LUTBF	LUT Base address.	RW

9.4.23 EPDC LUT Size Registers

	EPDC_LUTSIZE																															
																	BASE + 0x0074															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_LUTPOS		Reserved																	E	P	D	C	_	L	U	T	S	I	Z	E	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	EPDC_LUTPOS	LUT start position: 00: 0x0 01: 0x800 10: 0x1800	RW
29:13	Reserved	Writing has no effect, read as zero.	R
12:0	EPDC_LUTSIZE	LUT buffer size, count in word.	RW

9.4.24 EPDC Current Buffer Base address Registers

	EPDC_CURBF																															
																	BASE + 0x0080															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_CURBF																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	EPDC_CURBF	Current buffer Base address.	RW

9.4.25 EPDC Current Buffer Size Registers

	EPDC_CURSIZE																																BASE + 0x0084
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_CURSIZE																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	EPDC_CURSIZE	Current buffer size, count in word.	RW

9.4.26 EPDC Working Buffer0 Base address Registers

	EPDC_WORK0BF																																BASE + 0x0090
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_WORK0BF																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	EPDC_WORK0BF	Working buffer0 Base address.	RW

9.4.27 EPDC Working Buffer0 Size Registers

	EPDC_WORK0SIZE																																BASE + 0x0094
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_WORK0SIZE																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	EPDC_WORK0SIZE	Working buffer0 size, count in word.	RW

9.4.28 EPDC Working Buffer1 Base address Registers

	EPDC_WORK1BF																															
																	BASE + 0x0098															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_WORK1BF																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	EPDC_WORK1BF	Working buffer1 Base address.	RW

9.4.29 EPDC Working Buffer0 Size Registers

	EPDC_WORK1SIZE																															
																	BASE + 0x009C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDC_WORK1SIZE																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	EPDC_WORK1SIZE	Working buffer1 size, count in word.	RW

9.4.30 EPDC Border Registers 0~1

	EPDC_VCOM0~ EPDC_VCOM15																															
																	BASE + 0x0100 ~ BASE + 0x0140															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved																BD															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	BD	The or border[1:0] of each frame up to 256-step EPDC_BD0: 16~1 EPDC_BD1: 32~17 EPDC_BD2: 48~33 EPDC_BD3: 64~49 EPDC_BD4: 80~65 EPDC_BD5: 96~81 EPDC_BD6: 112~97 EPDC_BD7: 128~113 EPDC_BD8: 144~129 EPDC_BD9: 160~145 EPDC_BD10: 176~161 EPDC_BD11: 192~177 EPDC_BD12: 212~193 EPDC_BD13: 228~213 EPDC_BD14: 240~229 EPDC_BD15: 256~241	RW

9.4.31 EPDC PRIORITY Registers

	EPDC_PRI																															BASE + 0x0410
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				pri_th2										pri_th1										pri_th0								
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The 3 thresholds cut the output fifo into 4 space,. When the entries remained triggers one of them, the priority level will be altered by hardware. And the 3 thresholds must follow the order: pri_th2 ≥pri_th1 ≥pri_th0.

Priority level

Priority_lv	Space
11	Empty ~ threshold0
10	Threshold0 ~ threshold1
01	Threshold1 ~ threshold2
00	Threshold2 ~ full

Bits	Name	Description	RW
28:20	pri_th2	Threshold2: 0~511.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18:10	pri_th1	Threshold1: 0~511.	RW
8:0	pri_th0	Threshold0: 0~511	RW

9.5 Application Guide

9.5.1 Data Buffer Requirement

Maximum current buffer: 4096x4096x4/8 = 8MB.

Maximum new+old+pixcnt buffer: 4096x4096x16/8 = 32MB.

The 2 buffer and LUT should be double words aligned.

New+old+pixcnt buffer need to be initialized to all zero.

Format of current buffer:

3/4bit

31:28	27:24	23:20	19:16	15:12	11:8	7:4	3:0
p8	p7	p6	p5	p4	p3	p2	p1

2bit

15:14	13:12	11:10	9:8	7:6	5:4	3:2	1:0
p8	p7	p6	p5	p4	p3	p2	p1
31:30	29:28	27:26	25:24	23:22	21:20	19:18	17:16
p16	p15	p14	p13	p12	p11	p10	p9

Format of working buffer:

New+old

31:28	27:24	23:20	19:16	15:12	11:8	7:4	3:0
p3_new	p3_old	p2_new	p2_old	p1_new	p1_old	p0_new	p0_old

Pixcnt

31:24	23:16	15:8	7:0
p3_pixcnt	p2_pixcnt	p1_pixcnt	p0_pixcnt

9.5.2 Power On/Off Sequence

The controller has individual bits to enable and set all power management signals' polarity. This is useful for using high and low active power switches. All delay times are in lines, calculated by, (Delay between PWRn to PWRm) = PWR_DLYnm * (HT * (SDCLK Clock Period)).

Those power management pins delays which are not used should be set to 0. Please refer to vendor's EPD displays panel data sheet to find out the required power on/off sequence for VDNS, VDPS, VDPG, VDNG, and VCOM_L and VCOM_H when using TYPE2 EPD. Confirm EPD controller enabled and EPDC_PMGR0~3 settings correct before you start power on/off sequence by setting the bit EPD_PWRON in EPDC_CTRL.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 9-1 Powers On/Off Sequence

9.5.3 Display Timing Setting

The controller is designed as part of LCD controller. The frequency of pixel clock of the LCD is always double of the Source Driver clock. Typically, the frame frequency is 50 Hz in EPD display. Calculate timing parameters as followings,

Frame Rate = Line Number * Line Period, and Line number is set as VT.

Line Period = HT * (SDCLK Clock Period), SDCLK Clock Period = 2 * Pixel Clock Period.

So, first set pixel clock frequency according vendor's data sheet to get requirement SDCLK and frame rate. The following diagrams illustrate the Source and Gate Drivers' timing parameters for TYPE1 and TYPE2 EPD.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 9-2 Source Drivers Reference Timing for TYPE1 EPD

Figure 9-3 Gate Drivers Reference Timing for TYPE1 EPD

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 9-4 Source Drivers Reference Timing for TYPE2 EPD

Figure 9-5 Gate Drivers Reference Timing for TYPE2 EPD

9.5.4 Auto Stop

Since the EPD display is bi-stable, once a frame is updated, it does not require continuous refresh. Therefore, once a frame update completes, the display pipeline can be stopped to save power. This feature is controlled by the EPDC_CFG [AUTO_STOP] register bit. A frame update is performed when all the waveform steps reach the end and all the current and new pixels are equal. At this point, the EPDC sets the EPDC_STA [REF_STOP] register bits to 1. If the

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

EPDC_CFG [AUTO_STOP] bit is set to 1, the display engine stops running.

The EPDC has DRAM memory bus snooping capability. If the host updates pixels in the frame buffer, the frame buffer tag RAM becomes dirty. The EPDC display restarts to perform an update. If the host changes the EPDC_CURBF register, the EPDC also updates the new frame.

9.5.5 Partial/Parallel Update

1 Partial update: In a regular pixel processing, after completion of one screen update, all the new and current pixels are the same. In the next scanning loop, if some parts of the current pixels are changed, these pixels start to update while the rest of the screen is unchanged.

2 Parallel update: Whenever a pixel value is changed in the frame buffer, it starts to update right away without waiting for others. In Parallel update, the current and new pixels are compared. If the current pixel is the same as the new pixel (no change), the new and old pixels and other data are sent to the LUT to produce output data until the waveform reaches the end. If the current pixel is different than the new pixel (changed), then the pixel FSM will take actions to process it. The Pixel FSM has to reach the end before starting a new waveform for the newly changed current pixel.

9.5.6 Update BD0~15

For TYPE2 panel, before LUT updating and frame refreshing, EPDC_BD0~15 should be filled, [BD_STEP] should be configured, and enable [UNIPOL].

9.5.7 Border Display

1 Disable [UNIPOL] and enable [BDR_ENA].

2 Fill EPDC_D0~15, configure [BD_STEP].

3 Wait EPDC_STA [BDR_IDLE].

4 Set EPDC_CTRL [BDR_STRT].

5 Wait EPDC_STA [BDR_DONE].

9.5.8 Update flow

Auto-stop mode is recommende

AUTO-STOP:

1. set EPDC_CTRL.PWRON 1'b1.
2. Prepare current buffer, working buffer and LUT.
3. Initialize working buffer to all zero.
4. Configure EPDC registers and set EPDC_CFG [AUTO_STOP].
5. Set EPDC_CTRL [EPD_ENA] to 1.
6. Wait EPDC_STA [REF_IDLE] and EPDC_STA [LUT_IDLE].
7. Configure EPDC_LUTSIZE, EPDC_LUTBF, EPDC_CFG[STEP], EPDC_CFG[STEP1], EPDC_CFG [STEP2].
8. Set EPDC_CTRL [LUT_STRT] to load LUT and wait EPDC_STA [LUT_DONE].
9. Configure EPDC_CURSIZE, EPDC_WORKBF, EPDC_WORKSIZE, EPDC_CFG [IBPP].
10. Change EPDC_CURBF to update data.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11. set right value to DDR's relative registers, include EPD_HADDR.haddr and EPD_LADDR.laddr.
12. EPDC will stop when all pixels done.

UNAUTO-STOP:

1 Prepare current buffer, working buffer and LUT.
2 Initialize working buffer to all zero.
3 Configure EPDC registers and clear EPDC_CFG [AUTO_STOP].
4 Set EPDC_CTRL [EPD_ENA] to 1.
5 Wait EPDC_STA [REF_IDLE] and EPDC_STA [LUT_IDLE].
6 Configure EPDC_LUTSIZE, EPDC_LUTBF, EPDC_CFG[STEP], EPDC_CFG[STEP1], EPDC_CFG [STEP2].
7 Set EPDC_CTRL [LUT_STRT] to load LUT and wait EPDC_STA [LUT_DONE].
8 Configure EPDC_CURSIZE, EPDC_WORKBF, EPDC_WORKSIZE, EPDC_CURBF, EPDC_CFG [IBPP].
9 Set EPDC_CTRL [REF_STRT] to update data.
10 Set EPDC_CTRL [REF_STOP] to stop EPDC and wait EPDC_STA [REF_STOP].

Note:

1 In auto-stop mode, EPDC stops automatically, EPDC_CTRL [REF_STOP], EPDC_CTRL [REF_STRT] is unavailable. Changing EPDC_CURBF when EPDC is stopped will start refreshing. Otherwise, EPDC_CTRL [REF_STRT] should be set to start refreshing. EPDC_CTRL [REF_STOP] should be set to stop refreshing.
2 EPDC_CURBF, EPDC_CURSIZE, EPDC_WORKBF, EPDC_WORKSIZE could be configured anytime, but are updated when all pixels are done.
3 EPDC_CFG[IBPP],EPDC_CFG[REF_MODE],EPDC_CFG[STEP],EPDC_LUTBF,EPDC_LUTSIZE should be configured when both EPDC_STA [REF_IDLE] and EPDC_STA [LUT_IDLE] are 1.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10 Image Process Unit

10.1 Overview

IPU (Image process unit) contains Resize and CSC (color space conversion), which is used for image post processing.

10.1.1 Feature

Location: AXI bus

2-stage TLB support

Input format

- Separate frame: YUV /YCbCr (4:2:0, 4:2:2, 4:4:4, 4:1:1), RGB888
- Packaged data: YUV422, ARGB888, RGB565, RGB555, YUV444
- Separate frame in block format: YUV/YCbCr 420

Output data format

- RGB (565, 555, 888, AAA)
- Packaged data YUV422

Color convention coefficient: configurable (CSC enable)

Minimum input image size (pixel): 4x4

Maximum input image size (pixel): 8096x8096

Maximum output image size (pixel)

- Width: up to 4095 (without vertical resizing)
  up to **1280** (with vertical resizing)
- Height: up to 4095

Image resizing

- bi-cube zooming mode
- Up scaling ratios up to 1:31 in fractional steps with 1/32 accuracy
- Down scaling ratios up to 31:1 in fractional steps with 1/32 accuracy

**For more details, refer to Special Instruction.*

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.2 Block

Figure 10-1 The Block about the IPUData flow

10.3 Data flow

10.3.1 Input data

- Separated YUV (or YcbCr/RGB888; the following use YUV for convenience) Frame case: Y, U, V data would be fetched from external memory by DMA burst read operation.
- Packaged YUV422 (or ARGB/ RGB888/ RGB565/ RGB555/YUV444) case: Packaged YUV(ARGB/RGB888/ RGB565/ RGB555/YUV444) data would be fetched from external memory by DMA burst read operation.

10.3.2 Output data

- DMA output to external memory case: The output data format could be RGB (565, 555, 888) or YUV (package422) or ARGB, and the data would be stored to the external memory by DMA burst write operation.
- Flow into LCDC case: The output data format can be RGB or YUV (package), and the transfer would not cross system BUS.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4 Function description

IPU is the module which can resize picture in high performance. It supports almost all normal picture format just like YUV444, YUV422s, YUV420s, YUV 422(package), RGB555, RGB565, RGB888 and YUV TILE. For high quality, IPU uses bi-cube algorithm for scaling. IPU also supports color convention for YUV to RGB. And it can output RGB888, RGB565, RGB555, YUV422(package) to memory or to LCD directly to display.

10.4.1 Source picture and buffer

To the IPU, it needs to know where the picture stored and what format the picture is. People can use Data format register(0x08) to tell IPU what format it is. Besides the format, the IPU has to know the address, where store the picture, through input Y Data Address(0xC), input U Data Address (0x10), input V Data Address(0x14). The address must be 8 Byte align. For one picture stored in memory, it also has some other feature, just like width, height, and stride. The width and height can be configured by the input geometric size register(0x18). And the stride will be configured by the input Y data line stride(0x1C) and input uv data line stride(0x20). The relationship about the three actors is shown in the following figure.

The above figure is just an example. When the picture is RGB555, RGB565, RGB888, YUV422(package), the up figure can show the relation between width, height, and stride. The width and height are just the picture feature. And the stride is the feature of the buffer where the picture is stored. Attention that the width value may change depending on different format, for detail, people can refer to Input Geometric Size (0x18).

And if the input format is YUV420s, YUV422s and YUV444, the buffer will refer to three. People

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

needs to configure them through the input Y buffer address and stride, and the same to U,V buffer address
and stride.

And if the source format is YUVTILE, the stride need focused. The Y stride should be 256 byte align, and Y address also should be 256 byte align. The U(V) stride should be 128 byte align, and U(V) address also should be 128 byte align. The stride must obey the following rules:

Y stride >= (picture width(unit: pixel) * 256)

U(V) stride >= (picture width(unit: pixel) * 128)

10.4.2 Picture resize

IPU uses bi-cube algorithm to finish the scaling. Without complex operation, people can push IPU to do the scaling by just setting the ratio register (resize coefficients 0xB0). The vertical (or horizontal) resize coefficients needs obey the following equations:

VCOEF= (int)(((double)Source Height / (double)Destination Height) * 512)

HCOEF = (int)(((double)Source Width / (double)Destination Width) * 512)

Ofcourse, the VCOEF and HCOEF can be set to any value. The IPU will just select the necessary pixels to do the resizing, which action only depends on the VCOEF and HCOEF. The IPU can support from 31 to 1/32 resizing.

10.4.3 Color convention

The IPU supports absolutely DIY color convention. People can set C0, C1, C2, C3, C4, LUMA_OF and CHROM_OF in free. The color convention will act if the switcher is on, which means the bit of IPU FRAM control(0x0) **IPU_FM_CONTROL.CSC_EN** is set to 1. The following is the color convention equations.

R = C0*(Y –LUMA_OF) + C1*(Cr-CHROM_OF)

G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF)

B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Attention that **IPU_FM_CONTROL.CSC_EN** is set to 0, the LUMA_OF and CHROM_OF are on work the same, and the result will be as following equations.

Y = (Y –LUMA_OF)

Cb =(Cb-CHROM_OF)

Cr = (Cr-CHROM_OF)

10.4.4 Result picture and buffer

After the resizing and color convention, the IPU can output the result in RGB888, RGB555, RGB565, and YUV(package). Just like the source picture and buffer, the result picture also has width, height and buffer stride.

The result width and height can be set through the output geometric size register (0x28). And stride is set through the output data line stride (0x2C). The result buffer address is set through the output frame start address (0x24).

10.4.5 Result picture direct to LCD

The IPU can send the result to lcdc directly without to the memory, which can save much bus consume bandwidth. The way to the lcd is open when the LCDC_SEL bit in ipu frame control register (0x0) is 1. At the start time of each frame, the IPU will handshake with LCD, and send some necessary parameter to LCD, just like IPU OSD control register (0x9C), IPU output geometric size (0x28), data format (0x08) and current field information. And if the handshake fail, the IPU will continue to do the handshake until it success.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4.6 Picture crop

Because the source width and height, result width and height, and the resize ratio can set independently, so the IPU can crop the picture as user's configuration.

And if the result picture's width and height are larger than the scale picture's width and height, the last pixel will be repeatedly filled to the extend space.

10.4.7 Run the IPU

After people set the necessary register in IPU, it need to tell IPU which register has be re-configured.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

People can set the correspond bit in register configure enable control register (0x64) to tell IPU. If one register has no correspond bit in 0x64 register, it will act immediately when it has been changed.

After set the 0x64 register, people just need to write 1 to the IPU trigger register (0x74) to run the IPU.

When value 1 has been write to 0x74, it will record it to record register first. Then IPU check the record register when it is free. If the record register is 1, IPU will start one new frame operation. The IPU will check the 0x64 register at the start time of each frame operation, clear the record register, and the 0x64 register.

10.4.8 IPU work finish and free

IPU gives the IPU status register (0x4) to let people know his inner status. If IPU has finish one frame, and it is on free, the MSTATUS bits in the IPU status register (0x4) will turn to 0. There are 3 case which will make IPU turn to free.

1. IPU finish one frame work and no new frame work again.
2. People stop the IPU and no new frame work again.
3. Source TLB error when source TLB is enabled or destination TLB error occur when destination TLB is enabled, and the TFILL_MOD bit of the ipu global control register (0x7C) has been set to 0.

10.4.9 STOP, reset the IPU

If it needs the IPU stop as soon as possible, people need to stop , not reset IPU. Reset IPU may result some critical consequence. The stop bit in ipu trigger register (0x74) written to 1 will trigger the stop mechanism, and then people just needs wait IPU to turn to free.

The reset to IPU can be used only when IPU is free. All the register in IPU will be set to reset value when reset.

10.4.10 Interrupt

The IPU can trigger the interrupt to the system. People needs enable the interrupt by set the IRQ_EN bit of the ipu global control (0x7C) to 1. There is 3 case which will generate the interrupt.

1. IPU turn to free from the busy.
2. Source TLB error occur if the TLB_IRQ_MSK bit of the ipu global control register (0x7C) is 0.
3. Destination TLB error occur if the TLB_IRQ_MSK bit of the ipu global control register (0x7C) is 0.

10.4.11 Switch between to lcd and to result buffer

If people need switch the ipu output to lcd from to buffer, it just needs following the 1.4.5 chapter and then run the IPU. It is very easy.

And if people need switch the ipu output to buffer from to lcd, it need be careful. Because that the

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

IPU needs send pixels to lcd in one frame without any lack, the ipu will just stop after all pixels of one frame have been to lcd entirely. So the people need to stop IPU to LCD first, which can be acted by writing 1 to the STOP_LCD bit of the IPU trigger register (0x74) or . And then set IPU to output to buffer, and run IPU again.

10.4.12 TLB

10.4.12.1 TLB tab description

For organizing the memory basing the OS, the IPU supports TLB. The tab structure of the TLB is shown in the following:

As the above figure, the TLB tab is constructed by two stage tabs. The first stage tab called primary

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

tab is the entrance of TLB. Any virtual address (vaddr) will need to look up the primary tab first. The virtual address needs using its [31:22] bits as the index to look up the primary tab, and has the corresponding entry PEntryx. If the PEntryx[0] bit is 1, it means the PEntryx is valid, the PEntryx has right secondary tab x. With using itself [21:12] bits as index, virtual address looks up at the secondary tab x, whose start physical address is stored in the PEntryx of the primary tab, to find the right SEntry k.

The bits [31:12] of each SEntry will store the value of the bits [31:12] of the start physical address which is one 4096 byte align continuous memory space. So the vaddr can find its correspond physical address using the SEntry k[31:12] which is called PPN k and vaddr[11:0] by ((PPN k << 12) + vaddr[11:0]). And notice that the bits [11:0] of each SEntry is called TAG. The TAG will indicate the status of the SEntry k.

The flow of one virtual address to look up its physical address is show in the following figure.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4.12.2 IPU TLB mechanism

IPU supports source buffer TLB and destination frame buffer TLB independently. People can control these by set the SPAGE_MAP or DPAGE_MAP in the ipu frame control register (0x0).
At the beginning, people need to tell IPU the start physical address of the primary tab of the TLB by set the input source tlb base address (0x54) and the destination tlb base address (0x58).

If the IPU has been enabled the source TLB or destination TLB function, the correspond address of the source Y(u,v) address and output frame start address need to be set to virtual address. IPU will look up the primary tab and secondary tab to find the PEntryx and SEntry k. It will do the following step to check the entry's validation.

1st: check the validation in the PEntryx, if the PEntryx[0] is 0, IPU will occur the source or destination TLB error.

2nd: check the SEntry k. It the (TAG & check mask) != (golden code & check mask), it will occur TLB error too. (golden code can be set by the **TLB_PARA**. TLB_GCHKC (0xA0), and the check mask can be set the **TLB_PARA**. TLB_MASK (0xA0))

When the TLB error occur, if the TFILL_MOD bit of the ipu global control register (0x7C) has been set to 1, the IPU will suspend all action and wait re-trigger. People can get to know which TLB error occur by read the ipu status (0x4). Besides which TLB error occur, people can know which virtual address makes the TLB error happen by reading **VADDR_STLB_ERR** (0xA4) and **VADDR_DTLB_ERR** (0xA8). By using the **VADDR_STLB_ERR** (0xA4) and **VADDR_DTLB_ERR** (0xA8), people can to refill PEntryx, SEntry k and valid them, and then write the SRC_RF or DST_RF bit to 1 in ipu trigger (0x74) to make the IPU continue to run.

If the TFILL_MOD bit of the ipu global control register (0x7C) has been set to 0, when the TLB error happened, the IPU will turn to stop flow and it will finish the current frame operation immediately. So TLB error should never happen when IPU output to lcdc.

10.5 Registers Descriptions

The physical address base for the address-mapped registers of IPU is **0x13080000.**

10.5.1 IPU Frame Control (0x0)

	IPU_FM_CONTROL																															0x0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													ZOOM_SEL	DFIX_SEL	FIELD_SEL	FIELD_CONF	DISP_SEL	DPAGE_MAP	SPAGE_MAP	LCDC_SEL	SPKG_SEL	Reserved					CSC_EN	VRSZ_EN	HRSZ_EN	Reserved		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:19	Reserved	Writing has no effect, read as zero.	R
18	ZOOM_SEL	IPU rooming mode selector. 1: bi-cube 0: bilinear	RW
17	DFIX_ SEL	Fixed destination address choose. (Valid when LCDC_SEL == 0) 0: not use the fixed address 1: use the fixed address	RW
16	FIELD_SEL *1	Destination field choose. (Valid when FIELD_CONF_EN == 1) 0: top field 1: bottom field	RW
15	FIELD_CONF_EN *1	FIELD_SEL mask. 0: do not change FIELD_SEL 1: configure FIELD_SEL Read as zero.	W
14	DISP_ SEL	Destination display choose. 0: frame display mode 1: field display mode	RW
13	DPAGE_MAP	Destination address page mapping choose. 0: not use the page mapping 1: use the page mapping	RW
12	SPAGE_MAP	Source address page mapping choose. 0: not use the page mapping 1: use the page mapping	RW
11	LCDC_SEL	Output data destination choose. 0: output to external memory 1: output to LCDC FIFO	RW
10	SPKG_SEL	Input data case choose. 0: Separated Frame 1: Packaged Frame	RW
9:5	Reserved	Writing has no effect, read as zero.	R
4	CSC_EN	CSC enable. 1: enable; 0: disable.	RW
3	VRSZ_EN	Vertical Resize enable. 1: enable; 0: disable.	RW
2	HRSZ_EN	Horizontal Resize enable. 1: enable; 0: disable.	RW
1: 0	Reserved	Writing has no effect, read as 1.	R

NOTES:

1 *[1]: The FIELD_SEL will work when the DISP_ SEL is 1, which indicates the IPU is under the field display mode. And the IPU will output the picture from the initial field (top or bottom) to the next field (bottom or top) automatically when the current field has been outputted or

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

stopped. The initial field can be configured by setting the FIELD_SEL to 0 or 1 with FIELD_CONF_EN is 1. The FIELD_CONF_EN is just the trigger that controls the FIELD_SEL's valuation.

10.5.2 IPU Status (0x4)

	IPU_STATUS																															0x4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ID																Reserved						DTLB_ERR	STLB_ERR			MSTATUS		Reserved	SIZE_ERR	Reserved	OUT_END
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

Bits	Name	Description	R/W
31:16	ID	IPU version ID, can not written, read value is 0x8001	R
15:10	Reserved	Writing has no effect, read as zero.	R
9	DTLB_ERR	Destination TLB error.	R
8	STLB_ERR	Source TLB error	R
7:6	Reserved	Writing has no effect, read as zero.	R
5:4	MSTATUS	IPU main status. 00: IPU is free others: IPU is running now.	R
3	Reserved	Writing has no effect, read as zero.	R
2	SIZE_ERR	The size error flag. 1: size error; 0: size ok.	R
1	Reserved	Writing has no effect, read as zero.	R
0	OUT_END *1	Output termination flag. 1: finished; 0: not finished.	R/W

NOTE:

*[1]: If **GLB_CTRL**.IRQ_EN has been set 1, once OUT_END is set value 1 which denotes a frame's post process done, an low level active interrupt request will be issued until corresponding software handler read IPU_STATUS and clean end flag.

When the **IPU_CONTROL**.FM_LCDC_SEL has been set 1, and the IPU has finished one transfer, the LCDC and CPU need to occupy the IPU control. The IPU will monitor the request signal from LCDC and the read signal from the CPU, then it will determine whether dre-configure itself by the CPU if the CPU read first or output the same frame to LCDC again if the LCDC get the control.

10.5.3 Data Format (0x08)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	D_FMT																																0x8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved						RGB_888	RGB_OUT_OFT			OUT_FMT			YUV_PKG_OUT_OFT			Reserved									BLK_SEL			IN_OFT		IN_FMT	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:26	Reserved	Writing has no effect, read as zero.	R
25	RGB_888_OUT_FMT	RGB888 output format indicator. (only used in RGB888 out OUT_FMT == 010) 0: the low 24 bits will be the pixel in a word 1: the high 24 bits will be the pixel in a word	RW
24:22	RGB_OUT_OFT	Output data packaged offset. (only used in RGB out OUT_FMT != 011) [31:0] 000: RGB 001: RBG 010: GBR 011: GRB 100: BRG 101: BGR Others: reserved	RW
21:19	OUT_FMT	Indicates the destination data format. 000: RGB555 001: RGB565 010: RGB888 011: YUV422 package 100: RGBAAA(R(or G or B) is 10 bits wide)	RW
18:16	YUV_PKG_OUT_OFT	Output data packaged offset. (only used in CSC disable case and in the source is YUV422 packaged case (**IPU_CONTROL.** SPKG_SEL == 1)) [31:0] 000: Y1UY0V 001: Y1VY0U. 010: UY1VY0 011: VY1UY0 100: Y0UY1V 101: Y0VY1U 110: UY0VY1 111: VY0UY1	RW
15:7	Reserved	Writing has no effect, read as zero.	R
6:4	BLK_SEL	Indicate the source data format when source is YUV420	RW
3:2	IN_OFT	Input data packaged offset. (when the source is YUV422 packaged case (**IPU_CONTROL.** SPKG_SEL == 1 && IN_FMT == 01)) [31:0] bit format control as following: 00: Y1UY0V 01: Y1VY0U 10: UY1VY0 11: VY1UY0	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1:0	IN_FMT	Indicates the source data format. When **IPU_CONTROL.**SPKG_SEL == 0 00: YUV 4:2:0 01: YUV 4:2:2 10: YUV 4:4:4 11: YUV 4:1:1 When **IPU_CONTROL.**SPKG_SEL == 1 00: RGB555 01: YUV 4:2:2 10: RGB888 or YUV444 11: RGB565	RW

NOTES:

1 When the source frame is packed RGB, the **IPU_CONTROL.** SPKG_SEL must be 1 and IN_FMT must be 10, and when the source frame is packed YUV, the **IPU_CONTROL.** SPKG_SEL must be 1 and IN_FMT must be 01.

2 When the source frame is packed YUV444, the data format about a pixel should be as following:

31 24	23 16	15 8	7 0
0	Y	U	V

10.5.4 Input Y Data Address (0xC)

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

Bits	Name	Description	R/W
31:0	Y_ADDR *1	In separated Frame case, it indicates the source Y (or R) data buffer's start address. In source YUV422 package case, it indicates the start Address of the packaged Frame.	RW

NOTES:

1 When the **IPU_CONTROL**.SPAGE_MAP == 1, the Y_ADDR should be the start virtual address.

2 Y_ADDR should be **word align**. And when the source format is YUVTILE, Y_ADDR should be 256Byte align.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.5 Input U Data Address (0x10)

	U_ADDR																															0x10
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	U_ADDR																														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	U_ADDR *1	The source U data buffer's start address of separated frame case.	RW

NOTES:

1 When the IPU_CONTROL.SPAGE_MAP == 1, the U_ADDR should be the start virtual address.

2 U_ADDR should be **dual-word align**. And when the source format is YUVTILE, U_ADDR should be 128Byte align.

10.5.6 Input V Data Address (0x14)

	V_ADDR																															0x14
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	V_ADDR																														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	V_ADDR	The source V data buffer's start address of separated Frame case.	RW

NOTES:

1 When the IPU_CONTROL.SPAGE_MAP == 1, the V_ADDR should be the start virtual address.

2 V_ADDR should be dual-word align. And when the source format is YUVTILE, V_ADDR should be 128Byte align.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.7 Input Geometric Size (0x18)

	IN_FM_GS																															0x18
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IN_FM_W																IN_FM_H															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	IN_FM_W	The width of the input frame (unit: pixel). Y data width is the same as this value while U/V or Cb/Cr data width should do relatively zoom in according to the source data format. And **in the source package pattern**, this value should be the Y data width also.	RW
150	IN_FM_H	The height of the input frame (unit: pixel). Y data width is same as this value while U/V or Cb/Cr data width should do relatively zoom in according to the source data format.	RW

10.5.8 Input Y Data Line Stride (0x1C)

	Y_STRIDE																															0x1c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								Y_S																						0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:20	Reserved	Writing has no effect, read as zero.	R
19:0	Y_S	The line stride of the source Y data in the external memory of separated Frame case or packaged Frame stride(Unit: byte).	RW

NOTE: Y_S should be dual-world align. And when the source frame format is YUVTILE, the Y_S should be 256Byte align.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.9 Input UV Data Line Stride (0x20)

	UV_STRIDE																																0x20
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	U_S														0		V_S														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	U_S	The line stride of the source U data in the external memory.	RW
15:0	V_S	The line stride of the source V data in the external memory.	RW

NOTE: U_S and V_S should be dual-word align and unit is byte. And when the source frame format is YUVTILE, the U(V)_S should be 128yte align.

10.5.10 Output Frame Start Address (0x24)

	OUT_ ADDR																																0x24
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OUT_ ADDR																														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	OUT_ ADDR *1	The output buffer's start address.	RW

NOTES:

1 *[1]: When the IPU_CONTROL.DPAGE_MAP == 1, the OUT_ADDR should be the start virtual address.

2 it should be dual-word align.

10.5.11 Output Geometric Size (0x28)

	OUT_GS																																0x28
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved	OUT_FM_W															Reserved			OUT_FM_H													
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31	Reserved	Writing has no effect, read as zero.	R
30:16	OUT_FM_W	The width of the output destination frame (unit: byte).	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	OUT_FM_H	The height of the output destination frame (unit: byte).	RW

NOTES:

1 In the package YUV out pattern, the OUT_FM_W should be the **pixel number** in a line.

2 In the **RGB** out pattern, the OUT_FM_W should be the **data space** width in the RAM.

3 In the out package pattern, the OUT_FM_W **must be** even number, and otherwise IPU will not run.

e.g.

Package YUV out: 480x273 (OUT_FM_W: 480 OUT_FM_H: 273)

RGB 888 (or AAA) out: 480x273 (OUT_FM_W: 480*4 OUT_FM_H: 273)

RGB 555 (or 565) out: 480x273 (OUT_FM_W: 480*2 OUT_FM_H: 273)

10.5.12 Output Data Line Stride (0x2C)

	OUT_STRIDE																															0x2c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																OUT_S														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	Reserved	Writing has no effect, read as zero.	R
15:0	OUT_S	The line stride of the destination data buffer in the external memory(Unit: byte).	RW

Note： it should be dual-word align.

10.5.13 CSC C0 Coefficient (0x34)

	CSC_C0_COEF																															0x34
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				C0_COEF											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:12	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11:0	C0_COEF	The C0 coefficient of the YUV/YCbCr to RGB conversion. C0_COEF = [C0 * 1024 + 0.5].	RW
NOTE: R = C0*(Y –LUMA_OF) + C1*(Cr-CHROM_OF) G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF) B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)			

10.5.14 CSC C1 Coefficient (0x38)

	CSC_C1_COEF																															0x38
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				C1_COEF											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:12	Reserved	Writing has no effect, read as zero.	R
11:0	C1_COEF	The C1 coefficient of the YUV/YCbCr to RGB conversion. C1_COEF = [C1 * 1024 + 0.5].	RW
NOTE: R = C0*(Y – LUMA_OF) + C1*(Cr-CHROM_OF) G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF) B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)			

10.5.15 CSC C2 Coefficient (0x3C)

	CSC_C2_COEF																															0x3C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				C2_COEF											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:12	Reserved	Writing has no effect, read as zero.	R
11:0	C2_COEF	The C2 coefficient of the YUV/YCbCr to RGB conversion. C2_COEF = [C2 * 1024 + 0.5].	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE:

R = C0*(Y – LUMA_OF) + C1*(Cr-CHROM_OF)

G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF)

B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)

10.5.16 CSC C3 Coefficient (0x40)

	CSC_C3_COEF																															0x40
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				C3_COEF											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:12	Reserved	Writing has no effect, read as zero.	R
11:0	C3_COEF	The C3 coefficient of the YUV/YCbCr to RGB conversion. C3_COEF = [C3 * 1024 + 0.5].	RW

NOTE:

R = C0*(Y – LUMA_OF) + C1*(Cr-CHROM_OF)

G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF)

B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)

10.5.17 CSC C4 Coefficient (0x44)

	CSC_C4_COEF																															0x44
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				C4_COEF											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:12	Reserved	Writing has no effect, read as zero.	R
11:0	C4_COEF	The C4 coefficient of the YUV/YCbCr to RGB conversion. C4_COEF = [C4 * 1024 + 0.5].	RW

NOTE:

R = C0*(Y – LUMA_OF) + C1*(Cr-CHROM_OF)

G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF) – C3*(Cr-CHROM_OF)

B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.18 CSC Offset Parameter (0x50)

	CSC_OFSET_PARA																															0x50
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								CHROM_OF								Reserved								LUMA_OF							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CHROM_OF	Chroma offset value.	RW
15:8	Reserved	Writing has no effect, read as zero.	R
7:0	LUMA_OF	Luma offset value.	RW

NOTE:

R = C0*(Y – LUMA_OF) + C1*(Cr-CHROM_OF)

G = C0*(Y – LUMA_OF) – C2*(Cb-CHROM_OF)– C3*(Cr-CHROM_OF)

B = C0*(Y – LUMA_OF) + C4*(Cb-CHROM_OF)

10.5.19 Input source TLB base address (0x54)

	SRC_TLB_ADDR																															0x54
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRC_TLB_ADDR																														0	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	SRC_TLB_ ADDR	The SOURCE TLB base address. (This register will act when the IPU_CONTROL.PAGE_MAP==1)	RW

NOTES:

SRC_TLB_ADDR should be dual-word align.

10.5.20 Destination TLB base address (0x58)

	DEST_TLB_ADDR																															0x58
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEST_TLB_ADDR																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:0	DEST_TLB_ ADDR	The destination frame's TLB base address. (This register will work when the IPU_CONTROL.DPAGE_MAP==1)	RW

NOTES:

DEST_TLB_ADDR should be dual-word align.

10.5.21 Register configure enable control (0x64)

	IPU_REG_CTRL																															0x64
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													OSD_RY	OFT_RY	reserved		RCT_RY	CSCC_RY	DSTR_RY	DGS_RY	CSTR_RY	YSTR_RY	IGS_RY	DF_RY	CTRL_RY	PTD_RY	PTS_RY	D_RY	V_RY	U_RY	Y_RY
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

Bits	Name	Description	R/W
31:6	Reserved	Writing has no effect, read as zero.	R
18	OSD_RY	New **OSD_CTRL** ready.	RW
17	OFT_RY	new **FM_XYOFT** ready.	RW
16:15	Reserved	Writing has no effect.	R
14	RCT_RY	**RSZ_COEF_INDEX** ready.	RW
13	CSCC_RY	**CSC_C0(1,2,3)_COEF** and **CSC_OFSET_PARA** ready.	RW
12	DSTR_RY	New destination stride (**OUT_STRIDE**) ready.	RW
11	DGS_RY	New destination geometric size (**OUT_GS**) ready.	RW
10	CSTR_RY	New UV stride (**UV_STRIDE**) ready	RW
9	YSTR_RY	New Y stride (**Y_STRIDE**) ready	RW
8	IGS_RY	New input geometric size (**IN_FM_GS**) ready.	RW
7	DF_RY	New data format (**D_FMT**) ready.	RW
6	CTRL_RY	New frame control (**IPU_FM_CONTROL**) ready	RW
5	PTD_RY	New destination TLB base address ready.	RW
4	PTS_RY	New source TLB base address ready.	RW
3	D_RY	New destination address ready.	RW
2	V_RY	New source V address ready.	RW
1	U_RY	New source U address ready.	RW
0	Y_RY	New source Y address ready.	RW

NOTES:

1 When the xx_RY bit has been set 1, the IPU will use the new corresponding register in the next frame, and use the old address value whose corresponding bit in **IPU_REG_CTRL** is 0.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.22 IPU trigger (0x74)

	IPU_TRIGER																															0x74
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved																										DST_RF	SRT_RF	RESET	STOPLCD	STOP	RUN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

Bits	Name	Description	R/W
31:6	Reserved	Writing has no effect, read as zero.	R
5	DST_RF	Destination TLB refill	W
4	SRC_RF	Source TLB refill	W
3	RESET	reset	W
2	STOP_LCD	Stop ipu to lcd, ipu will not output to lcd next frame	W
1	STOP	Stop ipu, ipu will stop immediately in safe	W
0	RUN	New fame configurable need run.	RW

10.5.23 Output Frame offset on screen (0x78)

	FM_XYOFT																															0x78
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				YOFT												Reserved				XOFT											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	YOFT	Vertical offset	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	XOFT	Horizon offset	RW

10.5.24 IPU global Control(0x7C)

	GLB_CTRL																															0x7C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved														TLB_IRQ_MSK	TFILL_MOD	Reserved						SHARP_LEVE		Reserved							IRQ_EN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:18	Reserved	Writing has no effect, read as zero.	R
17	TLB_IRQ_MSK	TLB error interrupt mask. 1. valid, mask the TLB error. 0. no mask to the TLB error	RW
16	TFILL_MOD	TLB refill mode enable. 0: disable 1:enable	RW
15:10	Reserved	Writing has no effect, read as zero.	R
9:8	SHARP_LEVE	The sharp level selector: 0: most blur 3: highest sharp	RW
7:1	Reserved	Writing has no effect, read as zero.	R
0	IRQ_EN	Interrupt enable 0: disable 1: enable	R/W

10.5.25 Input Y Data Address of next frame (0x84)

	Y_ADDR_N																															0x84
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Y_ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	Y_ADDR_N	In separated Frame case, it indicates the source Y (or R) data buffer's start address of the next frame. In package case, it indicates the start address of the packaged Frame of the next frame.	R

10.5.26 Input U Data Address of next frame (0x88)

	U_ADDR_N																															0x88
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	U_ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:0	U_ADDR_N	The source U (G) data buffer's start address of the next frame in separated frame case.	R

NOTE: When the IPU_CONTROL.SPAGE_MAP == 1, the U_ADDR_N will be the start virtual address.

10.5.27 Input V Data Address of next frame (0x8C)

	V_ADDR_N																															0x8C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	V_ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	V_ADDR_N	The source V (B) data buffer's start address of the next frame in separated frame case.	R

NOTE: When the IPU_CONTROL.SPAGE_MAP == 1, the V_ADDR_N will be the start virtual address.

10.5.28 Output Data Address of next frame (0x90)

	OUT_ ADDR_N																															0x90
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OUT_ ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	OUT_ADDR_N *1	The output buffer's start address.	R

NOTES:

1 *[1]: When the IPU_CONTROL.DPAGE_MAP == 1, the OUT_ADDR_N should be the start virtual address.

2 it should be dual-word align.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.5.29 Source TLB base address of next frame (0x94)

	SRC_TLB_ADDR_N																															0x94
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRC_TLB_ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	SRC_TLB_ADDR_N	The TLB base address about the next source frame's data which will be DMA in.	R

10.5.30 Destination TLB base address of next frame (0x98)

	DEST_TLB_ADDR_N																															0x98
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEST_TLB_ADDR_N																													0		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	DEST_TLB_ADDR_N	The TLB base address about the next frame's data that will be DMA out.	R

10.5.31 IPU OSD Control(0x9C)

	OSD_CTRL																															0x9C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																GALPHA								Reserved					OSD_PM	OSD_MOD	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	Reserved	Writing has no effect, read as zero.	R
15:8	GALPHA	Global alpha	RW
7:3	Reserved	Writing has no effect, read as zero.	R
2	OSD_PM	IPU layer has been pre-multiply	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: without 1: has pre-multiply	
1:0	OSD_MOD	Osd mode selector, please refer to the LCD OSD_MOD	R/W

1.1.1 TLB parameter (0xA0)

	TLB_PARA																															0xA0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				TLB_GCHKC												Reserved				TLB_MASK											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:28	Reserved	Writing has no effect, read as zero.	
27:16	TLB_GCHKC	TLB golden check code	RW
15:12	Reserved	Writing has no effect, read as zero.	
11:0	TLB_MASK	TLB check code mask	RW

Note: For TLB usage, any virtual address will trigger the TLB error when the H2D read the SEntry through the PEntry. If the (SEntry[11:0] & TLB_MASK) != (TLB_GCHKC & TLB_MASK) or the VLD in the PEntry is not 1, the TLB error will occur. People can read the TLB error in the **GLB_STATUS.**

10.5.32 Virtual address for source TLB error (0xA4)

	VADDR_STLB_ERR																															0xA4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VADDR_STLB_ERR																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	VADDR_STLB _ERR	The virtual address which make the source TLB error occur.	R

10.5.33 Virtual address for destination TLB error (0xA8)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	VADDR_DTLB_ERR																															0xA8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VADDR_DTLB_ERR																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:0	VADDR_DTLB _ERR	The virtual address which make the destination TLB error occur.	R

10.5.34 Resize Coefficients (0xB0)

	RSZ_COEF																															0xB0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			HCOEF																VCOEF													
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:30	Reserved	Writing has no effect, read as zero.	R
29:16	HCOEF	Horizontal coef	RW
15:14	Reserved	Writing has no effect, read as zero.	R
13:0	VCOEF	Vertical coef	RW

NOTE: The VCOEF equals the (int)(srcH/dstH)*512, and the HCOEF equals the (int)(srcW/dstW)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.6 IPU Operation Flow

10.6.1 Data out to frame buffer

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.6.2 Data out to lcdc

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.6.3 TLB refill

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.7 Special Instruction

10.7.1 Resizing size feature

Input size (W x H)		Output size (W x H)	
Min	4x4	Disable vertical scale	Min: 4x4
			Max: 4095x4095
Max	8096x8096	Enable vertical scale	Min: 4x4
			Max: 1280x4095

10.7.2 Color convention feature

Source format	Output format	Parameter configure (necessary)
RGB	RGB	IPU_CONTROL.CSC_EN =0 IPU_CONTROL. SPKG_SEL = 0 or 1
		D_FMT. IN_FMT
		D_FMT.OUT_FMT = 2'b00, 2'b01, 2'b10
YUV	RGB	IPU_CONTROL.CSC_EN =1
		IPU_CONTROL. SPKG_SEL
		D_FMT. IN_FMT
		D_FMT. IN_OFT ***(IPU_CONTROL. SPKG_SEL == 1)***
		D_FMT.OUT_FMT = 2'b00, 2'b01, 2'b10
		D_FMT.RGB_OUT_OFT.
		CSC_C0 (1,2,3,4)_COEF, CSC_OFSET_PARA
YUV	YUV (package)	IPU_CONTROL.CSC_EN =0
		IPU_CONTROL. SPKG_SEL
		D_FMT. IN_FMT
		D_FMT. IN_OFT ***(IPU_CONTROL. SPKG_SEL == 1)***
		D_FMT.OUT_FMT = 2'b11

10.7.3 YUV/YCbCr to RGB CSC Equations

Input data	Matrix	CSC_COEF
	R = C0*(Y – X0) + C1*(V-128)	CSC_C0_COEF = 0x400
	G = C`0*(Y – X0) – C2*(U-128) – C3*(V-128)	CSC_C1_COEF= 0x59C
YUV	B = C0*(Y – X0) + C4*(U-128)	CSC_C2_COEF = 0x161
	X0: 0	CSC_C3_COEF = 0x2DC
	C0: 1	CSC_C4_COEF = 0x718

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	C1: 1.4026 C2: 0.3444 C3: 0.7144 C4: 1.7730	
YCbCr	R = C0*(Y – X0) + C1*(Cr-128) G = C0*(Y – X0) – C2*(Cb-128) – C3*(Cr-128) B = C0*(Y – X0) + C4*(Cb-128) X0: 16 C0: 1.164 C1: 1.596 C2: 0.391 C3: 0.813 C4: 2.018	CSC_C0_COEF = 0x4A8 CSC_C1_COEF = 0x662 CSC_C2_COEF = 0x191 CSC_C3_COEF = 0x341 CSC_C4_COEF = 0x811

10.7.4 Output data package format (RGB order)

Format	Package
RGBAAA	
RGB888	 **Or**
RGB555	
RGB565	
NOTE: All R/G/B data are little-endian type; all the empty bits in the above figure are filled with 0.	

10.7.5 Input data package format (RGB order)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Format	Package
RGBPACK	

10.7.6 Source Data storing format in external memory (separated YUV Frame)

Example: YUV420 118x80 frame

NOTES:

1 Every line's start address should be word aligned.

2 All pixel data should be stored as little-endian format.

3 Destination data (RGB) storing format in external memory is similar with above figure, but RGB555 and RGB565 frame's every line start address can be half-word aligned. (RGB888 frame still need word aligned)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11 MIPI_CSI Interface

11.1 Overview

The CSI-2 Host Controller is a digital core that implements all protocol functions defined in the MIPI CSI-2 Specification, providing an interface between the System and the MIPI D-PHY, allowing the communication with a MIPI CSI-2 compliant Camera Sensor.

11.1.1 Features

- Support i2c interface to configure.
- Support up to 2 D-PHY Rx Data Lanes.
- Compliant with MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2).
- Dynamically configurable multi-lane merging.
- Long and Short packet decoding.
- Timing accurate signaling of Frame and Line synchronization packets.
- 32-bit Image Data.
- Support all primary and secondary data formats:
  - RGB, YUV and RAW color space definitions
  - From 24-bit down to 6-bit per pixel
  - Generic or user-defined byte-based data types.
- Error detection and correction.

11.2 Register Descriptions

11.2.1 Register Mapping

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RW - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

? - value unknown after reset

The base address of MIPI_CSI is 0x13013000.

Table 11-1 MIPI_CSI Registers Description

Name	Description	RW	Reset Value	Offset Address	Width
VERSION	Version of the CSI-2 host controller	R	CSI_VERSION_ID	0x000	32
N_LANES	Number of active data lanes	R/W	CSI_N_LANES	0x004	2
PHY_SHUTDOWNZ	PHY shutdown control	R/W	0	0x008	1
DPHY_RSTZ	DPHY reset control	R/W	0	0x00c	1
CSI2_RESETN	CSI-2 controller reset	R/W	0	0x010	1
PHY_STATE	General settings for all blocks	R	0	0x014	12
DATA_IDS_1	List of Data Ids for which IDI reports line boundary matching errors (CSI2_HOST_N_DATA_IDS > 0)	R/W	0	0x018	32
DATA_IDS_2	List of Data Ids for which IDI reports line boundary matching errors (CSI2_HOST_N_DATA_IDS > 1)	R/W	0	0x01c	32
ERR1	Error state register 1	R	0	0x020	29
ERR2	Error state register 2	R	0	0x024	24
MASK1	Masks for errors 1	R/W	0	0x028	29
MASK2	Masks for errors 2	R/W	0	0x02c	24
PHY_TST_CRTL0	D-PHY Test interface control 0	R/W	0	0x030	2
PHY_TST_CRTL1	D-PHY Test interface control 1	R/W	0	0x034	17

11.2.2 VERSION

- Name: Controller Version Identification
- Address Offset:0x000
- Size:32 bits
- Value after Reset: CSI_VERSION_ID
- Access: Read

	VERSION																															BASE+0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION																															
RST																																

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:0	VERSION	Version of the CSI-2 Host Controller	R

11.2.3 N_LANES

- Address Offset:0x004
- Size:2 bits
- Value after Reset:CSI_N_LANES
- Access:Read/Write

	N_LANES	BASE+0x004	
Bit		1	0
		N_LANES	
RST			

Bits	Name	Description	R/W
1:0	N_LANES	Number of Active Data Lanes. • 00: 1 Data Lane (Lane 0) • 01: 2 Data Lanes (Lane 0, and 1) • 10: 3 Data Lanes (Lane 0,1 and 2) • 11: 4 Data Lanes (All) Can only be updated when the PHY lane is in stopstate	R/W

11.2.4 PHY_SHUTDOWNZ

- Name: Phy shutdown control
- Address Offset:0x008
- Size:1 bit
- Value after Reset:0
- Access: Read/Write

	PHY_SHUTDOWNZ	BASE+0x008
Bit		0
		PHY_SHUTDOWNZ
RST		0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
0	PHY_SHUTDOWNZ	Shutdown input. This line is used to place the complete macro in power down. All analog blocks are in power down mode and digital logic is cleared. Active Low.	R/W

11.2.5 DPHY_RSTZ

- Name: Phy reset control
- Address Offset:0x00C
- Size:1 bit
- Value after Reset:0
- Access: Read/Write

	DPHY_RSTZ	BASE+0x00C
Bit		0
		DPHY_RSTZ
RST		0

Bits	Name	Description	R/W
0	DPHY_RSTZ	DPHY reset output. Active Low.	R/W

11.2.6 CSI2_RESETN

- Name:CSI-2 controller reset
- Address Offset:0x010
- Size:1 bit
- Value after Reset:0
- Access: Read/Write

	CSI2_RESETN	BASE+0x010
Bit		0
		CSI_RESETN
RST		0

Bits	Name	Description	R/W
0	CSI_RESETN	CSI-2 controller reset output. Active Low.	R/W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.2.7 PHY_STATE

- Name: General settings for all blocks
- Address Offset:0x014
- Size:12 bits
- Value after Reset:0
- Access: Read (with the exception of bit 11 which is Read/Write)

	PHY_STATE																				BASE+0x014											
Bit																					11	10	9	8	7	6	5	4	3	2	1	0
																					bypass_2ecc_tst	phy_stopstateclk	phy_rxclkactivehs	phy_rxclkactiivehs	phy_stopstatedata_3	phy_stopstatedata_2	phy_stopstatedata_1	phy_stopstatedata_0	phy_rxulpsesc_3	phy_rxulpsesc_2	phy_rxulpsesc_1	phy_rxulpsesc_0
RST																					0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
0	phy_rxulpsesc_0	Lane module 0 has entered the Ultra Low Power mode	R
1	phy_rxulpsesc_1	Lane module 1 has entered the Ultra Low Power mode	R
2	phy_rxulpsesc_2	Lane module 2 has entered the Ultra Low Power mode	R
3	phy_rxulpsesc_3	Lane module 3 has entered the Ultra Low Power mode	R
4	phy_stopstatedata_0	Data Lane 0 in Stop state	R
5	phy_stopstatedata_1	Data Lane 1 in Stop state	R
6	phy_stopstatedata_2	Data Lane 2 in Stop state	R
7	phy_stopstatedata_3	Data Lane 3 in Stop state	R
8	phy_rxclkactivehs	Indicates that the clock lane is actively receiving a DDR clock	R
9	phy_rxulpsclknot	Active Low. This signal indicates that the Clock Lane module has entered the Ultra Low Power state	R
10	phy_stopstateclk	Clock Lane in Stop state	R
11	bypass_2ecc_tst	Payload Bypass test mode for double ECC errors	R/W

11.2.8 DATA_IDS_1

- Address Offset:0x018
- Size:32 bits
- Value after Reset:0
- Access: Read/Write

	DATA_IDS_1																												BASE+0x018			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	di3_vc		di3_dt						di2_vc		di2_dt						di1_vc		di1_dt						di0_vc		di0_dt					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
5:0	di0_dt	Data ID 0 Data Type	R/W
7:6	di0_vc	Data ID 0 Virtual channel	R/W
13:8	di1_dt	Data ID 1 Data Type	R/W
15:14	di1_vc	Data ID 1 Virtual channel	R/W
21:16	di2_dt	Data ID 2 Data Type	R/W
23:22	di2_vc	Data ID 2 Virtual channel	R/W
29:24	di3_dt	Data ID 3 Data Type	R/W
31:30	di3_vc	Data ID 3 Virtual channel	R/W

11.2.9 DATA_IDS_2

- Address Offset:0x01C
- Size:32 bits
- Value after Reset:0
- Access: Read/Write

	DATA_IDS_2																															BASE+0x01C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	di7_vc		di7_dt						di6_vc		di6_dt						di5_vc		di5_dt						di4_vc		di4_dt					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
5:0	di4_dt	Data ID 4 Data Type	R/W
7:6	di4_vc	Data ID 4 Virtual channel	R/W
13:8	di5_dt	Data ID 5 Data Type	R/W
15:14	di5_vc	Data ID 5 Virtual channel	R/W
21:16	di6_dt	Data ID 6 Data Type	R/W
23:22	di6_vc	Data ID 6 Virtual channel	R/W
29:24	di7_dt	Data ID 7 Data Type	R/W
31:30	di7_vc	Data ID 7 Virtual channel	R/W

11.2.10 ERR1

- Name: Error state register 1
- Address Offset:0x020
- Size:29 bits
- Value after Reset:0
- Access: Read

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ERR1																															BASE+0x020
Bit				28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				err_ecc_double	vc3_err_crc	vc2_err_crc	vc1_err_crc	vc0_err_crc	err_l_seq_di3	err_l_seq_di2	err_l_seq_di1	err_l_seq_di0	err_l_bndry_match_di3	err_l_bndry_match_di2	err_l_bndry_match_di1	err_l_bndry_match_di0	err_frame_data_vc3	err_frame_data_vc2	err_frame_data_vc1	err_frame_data_vc0	err_f_seq_vc3	err_f_seq_vc2	err_f_seq_vc1	err_f_seq_vc0	err_f_bndry_match_vc3	err_f_bndry_match_vc2	err_f_bndry_match_vc1	err_f_bndry_match_vc0	phy_errsotsynchs_3	phy_errsotsynchs_2	phy_errsotsynchs_1	phy_errsotsynchs_0
RST				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
0	phy_errsotsynchs_0	Start of Transmission Error on data lane 0 (no synchronization achieved)	R
1	phy_errsotsynchs_1	Start of Transmission Error on data lane 1 (no synchronization achieved)	R
2	phy_errsotsynchs_2	Start of Transmission Error on data lane 2 (no synchronization achieved)	R
3	phy_errsotsynchs_3	Start of Transmission Error on data lane 3 (no synchronization achieved)	R
4	err_f_bndry_match_vc0	Error matching Frame Start with Frame End for Virtual Channel 0	R
5	err_f_bndry_match_vc1	Error matching Frame Start with Frame End for Virtual Channel 1	R
6	err_f_bndry_match_vc2	Error matching Frame Start with Frame End for Virtual Channel 2	R
7	err_f_bndry_match_vc3	Error matching Frame Start with Frame End for Virtual Channel 3	R
8	err_f_seq_vc0	Incorrect Frame Sequence detected in Virtual Channel 0	R
9	err_f_seq_vc1	Incorrect Frame Sequence detected in Virtual Channel 1	R
10	err_f_seq_vc2	Incorrect Frame Sequence detected in Virtual Channel 2	R
11	err_f_seq_vc3	Incorrect Frame Sequence detected in Virtual Channel 3	R
12	err_frame_data_vc0	Last received frame, in Virtual Channel 0, had at least one CRC error	R
13	err_frame_data_vc1	Last received frame, in Virtual Channel 1, had at least one CRC error	R
14	err_frame_data_vc2	Last received frame, in Virtual Channel 2, had at least one CRC error	R
15	err_frame_data_vc3	Last received frame, in Virtual Channel 3, had at least one CRC error	R
16	err_l_bndry_match_di0	Error matching Line Start with Line End for vc0 and dt	R
17	err_l_bndry_match_di1	Error matching Line Start with Line End for vc1 and dt	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18	err_l_bndry_match_di2	Error matching Line Start with Line End for vc2 and dt	R
19	err_l_bndry_match_di3	Error matching Line Start with Line End for vc3 and dt	R
20	err_l_seq_di0	Error in the sequence of lines for vc0 and dt0	R
21	err_l_seq_di1	Error in the sequence of lines for vc0 and dt1	R
22	err_l_seq_di2	Error in the sequence of lines for vc0 and dt2	R
23	err_l_seq_di3	Error in the sequence of lines for vc0 and dt3	R
24	vc0_err_crc	Checksum Error detected on Virtual Channel 0	R
25	vc1_err_crc	Checksum Error detected on Virtual Channel 1	R
26	vc2_err_crc	Checksum Error detected on Virtual Channel 2	R
27	vc3_err_crc	Checksum Error detected on Virtual Channel 3	R
28	err_ecc_double	Header ECC contains 2 errors. Unrecoverable	R

11.2.11 ERR2

- Name: Error state register 2
- Address Offset:0x024
- Size:24 bits
- Value after Reset:0
- Access: Read

	ERR2																															BASE+0x024
Bit									23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
									err_l_seq_di7	err_l_seq_di6	err_l_seq_di5	err_l_seq_di4	err_l_bndry_match_di7	err_l_bndry_match_di6	err_l_bndry_match_di5	err_l_bndry_mastch_di4	err_id_vc3	err_id_vc2	err_id_vc1	err_id_vc0	Vc3_err_ecc_corrected	Vc2_err_ecc_corrected	Vc1_err_ecc_corrected	vc0_err_ecc_corrected	phy_errsoths_3	phy_errsoths_2	phy_errsoths_1	phy_errsoths_0	phy_ eresc 3	phy_ eresc _2	phy_ eresc _1	phy_ eresc_0
RST									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
0	phy_eresc_0	Escape Entry Error (ULPM) on data lane 0	R
1	phy_ eresc _1	Escape Entry Error (ULPM) on data lane 1	R
2	phy_ eresc _2	Escape Entry Error (ULPM) on data lane 2	R
3	phy_ eresc _3	Escape Entry Error (ULPM) on data lane 3	R
4	phy_errsoths_0	Start of Transmission Error on data lane 0(synchronization can still be achieved)	R
5	phy_errsoths_1	Start of Transmission Error on data lane 1(synchronization can still be achieved)	R
6	phy_errsoths_2	Start of Transmission Error on data lane 2(synchronization can still be achieved)	R
7	phy_errsoths_3	Start of Transmission Error on data lane 3(synchronization	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		can still be achieved)	
8	vc0_err_ecc_corrected	Header error detected and corrected on Virtual Channel 0	R
9	vc1_err_ecc_corrected	Header error detected and corrected on Virtual Channel 1	R
10	vc2_err_ecc_corrected	Header error detected and corrected on Virtual Channel 2	R
11	vc3_err_ecc_corrected	Header error detected and corrected on Virtual Channel 3	R
12	err_id_vc0	Unrecognized or unimplemented data type detected in Virtual Channel 0	R
13	err_id_vc1	Unrecognized or unimplemented data type detected in Virtual Channel 1	R
14	err_id_vc2	Unrecognized or unimplemented data type detected in Virtual Channel 2	R
15	err_id_vc3	Unrecognized or unimplemented data type detected in Virtual Channel 3	R
16	err_l_bndry_match_di4	Error matching Line Start with Line End for vc4 and dt4	R
17	err_l_bndry_match_di5	Error matching Line Start with Line End for vc5 and dt5	R
18	err_l_bndry_match_di6	Error matching Line Start with Line End for vc6 and dt6	R
19	err_l_bndry_match_di7	Error matching Line Start with Line End for vc7 and dt7	R
20	err_l_seq_di4	Error in the sequence of lines for vc0 and dt4	R
21	err_l_seq_di5	Error in the sequence of lines for vc0 and dt5	R
22	err_l_seq_di6	Error in the sequence of lines for vc0 and dt6	R
23	err_l_seq_di7	Error in the sequence of lines for vc0 and dt7	R

11.2.12 MASK1

- Name: Masks for errors 1
- Address Offset:0x028
- Size:29 bits
- Value after Reset:0
- Access: Read/Write

	MASK1																															BASE+0x028
Bit				28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				mask_err_ecc_double	mask_vc3_err_crc	mask_vc2_err_crc	mask_vc1_err_crc	mask_vc0_err_crc	mask_err_l_seq_di3	mask_err_l_seq_di2	mask_err_l_seq_di1	mask_err_l_seq_di0	mask_err_l_bndry_match_di3	mask_err_l_bndry_match_di2	mask_err_l_bndry_match_di1	mask_err_l_bndry_match_di0	mask_err_frame_data_vc3	mask_err_frame_data_vc2	mask_err_frame_data_vc1	mask_err_frame_data_vc0	mask_err_f_seq_vc3	mask_err_f_seq_vc2	mask_err_f_seq_vc1	mask_err_f_seq_vc0	mask_err_f_bndry_match_vc3	mask_err_f_bndry_match_vc2	mask_err_f_bndry_match_vc1	mask_err_f_bndry_match_vc0	mask_phy_errsotsynchs_3	mask_phy_errsotsynchs_2	mask_phy_errsotsynchs_1	mask_phy_errsotsynchs_0
RST				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
0	mask_phy_errsotsynchs_0	Mask for phy_errsotsynchs_0	R/W
1	mask_phy_errsotsynchs_1	Mask for phy_errsotsynchs_1	R/W
2	mask_phy_errsotsynchs_2	Mask for phy_errsotsynchs_2	R/W
3	mask_phy_errsotsynchs_3	Mask for phy_errsotsynchs_3	R/W
4	mask_err_f_bndry_match_vc0	Mask for err_f_bndry_match_vc0	R/W
5	mask_err_f_bndry_match_vc1	Mask for err_f_bndry_match_vc1	R/W
6	mask_err_f_bndry_match_vc2	Mask for err_f_bndry_match_vc2	R/W
7	mask_err_f_bndry_match_vc3	Mask for err_f_bndry_match_vc3	R/W
8	mask_err_f_seq_vc0	Mask for err_f_seq_vc0	R/W
9	mask_err_f_seq_vc1	Mask for err_f_seq_vc1	R/W
10	mask_err_f_seq_vc2	Mask for err_f_seq_vc2	R/W
11	mask_err_f_seq_vc3	Mask for err_f_seq_vc3	R/W
12	mask_err_frame_data_vc0	Mask for err_frame_data_vc0	R/W
13	mask_err_frame_data_vc1	Mask for err_frame_data_vc1	R/W
14	mask_err_frame_data_vc2	Mask for err_frame_data_vc2	R/W
15	mask_err_frame_data_vc3	Mask for err_frame_data_vc3	R/W
16	mask_err_l_bndry_match_di0	Mask for err_l_bndry_match_di0	R/W
17	mask_err_l_bndry_match_di1	Mask for err_l_bndry_match_di1	R/W
18	mask_err_l_bndry_match_di2	Mask for err_l_bndry_match_di2	R/W
19	mask_err_l_bndry_match_di3	Mask for err_l_bndry_match_di3	R/W
20	mask_err_l_seq_di0	Mask for err_l_seq_di0	R/W
21	mask_err_l_seq_di1	Mask for err_l_seq_di1	R/W
22	mask_err_l_seq_di2	Mask for err_l_seq_di2	R/W
23	mask_err_l_seq_di3	Mask for err_l_seq_di3	R/W
24	mask_vc0_err_crc	Mask for vc0_err_crc	R/W
25	mask_vc1_err_crc	Mask for vc1_err_crc	R/W
26	mask_vc2_err_crc	Mask for vc2_err_crc	R/W
27	mask_vc3_err_crc	Mask for vc3_err_crc	R/W
28	mask_err_ecc_double	Mask for err_ecc_double	R/W

11.2.13 MASK2

- Name: Masks for errors 2
- Address Offset:0x02C
- Size:24 bits
- Value after Reset:0
- Access: Read/Write

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MASK2																															BASE+0x02C
Bit									23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
									mask_err_l_seq_di7	mask_err_l_seq_di6	mask_err_l_seq_di5	mask_err_l_seq_di4	mask_err_l_bndry_match_di7	mask_err_l_bndry_match_di6	mask_err_l_bndry_match_di5	mask_err_l_bndry_mastch_di4	mask_err_id_vc3	mask_err_id_vc2	mask_err_id_vc1	mask_err_id_vc0	mask_vc3_err_ecc_corrected	mask_vc2_err_ecc_corrected	mask_vc1_err_ecc_corrected	mask_vc0_err_ecc_corrected	mask_phy_errsoths_3	mask_phy_errsoths_2	mask_phy_errsoths_1	mask_phy_errsoths_0	mask_phy_ erresc 3	mask_phy_ erresc _2	mask_phy_ erresc _1	mask_phy_erresc_0
RST									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
0	mask_phy_erresc_0	Mask for phy_erresc_0	R/W
1	mask_phy_ erresc _1	Mask for phy_erresc_1	R/W
2	mask_phy_ erresc _2	Mask for phy_erresc_2	R/W
3	mask_phy_ erresc _3	Mask for phy_erresc_3	R/W
4	mask_phy_errsoths_0	Mask for phy_errsoths_0	R/W
5	mask_phy_errsoths_1	Mask for phy_errsoths_1	R/W
6	mask_phy_errsoths_2	Mask for phy_errsoths_2	R/W
7	mask_phy_errsoths_3	Mask for phy_errsoths_3	R/W
8	mask_vc0_err_ecc_corrected	Mask for vc0_err_ecc_corrected	R/W
9	mask_vc1_err_ecc_corrected	Mask for vc1_err_ecc_corrected	R/W
10	mask_vc2_err_ecc_corrected	Mask for vc2_err_ecc_corrected	R/W
11	mask_vc3_err_ecc_corrected	Mask for vc3_err_ecc_corrected	R/W
12	mask_err_id_vc0	Mask for err_id_vc0	R/W
13	mask_err_id_vc1	Mask for err_id_vc1	R/W
14	mask_err_id_vc2	Mask for err_id_vc2	R/W
15	mask_err_id_vc3	Mask for err_id_vc3	R/W
16	mask_err_l_bndry_match_di4	Mask for err_l_bndry_match_di4	R/W
17	mask_err_l_bndry_match_di5	Mask for err_l_bndry_match_di5	R/W
18	mask_err_l_bndry_match_di6	Mask for err_l_bndry_match_di6	R/W
19	mask_err_l_bndry_match_di7	Mask for err_l_bndry_match_di7	R/W
20	mask_err_l_seq_di4	Mask for err_l_seq_di4	R/W
21	mask_err_l_seq_di5	Mask for err_l_seq_di5	R/W
22	mask_err_l_seq_di6	Mask for err_l_seq_di6	R/W
23	mask_err_l_seq_di7	Mask for err_l_seq_di7	R/W

11.2.14 PHY_TST_CRTL0

- Name: D-PHY Test interface control 0
- Address Offset:0x030

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Size:2 bits
- Value after Reset:0
- Access: Read/Write

	PHY_TST_CRTL0																														BASE+0x030	
Bit																															1	0
																															phy_testclk	phy_testclr
RST																															0	0

Bits	Name	Description	R/W
0	phy_testclr	PHY test interface clear. Used when active performs vendor specific interface initialization (Active High).	R/W
1	phy_testclk	PHY test interface strobe signal. Used to clock TESTDIN bus into the DPHY. In conjunction with TESTEN signal controls the operation selection.	R/W

11.2.15 PHY_TST_CRTL1

- Name: D-PHY Test interface control 1
- Address Offset:0x034
- Size:17 bits
- Value after Reset:0
- Access: Read/Write

	PHY_TST_CRTL1																															BASE+0x034	
Bit																	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	phy_testen	phy_testdout								phy_testdin							
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
0:7	phy_testdin	PHY test interface input 8-bit data bus for internal register programming and test functionalities access	R/W
8:15	phy_testdout	PHY output 8-bit data bus for read-back and internal probing functionalities.	R
16	phy_testen	PHY test interface operation selector: • 1: Configures address write operation on the falling edge of TESTCLK • 0: Configures a data write operation on the rising edge of TESTCLK	R/W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.3 Operation

The following sections describe the MIPI_CSI operations of configuration.

The Initialization steps are as follows:

1. De-assert CSI-2 presetn signal (global reset).
2. Set number of Lanes(register N_LANES).
3. De-assert PHY shutdown (register PHY_SHUTDOWNZ)
4. De-assert PHY reset (register PHY_RSTZ)
5. De-assert CSI reset (register CSI2-RESETN)
6. Wait for 200us, read the PHY status register (PHY_STATE) to confirm that all data and clock lanes of the D-PHY are in Stop State.
7. Configure the camera sensor to start transmitting a clock on the D-PHY clock lane.
8. Read the PHY status register(PHY_STATE) to confirm that the D-PHY is receiving a clock on the D-PHY clock lane.
9. Meting the steps above, configure the camera sensor to start transmitting packages.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12 AC97/I2S/SPDIF Controller

12.1 Overview

This chapter describes the AIC (AC'97 and I^2S Controller) included in this processor.

The AIC supports the Audio Codec '97 Component Specification 2.3 for AC-link format and I2S or IIS (for inter-IC sound), a protocol defined by Philips Semiconductor. Both normal I2S and the MSB-justified I2S formats are supported by AIC.

AIC consists of buffers, status registers, control registers, serializers, and counters for transferring digitized audio between the processor's system memory and an internal I2S CODEC, an external AC97 or I2S CODEC. AIC can record digitized audio by storing the samples in system memory. For playback of digitized audio or production of synthesized audio, the AIC retrieves digitized audio samples from system memory and sends them to a CODEC through the serial connection with AC-link or I2S formats. The internal or external digital-to-analog converter in the CODEC then converts the audio samples into an analog audio waveform. The audio sample data can be stored to and retrieved from system memory either by the DMA controller or by programmed I/O.

The AC-link is a synchronous, fixed-rate serial bus interface for transferring CODEC register control and status information in addition to digital audio. Where both normal I2S and MSB-justified-I2S work with a variety of clock rates, which can be obtained either by dividing the PLL clock by two programmable dividers or from an external clock source.

For I2S systems that support the L3 control bus protocol, additional pins are required to control the external CODEC. CODECs that use an L3 control bus require 3 signals: L3_CLK, L3_DATA, and L3_MODE for writing bytes into the L3 bus register. The AIC supports the L3 bus protocol via software control of the general-purpose I/O (GPIO) pins. The AIC does not provide hardware control for the L3 bus protocol.

To control the internal CODEC, internal CODEC Spec can be referenced.

SPDIF (Sony/Philips Digital Interconnect Format) is a digital audio interface. The transmission medium can be either electrical or optical. Supports IEC60958 two-channel PCM audio and IEC61937 multi-channel compressed audio (Dolby Digital, DTS, etc.).

This chapter describes the programming model for the AIC. The information in this chapter requires an understanding of the AC'97 specification, Revision 2.3.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.1.1 Block Diagram

Figure 12-1 AIC Block Diagram

The O_BIT_CLK and O_SYNC ports are only used by inter CODEC.

12.1.2 Features

AIC support following AC97/I2S/SPDIF features:

AC-link (AC97) features

- Up to 20 bit audio sample data sizes supported
- DMA transfer mode supported
- Stop serial clock supported
- Programmable Interrupt function supported
- Support mono PCM data to stereo PCM data expansion on audio play back
- Support endian switch on 16-bits normal audio samples play back
- Support variable sample rate in AC-link format
- Multiple channel output and double rated supported for AC-link format
- Power Down Mode and two Wake-Up modes Supported for AC-link format

I2S features

- 8, 16, 18, 20 and 24 bit audio sample data sizes supported, 16 bits packed sample data is supported
- DMA transfer mode supported

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Stop serial clock supported
- Programmable Interrupt function supported
- Support mono PCM data to stereo PCM data expansion on audio play back
- Support endian switch on 16-bits normal audio samples play back
- Internal programmable or external serial clock and optional system clock supported for I2S or MSB-Justified format
- Internal I2S CODEC supported
- Two FIFOs for transmit and receive respectively

SPDIF features

- 8, 16, 18, 20 and 24 bit audio sample data sizes supported
- DMA transfer mode supported
- Stop serial clock supported
- Programmable Interrupt function supported
- Support IEC60958 two-channel PCM audio
- Support IEC61937 multi-channel compressed audio
- Support consumer mode and only support transmitter mode
- Profession mode is not supported
- The User data bit is ‘0’as it is not supported in the chip
- Support sampling frequency from 32kHz to 192kHz

12.1.3 Interface Diagram

Figure 12-2 Interface to an External AC’97 CODEC Diagram

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-3 Interface to an External Master Mode I2S/MSB-Justified CODEC Diagram (Share Clock Mode)

Figure 12-4 Interface to an External Slave Mode I2S/MSB-Justified CODEC Diagram (Share Clock Mode)

Figure 12-5 Interface to a HDMI Transmitter via SPDIF Diagram

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-6 Interface to an internal Master Mode I2S CODEC Diagram

Please refer to the related CODEC specification for the details.

12.1.4 Signal Descriptions

There are all 5 pins used to connect between AIC and an external audio CODEC device. If an internal CODEC is used, these pins are not needed. Please refer to Chip Spec. They are listed and described in following table.

Table 12-1 AIC Pins Description

Function Name	PIN Name	I/O	Description
RESET# **SYS_CLK**	SCLK_RSTN	O	RESET#: AC-link format, active-low CODEC reset. SYS_CLK: I2S/MSB-Justified formats, supply system clock to CODEC.
SDATA_IN	SDATI	I	Serial audio input data from CODEC.
BIT_CLK	BCLK	I I/O	12.288 MHz bit-rate clock input for AC-link, and sample rate dependent bit-rate clock input/output for I2S/MSB-Jistified.
SYNC	SYNC	O	48-kHz frame indicator and synchronizer for AC-link format.
		I/O	Indicates the left- or right-channel for I2S/MSB-Justified format.
SDATA_OUT	SDATO	O	Serial audio output data to CODEC / I2S line 0 / SPDIF output.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.1.4.1 RESET# / SYS_CLK Pin

RESET# is AC97 active-low CODEC reset, which outputs to CODEC. The CODEC's registers are reset when this RESET# is asserted. This pin is useful only in AC-link format. If AIC is disabled, it retains the high.

SYS_CLK outputs the system clock to CODEC. This pin is useful only in I2S/MSB-justified format. It generates a frequency between approximately 2.048 MHz and 24.576 MHz by dividing down the PLL clock with a programmable divisor. This frequency can be 256, 384, 512 and etc. times of the audio sampling frequency. Or it can be set to a wanted frequency. If AIC is disabled, it retains the high.

12.1.4.2 BIT_CLK Pin

BIT_CLK is the serial data bit rate clock, at which AC97/I2S data moves between the CODEC and the processor. One bit of the serial data is transmitted or received each BIT_CLK period. It is fixed to 12.288 MHz in AC-link format, which inputs from the CODEC. In I2S and MSB-justified format it inputs from the CODEC in slave mode and outputs to CODEC in master mode. In the master mode, the clock is generated internally that is 64 times the sampling frequency. Table 12-7 lists the available sampling frequencies based on an internal clock source. If AIC is disabled, AICFR.AUSEL and AICFR.BCKD determine the direction. And it retains the low if it is output and the state is undefined if it is input.

The IBIT_CLK is for the SDATA_IN signal on division CLOCK function.

12.1.4.3 SYNC Pin

In AC-link format, SYNC provides frame synchronization, fixed to 48kHz, by specifying beginning of an audio sample frame and outputs to CODEC. In I2S/MSB-Justified formats, SYNC is used to indicate left- or right-channel sample data and toggled in sample rate frequency. It outputs to CODEC in master mode and inputs from CODEC in slave mode. If AIC is disabled, AICFR.AUSEL and AICFR.BCKD determine the direction. And it retains the low if it is output and the state is undefined if it is input.

The ISYNC is for the SDATA_IN signal on division CLOCK function.

12.1.4.4 SDATA_OUT Pin

SDATA_OUT is AIC output data pin, which outputs AC97/I2S serial audio data, SPDIF serial data or data of AC97 CODEC register control to an external audio CODEC device.
If in multi channels mode, it outputs the first two channels serial data.
If AIC is disabled, it retains the low.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.1.4.5 SDATA_IN Pin

SDATA_IN is AIC inputs data pin, which inputs serial audio data or data of AC97 CODEC register status from an external audio CODEC device. If AIC is disabled, its state is undefined.

12.2 Register Descriptions

AIC software interface includes 13 registers and 1 FIFO data port. They are mapped in IO memory address space so that program can access them to control the operation of AIC and the outside CODEC.

Table 12-2 AIC Registers Description

Name	Description	RW	Reset value	Address	Size
AICFR	AIC Configuration Register	RW	0x07100000	0x10020000	32
AICCR	AIC Common Control Register	RW	0x01240000	0x10020004	32
ACCR1	AIC AC-link Control Register 1	RW	0x00000000	0x10020008	32
ACCR2	AIC AC-link Control Register 2	RW	0x00000000	0x1002000C	32
I2SCR	AIC I2S/MSB-justified Control Register	RW	0x00000000	0x10020010	32
AICSR	AIC FIFO Status Register	RW	0x00000008	0x10020014	32
ACSR	AIC AC-link Status Register	RW	0x00000000	0x10020018	32
I2SSR	AIC I2S/MSB-justified Status Register	RW	0x00000000	0x1002001C	32
ACCAR	AIC AC97 CODEC Command Address Register	RW	0x00000000	0x10020020	32
ACCDR	AIC AC97 CODEC Command Data Register	RW	0x00000000	0x10020024	32
ACSAR	AIC AC97 CODEC Status Address Register	R	0x00000000	0x10020028	32
ACSDR	AIC AC97 CODEC Status Data Register	R	0x00000000	0x1002002C	32
I2SDIV	AIC I2S/MSB-justified Clock Divider Register	RW	0x00000003	0x10020030	32
AICDR	AIC FIFO Data Port Register	RW	0x????????	0x10020034	32
SPENA	SPDIF Enable Register	RW	0x00	0x10020080	8
SPCTRL	SPDIF Control Register	RW	0x0003	0x10020084	16
SPSTATE	SPDIF Status Register	RW	0x0000	0x10020088	16
SPCFG1	SPDIF Configure 1 Register	RW	0x00000000	0x1002008C	32
SPCFG2	SPDIF Configure 2 Register	RW	0x00000000	0x10020090	32
SPFIFO	SPDIF FIFO Register	W	0x????????	0x10020094	32
CKCFG	Clock Configure for the embedded CODEC to AIC	RW	0x00000000 0x00000002	0x100200A0	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

RGADW	Address, data in and write command for accessing to internal registers of embedded CODEC	RW	0x00000000	0x100200A4	32
RGDATA	The read out data and interrupt request status of Internal registers data in the embedded CODEC	R	0x00000000	0x100200A8	32

1 AICFR is used to control FIFO threshold, AC-link or I2S/MSB-justified selection, AIC reset, master/slave selection, and AIC enable.
2 AICCR is used to control DMA mode, FIFO flush, interrupt enable, internal loop-back, play back and recording enable. It also controls sample size and signed/unsigned data transfer.
3 ACCR1 is used to reflect/control valid incoming/outgoing slots of AC97.
4 ACCR2 is used to control interrupt enable, output/input sample size, and alternative control of RESET#, SYNC and SDATA_OUT pins in AC-link.
5 I2SCR is used to control BIT_CLK stop, audio sample size, I2S or MSB-justified selection in I2S/MSB-justified.
6 AICSR is used to reflect FIFOs status.
7 ACSR is used to reflect the status of the connected external CODEC in AC-link.
8 I2SSR is used to reflect AIC status in I2S/MSB-justified.
9 ACCAR and ACCDR are used to hold address and data for AC-link CODEC register read/write.
10 ACSAR and ACSDR are used to receive AC-link CODEC registers address and data.
11 I2SDIV is used to set clock divider for BIT_CLK generating in I2S/MSB-justified format.
12 AICDR is act as data input/output port to/from transmit/receive FIFO when write/read.
13 CKCFG, RGADW and RGDATA are used to access internal CODEC, please refer to CODEC Spec.

12.2.1 AIC Configuration Register (AICFR)

AICFR contains bits to control FIFO threshold, AC-link or I2S/MSB-justified selection, AIC reset, master/slave selection, and AIC enable.

	AICFR																															0x10020000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				RFTH				Reserved			TFTH					Reserved			MSB	SYSCKD	IBCKD	ISYNCD	DMODE	CDC_SLAVE	LSMP	ICDC	AUSEL	RST	BCKD	SYNCD	ENB
RST	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td>27:24</td><td>RFTH</td><td>Receive FIFO threshold for interrupt or DMA request. The RFTH valid value is 0 ~ 15.
This value represents a threshold value of (RFTH + 1) * 2. When the sample number in receive FIFO, indicated by AICSR.RFL, is great than or equal to the threshold value, AICSR.RFS is set. Larger RFTH value provides lower DMA/interrupt request frequency but have more risk to involve receive FIFO overflow. The optimum value is system dependent.</td><td>RW</td></tr>
<tr><td>23:21</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>20:16</td><td>TFTH</td><td>Transmit FIFO threshold for interrupt or DMA request. The TFTH valid value 0 ~ 31.
This value represents a threshold value of TFTH * 2. When the sample number in transmit FIFO, indicated by AICSR.TFL, is less than or equal to the threshold value, AICSR.TFS is set. Smaller TFTH value provides lower DMA/interrupt request frequency but have more risk to involve transmit FIFO underflow. The optimum value is system dependent.</td><td>RW</td></tr>
<tr><td>15:11</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>12</td><td>MSB</td><td>The big-endian format enable. When this bit is '1', the replay data will use the big-endian format. The sign bit at the bit 31. Else the replay data is little-endian format.</td><td>RW</td></tr>
<tr><td>11</td><td>SYSCKD</td><td>SYSCLK Direction. This bit specifies input/output direction of SYSCLK.
<table>
<tr><th>BCKD</th><th>BIT_CLK Direction</th></tr>
<tr><td>0</td><td>SYSCLK is generated internally and driven out to the CODEC.</td></tr>
<tr><td>1</td><td>SYSCLK is input from an external source.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>10</td><td>IBCKD</td><td>Keep this value to 0 in normal use.</td><td>RW</td></tr>
<tr><td>9</td><td>ISYNCD</td><td>Keep this value to 0 in normal use.</td><td>RW</td></tr>
<tr><td>8</td><td>DMODE</td><td>Keep this value to 0 in normal use.</td><td>RW</td></tr>
<tr><td>7</td><td>CDC_
SLAVE</td><td>Codec I2S interface slave mode. The mode is used for eliminated the frequency skew when internal codec transfer with other audio interface, such as PCM.</td><td>R</td></tr>
<tr><td>6</td><td>LSMP</td><td>Select between play last sample or play ZERO sample in TX FIFO underflow. ZERO sample means sample value is zero. This bit is better be changed while audio replay is stopped.
<table>
<tr><th>LSMP</th><th>CODEC used</th></tr>
<tr><td>0</td><td>Play ZERO sample when TX FIFO underflow.</td></tr>
<tr><td>1</td><td>Play last sample when TX FIFO underflow.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>5</td><td>ICDC</td><td>Internal CODEC used. Select between internal or external CODEC.
<table>
<tr><th>ICDC</th><th>CODEC used</th></tr>
<tr><td>0</td><td>External CODEC.</td></tr>
<tr><td>1</td><td>Internal CODEC.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>4</td><td>AUSEL</td><td>Audio Unit Select. Select between AC-link and I2S/MSB-justified. Change this bit in case of BIT_CLK is stopped (I2SCR.STPBK = 1).</td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		AUSEL / **Selected** 0 / **Select AC-link format.** 1 / **Select I2S/MSB-justified format.**	
3	RST	Reset AIC. Write 1 to this bit reset AIC registers and FIFO except AICFR and I2SDIV register. Writing 0 to this bit has no effect and this bit is always reading 0.	W
2	BCKD	BIT_CLK Direction. This bit specifies input/output direction of BIT_CLK. It is only valid in I2S/MSB-justified format. When AC-link format is selected, BIT_CLK is always input and this bit is ignored. Change this bit in case of BIT_CLK is stopped (I2SCR.STPBK = 1). **BCKD** / **BIT_CLK Direction** 0 / BIT_CLK is input from an external source. 1 / BIT_CLK is generated internally and driven out to the CODEC.	RW
1	SYNCD	SYNC Direction. This bit specifies input/output direction of SYNC in I2S/MSB-justified format. When AC-link format is selected, SYNC is always output and this bit is ignored. Change this bit in case of BIT_CLK is stopped (I2SCR.STPBK = 1). **SYNCD** / **SYNC Direction** 0 / SYNC is input from an external source. 1 / SYNC is generated internally and driven out to the CODEC.	RW
0	ENB	Enable AIC function. This bit is used to enable or disable the AIC function. **ENB** / **Description** 0 / **Disable AIC Controller.** 1 / **Enable AIC Controller.**	RW

The BCKD bit (bit 2) and SYNCD bit (bit 1) configure the mode of I2S/MSB-justified interface. This is compliant with I2S specification.

BCKD	SYNCD	Description
0 (input)	**0 (input)**	**AIC roles the slave of I2S/MSB-justified interface.**
	1 (output)	**AIC roles the master with external serial clock source of I2S/MSB-justified interface.**
1 (output)	**0 (input)**	**Reserved.**
	1 (output)	**AIC roles the master of I2S/MSB-justified interface.**

12.2.2 AIC Common Control Register (AICCR)

AICCR contains bits to control DMA mode, FIFO flush, interrupt enable, internal loop-back, play back and recording enable. It also controls sample size and signed/unsigned data transfer.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	AICCR																															0x10020004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		Reserved	PACK16	Reserved	CHANNEL			Reserved		OSS			ISS			RDMS	TDMS	Reserved		M2S	ENDSW	ASVTSU	TFLUSH	RFLUSH	EROR	ETUR	ERFS	ETFS	ENLBF	ERPL	EREC
RST	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29	Reserved	Keep this value to 0 in normal use.	R
28	PACK16	Output Sample data 16bit packed mode select. This bit reflects that one word contains two sample data or only one sample data with LSB align. The packed mode is only support 16bit sample size. **PACK16** / **Sample Size** 0: Unpacked data mode. One word only contains one 16bit sample data aligned LSB. 1: Packed data mode. One word contains two 16 bit sample data.	RW
27	Reserved	Writing has no effect, read as zero.	R
26:24	CHANNEL	Output Channel Number Select. These bits reflect output data channels from AIC to device. The data supported are: 1(mono), 2(stereo), 4, 6 and 8 channels. The sample data is LSB-justified in memory/register. **CHANNEL** / **Sample Size** 0x0: 1 channel, mono mode. 0x1: 2 channels, stereo mode. 0x2 ~ 0x7: Reserved.	RW
23:22	Reserved	Writing has no effect, read as zero.	R
21:19	OSS	Output Sample Size. These bits reflect output sample data size from memory or register. The data sizes supported are: 8, 16, 18, 20 and 24 bits. The sample data is LSB-justified in memory/register. **OSS** / **Sample Size** 0x0: 8 bit. 0x1: 16 bit. 0x2: 18 bit. 0x3: 20 bit. 0x4: 24 bit. 0x5~0x7: Reserved.	RW
18:16	ISS	Input Sample Size. These bits reflect input sample data size to memory	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>or register. The data sizes supported are: 8, 16, 18, 20 and 24 bits. The sample data is LSB-justified in memory/register.
<table>
<tr><th>ISS</th><th>Sample Size</th></tr>
<tr><td>0x0</td><td>8 bit.</td></tr>
<tr><td>0x1</td><td>16 bit.</td></tr>
<tr><td>0x2</td><td>18 bit.</td></tr>
<tr><td>0x3</td><td>20 bit.</td></tr>
<tr><td>0x4</td><td>24 bit.</td></tr>
<tr><td>0x5~0x7</td><td>Reserved.</td></tr>
</table></td><td></td></tr>
<tr><td>15</td><td>RDMS</td><td>Receive DMA enable. This bit is used to enable or disable the DMA during receiving audio data.
<table>
<tr><th>RDMS</th><th>Receive DMA</th></tr>
<tr><td>0</td><td>Disabled.</td></tr>
<tr><td>1</td><td>Enabled.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>14</td><td>TDMS</td><td>Transmit DMA enable. This bit is used to enable or disable the DMA during transmit audio data.
<table>
<tr><th>TDMS</th><th>Transmit DMA</th></tr>
<tr><td>0</td><td>Disabled.</td></tr>
<tr><td>1</td><td>Enabled.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>13:12</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>11</td><td>M2S</td><td>Mono To Stereo. This bit control whether to do mono to stereo sample expansion in play back. When this bit is set, every outgoing sample data in the steam plays in both left and right channels. This bit should only be set in 2 channels configuration. It takes effective immediately when the bit is changed. Change this before replay started.
<table>
<tr><th>M2S</th><th>Description</th></tr>
<tr><td>0</td><td>No mono to stereo expansion.</td></tr>
<tr><td>1</td><td>Do mono to stereo expansion.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>10</td><td>ENDSW</td><td>Endian Switch. This bit control endian change on outgoing 16-bits size audio sample by swapping high and low bytes in the sample data.
<table>
<tr><th>ENDSW</th><th>Description</th></tr>
<tr><td>0</td><td>No change on outgoing sample data.</td></tr>
<tr><td>1</td><td>Swap high and low byte for outgoing 16-bits size sample data.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>9</td><td>ASVTSU</td><td>Audio Sample Value Transfer between Signed and Unsigned data format. This bit is used to control the signed ←→unsigned data transfer. If it is 1, the incoming and outgoing audio sample data will be transferred by toggle its most significant bit.
<table>
<tr><th>ASVTSU</th><th>Description</th></tr>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td><table><tr><td>0</td><td>No audio sample value signed ←→unsigned transfer.</td></tr><tr><td>1</td><td>Do audio sample value signed ←→unsigned transfer.</td></tr></table></td><td></td></tr>
<tr><td>8</td><td>TFLUSH</td><td>Transmit FIFO Flush. Write 1 to this bit flush transmit FIFOs to empty. Writing 0 to this bit has no effect and this bit is always reading 0.</td><td>W</td></tr>
<tr><td>7</td><td>RFLUSH</td><td>Receive FIFO Flush. Write 1 to this bit flush receive FIFOs to empty. Writing 0 to this bit has no effect and this bit is always reading 0.</td><td>W</td></tr>
<tr><td>6</td><td>EROR</td><td>Enable ROR Interrupt. This bit is used to control the ROR interrupt enable or disable.<table><tr><th>EROR</th><th>ROR Interrupt</th></tr><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>5</td><td>ETUR</td><td>Enable TUR Interrupt. This bit is used to control the TUR interrupt enable or disable.<table><tr><th>ETUR</th><th>TUR Interrupt</th></tr><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>4</td><td>ERFS</td><td>Enable RFS Interrupt. This bit is used to control the RFS interrupt enable or disable.<table><tr><th>ERFS</th><th>RFS Interrupt</th></tr><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>3</td><td>ETFS</td><td>Enable TFS Interrupt. This bit is used to control the TFS interrupt enable or disable.<table><tr><th>ETFS</th><th>TFS Interrupt</th></tr><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>2</td><td>ENLBF</td><td>Enable AIC Loop Back Function. This bit is used to enable or disable the internal loop back function of AIC, which is used for test only. When the AIC loop back function is enabled, normal audio replay/record functions are disabled.<table><tr><th>ENLBF</th><th>Description</th></tr><tr><td>0</td><td>AIC Loop Back Function is Disabled.</td></tr><tr><td>1</td><td>AIC Loop Back Function is Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>1</td><td>ERPL</td><td>Enable Playing Back function. This bit is used to disable or enable the audio sample data transmitting.<table><tr><th>ERPL</th><th>Description</th></tr><tr><td>0</td><td>AIC Playing Back Function is Disabled.</td></tr><tr><td>1</td><td>AIC Playing Back Function is Enabled.</td></tr></table></td><td>RW</td></tr>
<tr><td>0</td><td>EREC</td><td>Enable Recording Function. This bit is used to disable or enable the audio sample data receiving.</td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			EREC	Description		
			0	**AIC Recording Function is Disabled.**		
			1	**AIC Recording Function is Enabled.**		

12.2.3 AIC AC-link Control Register 1 (ACCR1)

ACCR1 contains bits to reflect/control valid incoming/outgoing slots of AC97. It is used only in AC-link format.

	ACCR1																															0x10020008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved						RS										Reserved						XS									
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:26	Reserved	Writing has no effect, read as zero.	R
25:16	RS	Receive Valid Slots. These bits are used to indicate which incoming slots are valid. Slot 3 is mapped to bit 16 or RS[0], slot 4 to bit 17 or RS[1] and so on. When write to this field, a bit 1 means we expect a PCM data in the corresponding slot, a bit 0 means the corresponding slot PCM data will be discarded. When read from this field, a bit 1 means we receive an expected valid PCM data in the corresponding slot. This field should be written before record started. **RS[n] Value** — **Description** 0 — **Slot n+3 is invalid.** 1 — **Slot n+3 has valid PCM data.**	RW
15:10	Reserved	Writing has no effect, read as zero.	R
9:0	XS	Transmit Valid Slots. These bits making up slots map to the valid bits in the AC'97 tag (slot 0 on SDATA_OUT) and indicate which outgoing slots have valid PCM data. Bit 0 or XS[0] maps to slot 3, bit 1 or XS[1] to slot 4 and so on. Setting the corresponding bit indicates to AIC to take an audio sample from transmit FIFO to fill the respective slot. And it indicates to the CODEC that valid PCM data will be in the respective slot. The number of valid bits will designate how many words will be pulled out of the FIFO per audio frame. This field should be written before record and replay started. **XS[n] Value** — **Description** 0 — **Slot n+3 is invalid.** 1 — **Slot n+3 has valid PCM data.**	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.2.4 AIC AC-link Control Register 2 (ACCR2)

ACCR2 contains bits to control interrupt enable, output/input sample size, and alternative control of RESET#, SYNC and SDATA_OUT pins in AC-link. It is valid only in AC-link format.

	ACCR2																															0x1002000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													ERSTO	ESADR	ECADT	Reserved												SO	SR	SS	SA
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:19	Reserved	Writing has no effect, read as zero.	R
18	ERSTO	Enable RSTO Interrupt. This bit is used to control the RSTO interrupt enable or disable. **ERSTO** — **RSTO Interrupt** 0 — **Disabled.** 1 — **Enabled.**	RW
17	ESADR	Enable SADR Interrupt. This bit is used to control the SADR interrupt enable or disable. **ESADR** — **SADR Interrupt** 0 — **Disabled.** 1 — **Enabled.**	RW
16	ECADT	Enable CADT Interrupt. This bit is used to control the CADT interrupt enable or disable. **ECADT** — **CADT Interrupt** 0 — **Disabled.** 1 — **Enabled.**	RW
15:4	Reserved	Writing has no effect, read as zero.	R
3	SO	SDATA_OUT output value. When SA is 1, this bit controls SDATA_OUT pin voltage level, 0 for low, 1 for high; otherwise, it is ignored.	RW
2	SR	RESET# pin level. When AC-link is selected, this bit is used to drive the RESET# pin. **SR** — **RESET# Pin Voltage Level** 0 — High. 1 — Low.	RW
1	SS	SYNC value. When this bit is read, it returns the actual value of SYNC. When SA is 1, write value controls SYNC pin value. When SA is 0, write to it is ignored.	RW
0	SA	SYNC and SDATA_OUT Alternation. This bit is used to determine the driven signal of SYNC and SDATA_OUT. When SA is 0, SYNC and	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		SDATA_OUT being driven AIC function logic; otherwise, SYNC is controlled by the SS and SDATA_OUT is controlled by the SO. The true table of SYNC is described in following.				

SA	SS	Description
0	0	When read, indicated SYNC is 0.
		When write, not effect.
	1	When read, indicated SYNC is 1.
		When write, not effect.
1	0	When read, indicated SYNC is 0.
		When write, SYNC is driven to 0.
	1	When read, indicated SYNC is 1.
		When write, SYNC is driven to 1.

12.2.5 AIC I2S/MSB-justified Control Register (I2SCR)

I2SCR contains bits to control BIT_CLK stop, audio sample size, I2S or MSB-justified selection in I2S/MSB-justified. It is valid only in I2S/MSB-justified format.

	I2SCR																																0x10020010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved														RFIRST	SWLH	Reserved		ISTPBK	STPBK	Reserved							ESCLK	Reserved			AMSL
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:18	Reserved	Writing has no effect, read as zero.	R
17	RFIRST	Send R channel first in stereo mode. This bit should only be set in 2 channels configuration. The frame is LR like or RL like. It takes effective immediately when the bit is changed. Change this before replay started. **RFIRST** — **Description** 0 — **Send L channel first. (LR)** 1 — **Send R channel first. (RL)**	RW
16	SWLH	Switch LR channel in 16bit-packed stereo mode. This bit control whether the low address data is L or R. This bit should only be set in 2 channels configuration and 16bit-packed mode. That means it can only valid with packed mode (PACK16 =1) and 2 channels (CHANNEL=0x1). It takes effective immediately when the bit is changed. Change this before replay started. **SWLH** — **Description**	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			0	**16 bit LSB and 16bit MSB is not switched.**	
			1	**16 bit LSB and 16bit MSB is switched.**	
15:12	Reserved	Writing has no effect, read as zero.			R
13	ISTPBK	Keep this value to 0 in normal use.			RW
12	STPBK	Stop BIT_CLK. It is used to stop the BIT_CLK in I2S/MSB-justified format. When AC-link is selected, all of its operations are ignored. **STPBK** / **Description** 0 / **BIT_CLK is not stopped.** 1 / **BIT_CLK is stopped.** Please set this bit to 1 to stop BIT_CLK when change AICFR.AUSEL and AICFR.BCKD.			RW
11:5	Reserved	Writing has no effect, read as zero.			R
4	ESCLK	Enable SYSCLK output. When this bit is 1, the SYSCLK outputs to chip outside is enabled. Else, the clock is disabled.			RW
3:1	Reserved	Writing has no effect, read as zero.			R
0	AMSL	Specify Alternate Mode (I2S or MSB-Justified) Operation. **AMSL** / **Description** 0 / Select I2S Operation Mode. 1 / Select MSB-Justified Operation Mode.			RW

12.2.6 AIC Controller FIFO Status Register (AICSR)

AICSR contains bits to reflect FIFOs status. Most of the bits are read-only except two, which can be written a 0.

	AICSR																															0x10020014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		RFL						Reserved										TFL						Reserved	ROR	TUR	RFS	TFS	Reserved		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:24	RFL	Receive FIFO Level. The bits indicate the amount of valid PCM data in Receive FIFO. **RFL Value** / **Description**	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td><table><tr><td>0x00 ~ 0x20</td><td>RFL valid PCM data in receive FIFO.</td></tr><tr><td>0x21 ~ 0x3F</td><td>Reserved.</td></tr></table></td><td></td></tr>
<tr><td>23:14</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>13:8</td><td>TFL</td><td>Transmit FIFO Level. The bits indicate the amount of valid PCM data in Transmit FIFO.<table><tr><th>TFL Value</th><th>Description</th></tr><tr><td>0x00 ~ 0x3F</td><td>TFL valid PCM data in transmit FIFO.</td></tr></table></td><td>R</td></tr>
<tr><td>7</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>6</td><td>ROR</td><td>Receive FIFO Over Run. This bit indicates that receive FIFO has or has not experienced an overrun.<table><tr><th>ROR</th><th>Description</th></tr><tr><td rowspan="2">0</td><td>When read, indicates over-run has not been found.</td></tr><tr><td>When write, clear itself.</td></tr><tr><td rowspan="2">1</td><td>When read, indicates data has even been written to full receive FIFO.</td></tr><tr><td>When write, not effects.</td></tr></table></td><td>RW</td></tr>
<tr><td>5</td><td>TUR</td><td>Transmit FIFO Under Run. This bit indicates that transmit FIFO has or has not experienced an under-run.<table><tr><th>TUR</th><th>Description</th></tr><tr><td rowspan="2">0</td><td>When read, indicates under-run has not been found.</td></tr><tr><td>When write, clear itself.</td></tr><tr><td rowspan="2">1</td><td>When read, indicates data has even been read from empty transmit FIFO.</td></tr><tr><td>When write, not effects.</td></tr></table></td><td>RW</td></tr>
<tr><td>4</td><td>RFS</td><td>Receive FIFO Service Request. This bit indicates that receive FIFO level is or not below receive FIFO threshold, which is controlled by AICFR.RFTH. When RFS is 1, it may trigger interrupt or DMA request depends on the interrupt enable and DMA setting.<table><tr><th>RFS</th><th>Description</th></tr><tr><td>0</td><td>Receive FIFO level below RFL threshold.</td></tr><tr><td>1</td><td>Receive FIFO level at or above RFL threshold.</td></tr></table></td><td>R</td></tr>
<tr><td>3</td><td>TFS</td><td>Transmit FIFO Service Request. This bit indicates that transmit FIFO level is below Transmit FIFO threshold, which is controlled by AICFR.TFTH. When TFS is 1, it may trigger interrupt or DMA request depends on the interrupt enable and DMA setting.<table><tr><th>TFS</th><th>Description</th></tr><tr><td>0</td><td>Transmit FIFO level exceeds TFL threshold.</td></tr><tr><td>1</td><td>Transmit FIFO level at or below TFL threshold.</td></tr></table></td><td>R</td></tr>
<tr><td>2:0</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.2.7 AIC AC-link Status Register (ACSR)

ACSR contains bits to reflect the status of the connected external CODEC in AC-link format. Bits in this register are read-only in general, except some of them can be written a 0.

	ACSR																															0x10020018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved										SLTERR	CRDY	CLPM	RSTO	SADR	CADT	Reserved															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:22	Reserved	Writing has no effect, read as zero.	R
21	SLTERR	Hardware detects a Slot Error. This bit indicates an error in SLOTREQ bits on incoming data from external CODEC is detected. The error can be: (1) find 1 in a SLOTREQ bit, which corresponding to an inactive slot; (2) all active slots should be request in the same time by SLOTREQ, but an exception is found. All errors are accumulated to ACSR.SLTERR by hardware until software clears it. Software writes 0 clear this bit and write 1 has no effect.	RW
20	CRDY	External CODEC Ready. This bit is derived from the CODEC Ready bit of Slot 0 in SDATA_IN, and it indicates the external AC97 CODEC is ready or not. **CRDY** / **Description** 0 / CODEC is not ready. 1 / CODEC is ready.	R
19	CLPM	External CODEC Low Power Mode. This bit indicates the external CODEC is switched to low power mode or BIT_CLK is active from CODEC after wake up. **CLPM** / **Description** 0 / BIT_CLK is active. 1 / CODEC is switched to low power mode.	R
18	RSTO	External CODEC Registers Read Status Time Out. This bit indicates that the read status time out is detected or not. It is set to 1 if the data not return in 4 frames after a CODEC registers read command issued. **RSTO** / **Description** 0 / When read, indicates time out has not occurred. 1 / When read, indicates read status time out found. Write 0 clear this bit and write 1 is ignored. When RSTO is 1, it may trigger an interrupt depends on the interrupt enable setting.	RW
17	SADR	External CODEC Registers Status Address and Data Received. This bit indicates that address and data of an external AC '97 CODEC register	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>has or has not been received.
<table><tr><th>SADR</th><th>Description</th></tr><tr><td>0</td><td>When read, indicates no register address/data received.</td></tr><tr><td>1</td><td>When read, indicates address/data received.</td></tr></table>Write 0 clear this bit and write 1 is ignored. When SADR is 1, it may trigger an interrupt depends on the interrupt enable setting.</td><td></td></tr>
<tr><td>16</td><td>CADT</td><td>Command Address and Data Transmitted. This bit indicates that a CODEC register reading/writing command transmission has completed or not.
<table><tr><th>CADT</th><th>Description</th></tr><tr><td>0</td><td>When read, indicates the command has not done.</td></tr><tr><td>1</td><td>When read, indicates the command has done.</td></tr></table>Write 0 clear this bit and write 1 is ignored. When CADT is 1, it may trigger an interrupt depends on the interrupt enable setting.</td><td>RW</td></tr>
<tr><td>15:0</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
</table>

12.2.8 AIC I2S/MSB-justified Status Register (I2SSR)

I2SSR is used to reflect AIC status in I2S/MSB-justified. It is a read-only register.

<table>
<tr><td></td><td colspan="16">I2SSR</td><td colspan="16">0x1002001c</td></tr>
<tr><td>Bit</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td></td><td colspan="26">Reserved</td><td>CHBSY</td><td>TBSY</td><td>RBSY</td><td>BSY</td><td colspan="2">Reserved</td></tr>
<tr><td>RST</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31:3</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>5</td><td>CHBSY</td><td>AIC Transmitter busy in I2S/MSB-justified format.(Multi-channel status)
<table><tr><th>CHBSY</th><th>Description</th></tr><tr><td>0</td><td>AIC Transmitter part is idle or disabled.</td></tr><tr><td>1</td><td>AIC Transmitter part currently is transmitting or receiving a frame.</td></tr></table></td><td>R</td></tr>
<tr><td>4</td><td>TBSY</td><td>AIC Transmitter busy in I2S/MSB-justified format.
<table><tr><th>TBSY</th><th>Description</th></tr><tr><td>0</td><td>AIC Transmitter part is idle or disabled.</td></tr><tr><td>1</td><td>AIC Transmitter part currently is transmitting or receiving a frame.</td></tr></table></td><td>R</td></tr>
<tr><td>3</td><td>RBSY</td><td>AIC Receiver busy in I2S/MSB-justified format.
<table><tr><th>RBSY</th><th>Description</th></tr></table></td><td>R</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0	AIC Receiver part is idle or disabled.	
		1	AIC Receiver part currently is transmitting or receiving a frame.	
2	BSY	AIC busy in I2S/MSB-justified format.		R
		BSY	**Description**	
		0	AIC controller is idle or disabled.	
		1	AIC controller currently is transmitting or receiving a frame.	
1:0	Reserved	Writing has no effect, read as zero.		R

12.2.9 AIC AC97 CODEC Command Address & Data Register (ACCAR, ACCDR)

ACCAR and ACCDR are used to hold register address and data for external AC-link CODEC register read/write operation through SDATA_OUT. The format of ACCAR.CAR and ACCDR.CDR is compliant with AC'97 Component Specification 2.3 where ACCAR.CAR[19] of "1" specifies CODEC register read operation, of "0" specifies CODEC register write operation. The write access to ACCAR and ACCDR signals AIC to issue this operation. Please reference to 12.4.4 for software flow. These registers are valid only in AC-link. It is ignored in I2S/MSB-justified format.

	ACCAR																															**0x10020020**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												CAR																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writing has no effect, read as zero.	R
19:0	CAR	Command Address Register. This is used to hold 20-bit AC '97 CODEC register address transmitted in SDATA_OUT slot 1. After this field is write, it should not be write again until the operation is finished.	RW

	ACCDR																															**0x10020024**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												CDR																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writing has no effect, read as zero.	R
19:0	CDR	Command Data Register. This is used to hold 20-bit AC'97 CODEC register data transmitted in SDATA_OUT slot 2. After this field is write, it should not be write again until the operation is finished.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.2.10 AIC AC97 CODEC Status Address & Data Register (ACSAR, ACSDR)

ACSAR and ACSDR are used to receive the external AC-link CODEC registers address and data from SDATA_IN. When AIC receives CODEC register status from SDATA_IN, it set ACSR.SADR bit and put the address and data to ACSAR.SAR and ACSDR.SDR. Please reference to 12.4.4 for software flow. These registers are valid only in AC-link format and are ignored in I2S/MSB-justified format.

	ACSAR																															0x10020028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												SAR																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writing has no effect, read as zero.	R
19:0	SAR	CODEC Status Address Register. This is used to receive 20-bit AC '97 CODEC status address from SDATA_IN slot 1. Which reflect the register index for which data is being returned. The write operation is ignored.	R

	ACSDR																															0x1002002C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												SDR																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writing has no effect, read as zero.	R
19:0	SDR	CODEC Status Data Register. This is used to receive 20-bit AC '97 CODEC status data from SDATA_IN slot 2. The register data of external CODEC is returned. The write operation is ignored.	R

12.2.11 AIC I2S/MSB-justified Clock Divider Register (I2SDIV)

I2SDIV is used to set clock divider to generated BIT_CLK from SYS_CLK in I2S/MSB-justified format.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	I2SDIV																															0x10020030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							IDV									Reserved							DV								
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Writing has no effect, read as zero.	R
24:16	IDV	Audio IBIT_CLK clock divider value minus 1. I2SDIV.IDV is used to control the generating of IBIT_CLK from dividing SYS_CLK. The dividing value should be even and I2SDIV.IDV should be set to the dividing value minus 1. So I2SDIV.IDV bit0 is fixed to 1. IBIT_CLK frequency is fixed to 64 f_S in AIC, where f_S is the audio sample frequency. I2SDIV.IDV depends on SYS_CLK frequency f_{SYS_CLK}, which is selected according to external CODEC's requirement and internal PLL frequency. Please reference to 1.4.10 Serial Audio Clocks and Sampling Frequencies for further description.	RW
15:9	Reserved	Writing has no effect, read as zero.	R
8:0	DV	Audio BIT_CLK clock divider value minus 1. I2SDIV.DV is used to control the generating of BIT_CLK from dividing SYS_CLK. The dividing value should be even and I2SDIV.DV should be set to the dividing value minus 1. So I2SDIV.DV bit0 is fixed to 1. BIT_CLK frequency is fixed to 64 f_S in AIC, where f_S is the audio sample frequency. I2SDIV.DV depends on SYS_CLK frequency f_{SYS_CLK}, which is selected according to external CODEC's requirement and internal PLL frequency. Please reference to 1.4.10 Serial Audio Clocks and Sampling Frequencies for further description.	RW

12.2.12 AIC FIFO Data Port Register (AICDR)

AICDR is act as data input port to transmit FIFO when write and data output port from receive FIFO when read, one audio sample every time. The FIFO width is 24 bits. Audio sample with size N that is less than 24 is located in LSB N-bits. The sample size is specified by ACCR2.OASS and ACCR2.IASS in AC-link, and by I2SCR.WL in I2S/MSB-justified. The sample order is specified by ACCR1.XS and ACCR1.RS in AC-link. In I2S/MSB-justified, the left channel sample is prior to the right channel sample.

Care should be taken to monitor the status register to insure that there is room for data in the FIFO when executing a program read or write transaction. This is taken care automatically in DMA.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	AICDR																															0x10020034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								DATA																							
RST	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:0	DATA	FIFO port. When write to it, data is push to the transmit FIFO. When read from it, data is pop from the receiving FIFO.	RW

12.2.13 SPDIF Enable Register (SPENA)

The register SPENA is used to trigger SPDIF transmitter to work.

	SPENA																															0x10020080
Bit																									7	6	5	4	3	2	1	0
																									Reserved							SPEN
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:1	Reserved	Writing has no effect, read as zero.	R
0	SPEN	Enable / disable the SPDIF transmitter. 0: SPDIF transmitter is disabled 1: SPDIF transmitter is enabled	RW

12.2.14 SPDIF Control Register (SPCTRL)

The register SPCTRL is used to control SPDIF to work.

	SPCTRL																															0x10020084
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	DMA_EN	D_TYPE	SIGN_N	INVALID	SFT_RST	SPDIF_I2S	Reserved								M_TRIG	M_FFUR
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
15	DMA_EN	DMA transmitter enable bit. 0: DMA transmitter disable 1: DMA transmitter enable	RW
14	D_TYPE	If the bit number of data is less than16, the data in memory is as follows: 0: XXXXXXXXXXXXXXXX Data 0 XXXXXXXXXXXXXXXX Data 1 1: Data 1 Data 0 Data 3 Data 2	RW
13	SIGN_N	Signed to unsigned or not. If it is 1, the incoming and outgoing audio sample data will be transferred by toggle its most significant bit. 0: Not transfer 1: Do transfer	RW
12	INVALID	Data invalid bit. The data transmitted on SPDIF is valid or not. 0: Valid 1: Invalid	RW
11	SFT_RST	SPDIF FIFO software-reset. Set it to 1 and later it will be cleared by hardware auto. When SFT_RST returns back to 0, the FIFO finish reset. 0: Stop reset 1: Start reset	RW
10	SPDIF_I2S	Choose SPDIF or I2S. 0: I2S 1: SPDIF	
9:2	Reserved	Writing has no effect, read as zero.	R
1	M_TRIG	Trigger interrupt mask. 0: Enabled 1: Masked	RW
0	M_FFUR	FIFO underrun interrupt mask. 0: Enabled 1: Masked	RW

12.2.15 SPDIF State Register (SPSTATE)

The register SPSTATE is used to keep the state of SPDIF.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SPSTATE																															0x10020088
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved	FIFO_LVL							BUSY	Reserved					F_TRIG	F_FFUR
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	Reserved	Writing has no effect, read as zero.	R
14:8	FIFO_LVL	FIFO level. The bits indicate the amount of valid data in FIFO.	R
7	BUSY	SPDIF busy bit. 0: SPDIF is not working 1: SPDIF is working	R
6:2	Reserved	Writing has no effect, read as zero.	R
1	F_TRIG	Trigger flag. 0: Not active 1: Active	R
0	F_FFUR	FIFO underrun flag. 0: Not active 1: Active	RW

12.2.16 SPDIF Configure 1 Register (SPCFG1)

The register SPCFG1 is used to configure SPDIF.

	SPCFG1																															0x1002008C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved														INIT_LVL	ZRO_VLD	Reserved		TRIG		SRC_NUM				CH1_NUM				CH2_NUMl			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:18	Reserved	Writing has no effect, read as zero.	R
17	INIT_LVL	Initial level set bit. 0: SPDIF initial level is low 1: SPDIF initial level is high	
16	ZRO_VLD	The valid bit of channel state is 0 or 1 when play ZERO sample under FIFO underflow. 0: Valid	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Invalid	
15:14	Reserved	Writing has no effect, read as zero.	R
13:12	TRIG	Specify the trigger value of FIFO. TRIG / Description 00 / Trigger Value is 4. 01 / Trigger Value is 8. 10 / Trigger Value is 16. 11 / Trigger Value is 32.	RW
11:8	SRC_NUM	Source number. 0000:Unspecified 0001~1111:1~15	RW
7:4	CH1_NUM	Channel 1 number. 0000:Unspecified 0001~1111:A~O	RW
3:0	CH2_NUM	Channel 2 number. 0000:Unspecified 0001~1111:A~O	RW

12.2.17 SPDIF Configure 2 Register (SPCFG2)

The register SPCFG2 is used to configure SPDIF.

	SPCFG2																															0x10020090
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		FS				ORG_FRQ				SAMPL_WL			MAX_WL	CLK_ACU		CAT_CODE								CH_MD		Reserved		PRE	COPY_N	AUDIO_N	CON_PRO
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:26	FS	Sampling frequency. 0000:44.1kHz 0010:48kHz 0011:32kHz 1010:96kHz 1110:192kHz Others: Reference IEC60958-3	RW
25:22	ORG_FRQ	Original sampling frequency. 1111:44.1kHz 1101:48kHz	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1100:32kHz 0101:96kHz 0001:192kHz Others: Reference IEC60958-3	
21:19	SAMPL_WL	Sample word length. When MAX_WL=1: 001:20 bit 110:21 bit 010:22 bit 100:23 bit 101:24 bit Others: reserved When MAX_WL=0: 001:16 bit 110:17 bit 010:18 bit 100:19 bit 101:20 bit Others: reserved	RW
18	MAX_WL	Maximum audio sample word length. 0:20 bit；1:24 bit.	RW
17:16	CLK_ACU	Clock Accuracy of transmitted clock. 00: Level II 01: Level I 10: Level III 11: Interface frame rate not matched to sampling frequency	RW
15:8	CAT_CODE	Category code. Reference IEC60958-3 for full details. 00 indicates “general” mode.	RW
7:6	CH_MD	Channel mode choose bit. 00: Mode 0 01~11: Reserved	RW
5:4	Reserved	Writing has no effect, read as zero.	R
3	PRE	Pre-emphasis set bit. 0: None 1: 15us/15us	RW
2	COPY_N	Copyright set bit. 0: Copyright is asserted 1: Copyright is not asserted	RW
1	AUDIO_N	Linear PCM identification bit. 0: Audio sample word represents linear PCM samples 1: Audio sample word used for other purpose	RW
0	CON_PRO	Consumer mode and professional mode choose bit.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Consumer mode 1: Professional mode Professional is not supported in the chip.	

12.2.18 SPDIF FIFO Register (SPFIFO)

The register SPCFG1 is used to configure SPDIF.

	SPFIFO																															0x10020094
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								DATA																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:0	DATA	FIFO port. When write to it, data is push to the transmit FIFO. Read from it as 0.	W

12.3 Serial Interface Protocol

12.3.1 AC-link serial data format

Following figures are AC-link serial data format. Audio data is MSB adjusted, regardless of 8, 16, 18, 20, 24 bits sample size. When a 24-bits sample is transmitted, the LSB 4-bits are truncated. When try to record 24-bits sample, 4-bits of 0 are appended in LSB. Please reference to “AC ’97 Component Specification Revision 2.3, 2002”, provided by Intel Corporation, for details of AC ’97 architecture and AC-link specification.

Figure 12-7 AC-link audio frame format

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-8 AC-link tag phase, slot 0 format

Figure 12-9 AC-link data phases, slot 1 ~ slot 12 format

12.3.2 I2S and MSB-justified serial audio format

Normal I2S and MSB-justified are similar protocols for digitized stereo audio transmitted over a serial path.

The BIT_CLK supplies the serial audio bit rate, the basis for the external CODEC bit-sampling logic. Its frequency is 64 times the audio sampling frequency. Divided by 64, the resulting 8 kHz to 48 kHz or even higher signal signifies timing for left and right serial data samples passing on the serial data paths. This left/right signal is sent to the CODEC on the SYNC pin. Each phase of the left/right signal is accompanied by one serial audio data sample on the data pins SDATA_IN and SDATA_OUT.

Figure 12-10 I2S data format (A: LR mode)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

In the A: LR mode, first send the left channel in a stereo frame. One Left slot and one Right slot make a sample frame. It is the normal mode of I2S.

Figure 12-11 I2S data format (B: RL mode)

In the B: RL mode, first send the right channel in a stereo frame. One Right slot and one Left slot make a sample frame. It is used in same CODEC.

Figure 12-12 MSB-justified data format (C: LR mode)

In the C: LR mode, first send the left channel in a stereo frame. One Left slot and one Right slot make a sample frame. It is the normal mode in MSB-justified.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-13 MSB-justified data format (D: RL mode)

In the D: RL mode, first send the right channel in a stereo frame. One Right slot and one Left slot make a sample frame.

Figure 12-10 and Figure 12-12 provide timing diagrams that show formats for the normal I2S and MSB-justified modes of operations. Data is sampled on the rising edge of the BIT_CLK and data is sent out on the falling edge of the BIT_CLK.

Data is transmitted and received in frames of 64 BIT_CLK cycles(If BIT_CLK is generated internally). Each frame consists of a left sample and a right sample. Each sample holds 8, 16, 18, 20 or 24 bits of valid data. The LSB other bits of each sample is padded with zeroes.

In the normal I2S mode, the SYNC is low for the left sample and high for the right sample. Also, the MSB of each data sample lags behind the SYNC edges by one BIT_CLK cycle.
In the MSB-justified mode, the SYNC is high for the left sample and low for the right sample. Also, the
MSB of each data sample is aligned with the SYNC edges.

When use with the internal CODEC, the BIT_CLK and SYNC signals also with O_BIT_CLK and O_SYNC signals are provided by the internal CODEC from the SYSCLK, which is enabled by I2SCR.ESCLK and configured to 12MHz clock using CPM.

12.3.3 Audio sample data placement in SDATA_IN/SDATA_OUT

The placement of audio sample in incoming/outgoing serial data stream for all formats support in AIC is MSB (Most Significant Bit) justified. Suppose n bit sample composed by

S_{n-1}	S_{n-2}	…	S_2	S_1	S_0

Following table described the how sample data bits are transferred.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 12-3 Sample data bit relate to SDATA_IN/SDATA_OUT bit

AC-link Format						I2S/MSB-Justified Format					
SDATA	Audio Sample Size (bit)					SDATA					
IN/OUT	8	16	18	20	24	IN/OUT	8	16	18	20	24
B19	S7	S15	S17	S19	S23	B31	S7	S15	S17	S19	S23
B18	S6	S14	S16	S18	S22	B30	S6	S14	S16	S18	S22
B17	S5	S13	S15	S17	S21	B29	S5	S13	S15	S17	S21
B16	S4	S12	S14	S16	S20	B28	S4	S12	S14	S16	S20
B15	S3	S11	S13	S15	S19	B27	S3	S11	S13	S15	S19
B14	S2	S10	S12	S14	S18	B26	S2	S10	S12	S14	S18
B13	S1	S9	S11	S13	S17	B25	S1	S9	S11	S13	S17
B12	S0	S8	S10	S12	S16	B24	S0	S8	S10	S12	S16
B11	0	S7	S9	S11	S15	B23	0	S7	S9	S11	S15
B10	0	S6	S8	S10	S14	B22	0	S6	S8	S10	S14
B9	0	S5	S7	S9	S13	B21	0	S5	S7	S9	S13
B8	0	S4	S6	S8	S12	B20	0	S4	S6	S8	S12
B7	0	S3	S5	S7	S11	B19	0	S3	S5	S7	S11
B6	0	S2	S4	S6	S10	B18	0	S2	S4	S6	S10
B5	0	S1	S3	S5	S9	B17	0	S1	S3	S5	S9
B4	0	S0	S2	S4	S8	B16	0	S0	S2	S4	S8
B3	0	0	S1	S3	S7	B15	0	0	S1	S3	S7
B2	0	0	S0	S2	S6	B14	0	0	S0	S2	S6
B1	0	0	0	S1	S5	B13	0	0	0	S1	S5
B0	0	0	0	S0	S4	B12	0	0	0	S0	S4
						B11	0	0	0	0	S3
						B10	0	0	0	0	S2
						B9	0	0	0	0	S1
						B8	0	0	0	0	S0
						B7~ B0	0	0	0	0	0

If in 16 bits packed mode, the data transferred is the same as the 16 bits normal mode as shown above. But there are two samples in one word.

12.3.4 SPDIF Protocol

SPDIF block format is shown below:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-14 Block format

Sub-frame format in PCM mode is shown below:

0 3	4 7	8 27	28	29	30	31
Sync preamble	Auxiliary	LSB Audio sample word MSB	V	U	C	P

Figure 12-15 Sub-frame format in PCM mode

Sub-frame format in non-PCM mode is shown below:

0 3	4 10	11 27	28	29	30	31
Sync preamble	00000000	LSB Audio sample word MSB	V	U	C	P

Figure 12-16 Sub-frame format in non-PCM mode

12.4 AC97/I2S Operation

The AIC can be accessed either by the processor using programmed I/O instructions or by the DMA controller. The processor uses programmed I/O instructions to access the AIC and can access the following types of data.

- **The AIC memory mapped registers data**—All registers are 32 bits wide and are aligned to word boundaries.
- **AIC controller FIFO data—**An entry is placed into the transmit FIFO by writing to the I2S controller's Serial Audio Data register (AICDR). Writing to AICDR updates a transmit FIFO entry. Reading AICDR flushes out a receive FIFO entry.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- **The external CODEC registers for I2S CODEC—**CODEC registers can be accessed through the L3 bus. The L3 bus operation is emulated by software controlling three GPIO pins.
- **The external CODEC registers for AC97 CODEC**—An AC97 audio CODEC can contain up to sixty-four 16-bit registers. A CODEC uses a 16-bit address boundary for registers. The AIC supplies access to the CODEC registers through several registers.
- **The internal CODEC registers** can be accessed via memory-mapped registers in the CODEC.

The DMA controller can only access the FIFOs. Accesses are made through the data registers, as explained in the previous paragraph. The DMA controller responds to the following DMA requests made by the I2S controller:

- The transmit FIFO request is based on the transmit trigger-threshold (AICFR.TFTH) setting. See 12.2.1 for further details regarding AICFR.TFTH.
- The receive FIFO request is based on the receive trigger-threshold (AICFR.RFTH) setting. See 12.2.1 for further details regarding AICFR.RFTH.

Before operation to AIC, you may need to set proper PIN function selection from GPIO using if the pin is shared with GPIO.

Please also reference to “AC ’97 Component Specification Revision 2.3, 2002” when deal with AIC AC-link operations.

12.4.1 Initialization

At power-on or other hardware reset (WDT and etc), AIC is disabled. Software must initiate AIC and the internal or external CODEC after power-on or reset. If errors found in data transferring, or in other places, software must initial AIC and optional, the internal or external CODEC. Here is the initial flow.

1. Select internal or external CODEC (AICFR.ICDC).
2. If external CODEC is selected, select AC-link or I2S/MSB-Justified (AICFR.AUSEL). If internal CODEC is used, select I2S/MSB-Justified format (AICFR.AUSEL=1). If the resettlement without involving link format and architecture changing, this step can be skip.
3. If I2S/MSB-Justified is selected, select between I2S and MSB-Justified (I2SCR.AMSL).
4. Decide BIT_CLK direction (AICFR.BCKD) and SYNC direction (AICFR.SYNCD).
5. If BIT_CLK is configured as output, BIT_CLK divider I2SDIV.DV must be set to what correspond with the values as shown in Table 12-7. And the clock selection and the divider between PLL clock out and AIC also must be set (CFCR.I2S and I2SCDR in CPM). If internal CODEC is used, select 12MHz clock input (via set proper value in CFCR.I2S and I2SCDR), I2S format (I2SCR.AMSL=0), input BIT_CLK (AICFR.BCKD=0), input SYNC (AICFR.SYNCD=0).
6. Enable AIC by write 1 to AICFR.ENB.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7 If it needs to reset AIC registers and flush FIFOs, write 1 to AICFR.RST. If it need only flush FIFOs, write 1 to AICCR.FLUSH. BIT_CLK must exist here and after.
8 In AC-link format, issue a warm or cold CODEC reset.
9 In AC-link format, configure AC ’97 CODEC via ACCAR and ACCDR registers. If the resettlement doesn’t involving AC’97 CODEC registers changing, this step can be skipped.
10 In case of external CODEC with I2S/MSB-Justified format, configure I2S/MSB-justified CODEC via the control bus connected to the CODEC, for instance I2C or L3, depends on CODEC. In case of internal CODEC, configure CODEC via CODEC’s memory mapped registers. If the resettlement without involving I2S/MSB-justified CODEC or ADC/DAC function changing, this step can be skip.

12.4.2 AC ’97 CODEC Power Down

AC ‘97 CODEC can be placed in a low power mode. When the CODEC’s power-down register (26h), is programmed to the appropriate value, the CODEC will be put in a low power mode and both BIT_CLK and SDATA_IN will be brought to and held at a logic low voltage level.

Once powered down, re-activation of the AC-link via re-assertion of the SYNC signal must not occur for a minimum of four audio frame times following the frame in which the power down was triggered. When AC-link powers up it indicates readiness via the CODEC Ready bit (input slot 0, bit 15).

12.4.3 Cold and Warm AC ’97 CODEC Reset

AC-link reset operations occur when the system is initially powered up, when resuming from a lower powered sleep state, and in response to critical subsystem failures that can only be recovered from with a reset.

12.4.3.1 Cold AC ‘97 CODEC Reset

A cold reset is achieved by asserting RESET# for the minimum specified time. By driving RESET# low, BIT_CLK, and SDATA_IN will be activated, or re-activated as the case may be, and all AC ‘97 CODEC registers will be initialized to their default power on reset values.

RESET# is an asynchronous AC ‘97 CODEC input.

Figure 12-17 Cold AC ’97 CODEC Reset Timing

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 12-4 Cold AC ’97 CODEC Reset Timing parameters

Parameter	Symbol	Min	Type	Max	Units
RESET# active low pulse width	T_{rst_low}	1.0	-	-	μs
RESET# inactive to BIT_CLK startup delay	$T_{rst2clk}$	162.8	-	-	ns

12.4.3.2 Warm AC ‘97 CODEC Reset

A warm AC‘97 reset will re-activate the AC-link without altering the current AC‘97 register values. Driving SYNC high for a minimum of 1 μs in the absence of BIT_CLK signals a warm reset.

Within normal audio frames SYNC is a synchronous AC ‘97 CODEC input. However, in the absence of BIT_CLK, SYNC is treated as an asynchronous input used in the generation of a warm reset to AC ’97 CODEC.

Figure 12-18 Warm AC ’97 CODEC Reset Timing

Table 12-5 Warm AC ’97 CODEC Reset Timing Parameters

Parameter	Symbol	Min	Type	Max	Units
SYNC active high pulse width	T_{sync_high}	1.0	-	-	Ms
SYNC inactive to BIT_CLK startup delay	$T_{sync2clk}$	162.8	-	-	Ns

12.4.4 External CODEC Registers Access Operation

The external audio CODEC can be configured/controlled by its internal registers. To access these registers, an I2S/MSB-justified CODEC usually employs L3 bus, SPI bus, I2C bus or other control bus. The L3 bus operation can be emulated by software by using 3 GPIO pins of the chip. For AC ’97, “AC ’97 Component Specification” defines the CODEC register access protocol. Several registers are provided in AIC to accomplish this task.

The ACCAR and ACCDR are used to send a register accessing request command to external AC’97 CODEC. The ACSAR and ACSDR are used to receive a register’s content from external AC’97 CODEC. The register accessing request and the register’s content returning is asynchronous.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The AC'97 CODEC register accessing request flow:

1. If ACSR.CADT is 0, wait for 25.4μs. If no previous accessing request, this step can be skip.
2. Clear ACSR.CADT.
3. If read access, write read-command and register address to ACCAR, if write access, write write-command and register address to ACCAR and write data to ACCDR. Any order of write ACCAR and ACCDR is OK.
4. Polling for ACSR.CADT changing to 1, which means the request has been send to CODEC via AC-link.

The AC'97 CODEC register content receiving flow by polling:

1. Polling for ACSR.SADR changing to 1.
2. Read the CODEC register's address from ACSAR and content from ACSDR.
3. Clear ACSR.SADR.

The AC'97 CODEC register content receiving flow by interrupt:

1. Before accessing request, clear ACSR.SADR and set ACCR2.ESADR.
2. Waiting for the interrupt. When the interrupt is found, check if ACSR.SADR is 1, if not, repeat this step again.
3. Read the CODEC register's address from ACSAR and content from ACSDR.
4. Clear ACSR.SADR.

12.4.5 Audio Replay

Outgoing audio sample data (from AIC to CODEC) is written to AIC transmit FIFO from processor via store instruction or from memory via DMA. AIC then takes the data from the FIFO, serializes it, and sends it over the serial wire SDATA_OUT to an external CODEC or over an internal wire to an internal CODEC.

The audio transmission is enabled automatically when the AIC is enabled by set AICFR.ENB. But all replay data is zero at this time except both of the following conditions are true:

1. AICCR.ERPL must be 1. If AICCR.ERPL is 0, value of zero is send to CODEC even if there are samples in transmit FIFO.
2. At least one audio sample data in the transmit FIFO. If the transmit FIFO is empty, value of zero or last sample depends on AICFR.LSMP, is send to CODEC even if AICCR.ERPL is 1.

Here is the audio replay flow:

1. Configure the CODEC as needed.
2. Configure sample size by AICCR.OSS.
3. Configure sample channels (AICCR.CHANNEL).
4. If sample size is configured 16 bit, select packed or unpacked mode (AICCR.PACK16).
5. If two channels is configured, select the right-channel-first sample data or not (I2SCR.RFIRST).
6. If two channels is configured, select the sample data switched or not (I2SCR.SWLH).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7 Configure sample rate by clock dividers (for I2S/MSB-Justified format with BIT_CLK is provided internally) or by CODEC registers (for AC-link or BIT_CLK provided by external CODEC) or by accessing CODEC internal registers (for internal CODEC).
8 For AC-link, configure replay channels by ACCR1.XS.
9 Some other configurations: mono to stereo, endian switch, signed/unsigned data transfer, transmit FIFO configuration, play ZERO or last sample when TX FIFO under-run, and etc.
10 Write 1 to AICCR.ERPL.
It is suggested that at least a frame of PCM data is pre-filled in the transmit FIFO to prevent FIFO under-run flag (AICSR.TUR).
But when using internal CODEC, write first frame of PCM data to transmit FIFO till TX FIFO under-run (AICSR.TUR is set to 1), otherwise left/right channel may be switched.
11 Fill sample data to the transmit FIFO. Repeat this till finish all sample data. In this procedure, please control the FIFO to make sure no FIFO under-run and other errors happen. When the transmit FIFO under-run, noise or pause may be heard in the audio replay, AICSR.TUR is 1, and if AICCR.ETUR is 1, AIC issues an interrupt. Please reference to 12.4.7 for detail description on FIFO.
12 Waiting for AICSR.TFL change to 0. So that all samples in the transmit FIFO has been replayed, then we can have a clean start up next time.
13 Write 0 to AICCR.ERPL.

NOTE: Before replaying Open ADC BITCLK and close it to generating Record internal circuit reset when using internal CODEC.

12.4.6 Audio Record

Incoming audio sample data (from CODEC to AIC) is received from SDATA_IN (for an external CODEC) or an internal wire (for an internal CODEC) serially and converted to parallel word and stored in AIC receive FIFO. Then the data can be taken from the FIFO to processor via load instruction or to memory via DMA.

The audio recording is enabled automatically when the AIC is enabled by set AICFR.ENB. But all received data is discarded at this time except both of the following conditions are true:

1 AICCR.EREC must be 1. If AICCR.EREC is 0, the received data is discarded even if there are rooms in the receive FIFO.
2 At least one room left in the receive FIFO. If the receive FIFO is full, the received data is discarded even if AICCR.EREC is 1.

Here is the audio record flow:

1 Configure the CODEC as needed.
2 Configure sample size by AICCR.ISS.
3 Configure sample rate by clock dividers (for I2S/MSB-Justified format with BIT_CLK is provided internally) or by CODEC registers (for AC-link or BIT_CLK provided by external CODEC) or by CODEC memory mapped registers (for internal CODEC).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 Some other configurations: signed/unsigned data transfer, receive FIFO configuration, and etc.
5 Write 1 to AICCR.EREC. Make sure there are rooms available in the receive FIFO before set AICCR.EREC. Usually, it should empty the receive FIFO by fetch data from it before set AICCR.EREC.
6 Take sample data form the receive FIFO. Repeat this till the audio finished. In this procedure, please control the FIFO to make sure no FIFO over-run and other errors happen. When the receive FIFO over-run, same recorded audio samples will be lost, AICSR.ROR is 1, and if AICCR.EROR is 1, AIC issues an interrupt. Please reference to 12.4.7 for detail description on FIFO. For AC-link, ACCR1.RS tells which channels are recorded.
When using internal CODEC, the first data should be ignored.
7 Write 0 to AICCR.EREC.
8 Take sample data from the receive FIFO until AICSR.RFL change to 0. So that all samples in the receive FIFO has been taken away, then we can have a clean start up next time. When the receive FIFO is empty, read from it returns zero.

12.4.7 FIFOs operation

AIC has two FIFOs, one for transmit audio sample and one for receive. All AIC played/recorded audio sample data is taken from/send to transmit/receive FIFOs. The RX FIFO is in 24 bits width and 32 entries depth, one entry for keeping one audio sample regardless of the sample size. The RX FIFO is in 32 bits width and 64 entries depth, one entry for keeping one audio sample regardless of the sample size, but in 16 bits packed mode, one entry for keeping two audio samples. AICDR.DATA provides the access point for processor/DMA to write to transmit FIFO and read from receive FIFO. One time access to AICDR.DATA process one sample. The sample data should be put in LSB (Least Significant Bit) in memory or processor registers. For transmitting, bits exceed sample are discarded. For receiving, these bits are set to 0. Figure 12-19 illustrates the FIFOs access.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-19 Transmitting/Receiving FIFO access via APB Bus

The software and bus initiator must guarantee the right sample placement at the bus.

In case of DMA bus initiator, one 24, 20, 18 bits audio sample must occupies one 32-bits word in memory, so 32-bits width DMA must be used. One 16 bits sample occupies one 16-bits half word in memory, so 16-bits width DMA must be used. One 8-bits sample occupies one byte in memory, and use 8-bits width DMA except 16bits packed mode. If in 16 bits packed mode, Two 16 bits sample occupies one 32-bits word in memory, so 32-bits width DMA must be used.

In case of processor bus initiator, any type of the audio sample must occupy one CPU general-purpose register at LSB, and read/write from/to AICDR.DATA with 32-bits load/store instruction. When process small sample size, 16-bits or 8-bits, software may need to do the data pack/unpack except 16 bits packed mode. In the 16bits packed mode, the sample data is packed, and two 16 bits audio samples occupy one CPU general-purpose register.

The AICFR.TFTH and AICFR.RFTH are used to set the FIFO level thresholds, which are the trig levels of DMA request and/or FIFO service interrupt. The AICFR.TFTH and AICFR.RFTH should be set to proper value; too small or too big are not good. When AICFR.RFTH is too small, or

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

AICFR.TFTH is too big, the DMA burst length or the number of sample can be processed by processor is too small, which harms the bus or processor efficiency. When AICFR.RFTH is too big or AICFR.TFTH is too small, the bus or the interrupt latency left for under-run/over-run is too small, which may causes replay/record errors.

AICSR.TUR is set to 1 during transmit under-run conditions. If AICCR.ETUR is 1, this can trigger an interrupt. During transmit under-run conditions, zero or last sample is continuously sent out across the serial link. Transmit under-run can occur under the following conditions:

1 Valid transmit data is still available in memory, but the DMA controller/processor starves the transmit FIFO, as it is busy servicing other higher-priority tasks.
2 The DMA controller/processor has transferred all valid data from memory to the transmit FIFO.

AICSR.ROR is set to 1 during receive over-run conditions. If AICCR.EROR is 1, this can trigger an interrupt. During receive over-run conditions, data sent by the CODEC is lost and is not recorded.

When replay/record two channels data, the left channel is default the first data in FIFOs and in the serial link. If multiple channels in AC-link are used, the channel sample order is follows the slot order. In 16bits packed mode, could configure that the left channel is the first data or the right channel. By default, the 16 bits LSB is left channel, 16 bits MSB is the right channel. But it also could be switched the Left or the Right channel (I2SCR.SWLH).

12.4.8 Data Flow Control

There are three approaches provided to control/synchronize the audio incoming/outgoing data flow.

12.4.8.1 Polling and Processor Access

AICSR.RFL and AICSR.TFL reflect how many samples exist in receiving and transmitting FIFOs. Through read these register fields, processor can detect when there are samples in receiving FIFO in audio record and then load them from the RX-FIFO, and when there are rooms in transmitting FIFO in audio replay and then store samples to the TX-FIFO.

Polling approach is in very low efficiency and is not recommended.

12.4.8.2 Interrupt and Processor Access

Set proper values to AICFR.TFTH and AICFR.RFTH, the FIFO interrupts trig thresholds. Set AICCR.ETFS and/or AICCR.ERFS to 1 to enable transmitting and/or receiving FIFO level trigger interrupts. When the interrupt found, it means there are rooms or samples in the TX or RX FIFO, and processor can store or load samples to or from the FIFO.

Interrupt approach is more efficient than polling approach.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.4.8.3 DMA Access

Audio data is real time stream, though it is in low data bandwidth, usually less than 1.2Mbps. DMA approach is the most efficient and is the recommended approach.

To enable DMA operation, set AICCR.TDMS and AICCR.RDMS to 1 for transmit and receive respectively. It also needs to allocate two channels in DMA controller for data transmitting and receiving respectively. Please reference to the processor's DMA controller spec for the details.

The AICFR.TFTH and AICFR.RFTH are used to set the transmitting and receiving FIFO level thresholds, which determine the issuing of DMA request to DMA controller. To respond the request, DMAC initiator and controls the data movement between memory and TX/RX FIFO.

12.4.9 Audio Samples format

12.4.9.1 16 bits packed mode

One channel (mono) mode and two channels (stereo) mode:

Figure 12-20 One channel (Left) and Two channels (right) mode (16 bits packed mode)

Four channels mode and six channels mode:

Figure 12-21 Four channels (Left) and Six channels (right) mode (16 bits packed mode)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Eight channels mode:

bit31 …… bit0		
R3	R2	0xXXXXX00C
R1	R0	0xXXXXX008
L3	L2	0xXXXXX004
L1	L0	0xXXXXX000

Figure 12-22 Eight channels mode (16 bits packed mode)

12.4.9.2 Normal mode.

One channel (Mono) and two channels (stereo) mode:

bit31 …… bit0		
0	L0	0xXXXXX000

bit31 …… bit0		
0	R0	0xXXXXX004
0	L0	0xXXXXX000

Figure 12-23 One channel (Left) and Two channels (right) mode

Four channels mode and six channels mode:

bit31 …… bit0		
0	R1	0xXXXXX00C
0	R0	0xXXXXX008
0	L1	0xXXXXX004
0	L0	0xXXXXX000

bit31 …… bit0		
0	R2	0xXXXXX014
0	R1	0xXXXXX010
0	R0	0xXXXXX00C
0	L2	0xXXXXX008
0	L1	0xXXXXX004
0	L0	0xXXXXX000

Figure 12-24 Four channels (Left) and Six channels (right) mode

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Eight channel mode:

bit31 …… bit0		
0	R3	0xXXXXX01C
0	R2	0xXXXXX018
0	R1	0xXXXXX014
0	R0	0xXXXXX010
0	L3	0xXXXXX00C
0	L2	0xXXXXX008
0	L1	0xXXXXX004
0	L0	0xXXXXX000

Figure 12-25 Eight channels mode

12.4.10 Serial Audio Clocks and Sampling Frequencies

For internal CODEC, CODEC module containing the audio CODEC circuit/logic and corresponding controlling registers. CODEC needs a 12MHz clock from CPM called SYS_CLK and provides I_BITCLK, O_BITCLK and I_SYNC, O_SYNC (left-right clock which is the sample rate as ADC or DAC) to AIC for outgoing and incoming audio respectively. These clocks change when change the sample rate in CODEC controlling registers. When using internal CODEC, must configure SYNC and BIT_CLK as input, more details refers to CODEC Spec.

For AC-link, the bit clock is input from chip external and is fixed to 12.288MHz. The sample frequency of 48kHz is supported in nature. Variable Sample Rate feature is supported in this AIC. If the CODEC supports this feature, sample rate other than 48kHz audio data can be replay directly. Otherwise, software has to do the rate transfer to replay other sample rate audio data. Double rate, 96kHz or even 88.2kHz audio is also supported with proper CODEC.

Following are for BIT_CLK/SYS_CLK configuration in I2S/MSB-Justified format with external CODEC.

The BIT_CLK is the rate at which audio data bits enter or leave the AIC. BIT_CLK can be supplied either by the CODEC or an internally PLL. If it is supplied internally, BIT_CLK is configured as output pins, and is supplied out to the CODEC. If BIT_CLK is supplied by the CODEC, then it is configured as an input pin. Register bit AICFR.BCKD is used to select BIT_CLK direction.

The audio sampling frequency is the frequency of the SYNC signal, which must be 1/64 of BIT_CLK,

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

f_{BIT_CLK} = 64 f_S. But SYNC signal frequency is not fixed when using internal CODEC.

SYS_CLK is only for CODEC. It usually takes one of the two roles, as CODEC master clock input or as CODEC over-sampling clock input. If SYS_CLK roles as CODEC master clock input, it usually should be set to a fixed frequency according to CODEC requirement but independent to audio sample rate. In this case, usually there is a PLL in the CODEC and CODEC roles master mode. See Figure 12-3 for the interface diagram. This is the recommended AIC CODEC system configuration.

If SYS_CLK roles as CODEC over-sampling clock, its frequency is usually 4, 6, 8 or 12 times of BIT_CLK frequency, which are 256, 384, 512 and 768 times of audio sample rates. Following table lists the relation between sample rate, BIT_CLK and SYS_CLK frequencies.

Table 12-6 Audio Sampling rate, BIT_CLK and SYS_CLK frequencies

Sample Rate f_S (kHz)	BIT_CLK (MHz) f_{BIT_CLK} = 64 f_S	SYS_CLK (MHz)			
		256 f_S	384 f_S	512 f_S	768 f_S
48	3.072	12.288	18.432	24.576	36.864
44.1	2.8224	11.2896	16.9344	22.5792	33.8688
32	2.048	8.192	12.288	16.384	24.576
24	1.536	6.144	9.216	12.288	18.432
22.05	1.4112	5.6448	8.4672	11.2896	16.9344
16	1.024	4.096	6.144	8.192	12.288
11.025	0.7056	2.8224	4.2336	5.6448	8.4672
8	0.512	2.048	3.072	4.096	6.144

In this processor, SYS_CLK can be selected from EXCLK or generated by dividing the PLL output clock in a CPM divider controlled by I2SCDR. If BIT_CLK is chosen as an output, another divider in AIC is used to divide SYS_CLK for it.

Figure 12-26 SYS_CLK, BIT_CLK and SYNC generation scheme

Note: SCLK_A and SCLK_M please refer to CPM.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The setting of I2SDIV.DV is shown in Following table.

Table 12-7 BIT_CLK divider setting

I2SDIV.DV	f_{SYS_CLK}	f_{BIT_CLK}	f_{SYS_CLK} / f_{BIT_CLK}
0x1	128 f_S	64 f_S	2
0x2	196 f_S	64 f_S	3
0x3	256 f_S	64 f_S	4
0x5	384 f_S	64 f_S	6
0x7	512 f_S	64 f_S	8
0xB	768 f_S	64 f_S	12

As we observe in Table 12-6, if SYS_CLK is taken as over-sampling clock by CODEC, the common multiple of all SYS_CLK frequencies is much bigger than the PLL output clock frequency. To generate all different SYS_CLK frequencies, one approach is change PLL frequency according to sample rate. This is not realistic, since frequently change PLL frequency during normal operation is not recommended.

Another approach is to found some approximate common multiples of all SYS_CLK frequencies according to the fact that there tolerance in audio sample rate. Take f_{SYS_CLK} = 256 f_S, the following table list most frequencies, which are less than 400MHz, with relatively small sample rate errors. It is suggested to set PLL frequency as close to the frequencies listed as possible, then use clock dividers to generate different SYS_CLK/BIT_CLK for different sample rate.

Table 12-8 Approximate common multiple of SYS_CLK for all sample rates

Approximate Common Frequency (MHz)	Max Error Caused in Audio Sample Rate (%)
123.53	0.53
147.11	0.24
170.68	0.79
235.5	0.87
247.06	0.53
270.64	0.11
280.56	0.73
294.22	0.24
305.14	0.67
317.79	0.53
329.57	0.66
341.35	0.79
347	0.85

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

353.13	0.90
358.79	0.69
370.59	0.53
382.96	0.54
394.17	0.24

Take PLL = 270.64 MHz as an example, the following table lists the divider settings for various sample rates.

Table 12-9 CPM/AIC clock divider setting for various sampling rate if PLL = 270.64MHz

Sample Rate (kHz)	I2SCDR	I2SDIV.DV	Sample Rate Error (%)
48	1	11	0.11
44.1	1	12	-0.11
32	0	33	0.11
24	1	22	0.11
22.05	1	24	-0.11
16	1	33	0.11
12	1	44	0.11
11.025	1	48	-0.11
8	1	66	0.11

For an EXCLK clock frequency, try to generate PLL frequencies as close to the frequencies listed in Table 12-8 as possible. Table 12-10 lists the PLL parameters and audio sample errors at different PLL frequencies for EXCLK at 12MHz.

Table 12-10 PLL parameters and audio sample errors for EXCLK=12MHz

PLL			Max Sample Rate Error
M	N	Freq (MHz)	
103	10	123.6	0.59%
49	4	147	0.31%
128	9	170.67	0.79%
157	8	235.5	0.87%
103	5	247.2	0.59%
65	3	260	0.82%
45	2	270	0.35%
203	9	270.67	0.12%
113	5	271.2	0.32%
187	8	280.5	0.75%
237	10	284.4	0.81%
49	2	294	0.31%

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

178	7	305.14	0.67%
53	2	318	0.60%
302	11	329.45	0.70%
256	9	341.33	0.79%
318	11	346.91	0.88%
206	7	353.14	0.90%
299	10	358.8	0.69%
247	8	370.5	0.55%
351	11	382.91	0.55%
230	7	394.29	0.27%

The BIT_CLK should be stopped temporary when change the divider settings, or when change BIT_CLK source (from internal or external), to prevent clock glitch. Register I2SCR.STPBK is provided to assist the task. When I2SCR.STPBK = 1, BIT_CLK is disabled no matter whether it is generated internally or inputted from the external source. The operation flow is described in following.

1. Stop all replay/record by clear AICCR.ERPL and AICCR.EREC.
2. Polling I2SSR.BSY till it is 0.
3. Stop the BIT_CLK by write 1 to I2SCR.STPBK.
4. Operations concerning BIT_CLK.
5. Resume the BIT_CLK by write 0 to I2SCR.STPBK.

12.4.11 Interrupts

The following status bits, if enabled, interrupt the processor:

- Receive FIFO Service (AICSR.RFS). It's also DMA Request.
- Transmit FIFO Service (AICSR.TFS). It's also DMA Request.
- Transmit Under-Run (AICSR.TUR).
- Receive Over-Run (AICSR.ROR).
- Command Address and Data Transmitted, AC-link only (ACSR.CADT).
- External CODEC Registers Status Address and Data Received, AC-link only (ACSR.SADR).
- External CODEC Registers Read Status Time Out, AC-link only (ACSR.RSTO).

For further details, see the corresponding register description sections.

12.5 SPDIF Guide

12.5.1 Set SPDIF clock frequency

Set SPDIF clock frequency is as same as i2s clock.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.5.2 PCM audio mode operation (Reference IEC60958)

1 Set SPCFG1 and SPCFG2 to configure SPDIF transmitter.
   a Set SPCFG2.CON_PRO to 0 to choose consumer mode.
   b Set SPCFG2. AUDIO_N to 0 to choose linear PCM audio data mode.
   c Set SPCFG1.XXX to configure SPDIF.
   d Set SPCFG2.XXX to configure SPDIF.
2 Set SPCTRL.DMA_EN to choose DMA mode or CPU mode.
3 Set SPCTRL.SIGN_N to choose whether to transfer the most significant bit by toggle or not.
4 Set SPCTRL. SFT_RST to 1 reset FIFO.
5 Wait SPCTRL. SFT_RST set to be set 0 by hardware.
6 Set SPCTRL.M_TRIG and SPCTRL.M_FFUR to enable or disable the interrupt.
7 Set SPCTRL.INVALID 1 or 0 to set the V bit of sub-frame.
8 Set SPENA.SPEN to 1 to Enable SPDIF to transmitter.

12.5.3 Non-PCM mode operation (Reference IEC61937)

1 Set SPCFG1 and SPCFG2 to configure SPDIF transmitter.
   a Set SPCFG2.CON_PRO to 0 to choose consumer mode.
   b Set SPCFG2.AUDIO_N to 1 to choose non-PCM mode.
   c Set SPCFG1.SRC_NUM to 0.
   d Set SPCFG1.CH1_NUM to 0.
   e Set SPCFG1.CH2_NUM to 0.
   f Set SPCFG2.PRE to 0.
   g Set SPCFG2.CH_MD to 0.
   h Set SPCFG2.ORG_FRQ to 0.
   i Set SPCFG2.SAMPL_WL to 0.
   j Set SPCFG2.MAX_WL to 0.
   k Set SPCFG1.XXX to configure SPDIF.
   l Set SPCFG2.XXX to configure SPDIF.
2 Set SPCTRL.DMA_EN to choose DMA mode or CPU mode.
3 Set SPCTRL.SIGN_N to choose whether to transfer the most significant bit by toggle or not.
4 Set SPCTRL. SFT_RST to 1 reset FIFO.
5 Wait SPCTRL. SFT_RST to be set to 0 by hardware.
6 Set SPCTRL.M_TRIG and SPCTRL.M_FFUR to enable or disable the interrupt.
7 Set SPCTRL.INVALID 1 or 0 to set the V bit of sub-frame.
8 Set SPENA.SPEN to 1 to Enable SPDIF to transmitter.

12.5.4 Disable operation

1 Set SPENA.SPEN to 0 to disable SPDIF to transmitter.
2 Wait SPSTATE.BUSY to be set to 0 by hardware.
3 You can do other operation.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

13 PCM Interface

13.1 Overview

The PCM has the following features:

- Data starts with the frame PCMSYN or one PCMCLK later
- Support three modes of operation for PCM
  - Short frame sync mode
  - Long frame sync mode
  - Multi-slot mode
- Data is transferred and received with the MSB first
- Support master mode and slave mode
- The PCM serial output data, PCMDOUT, is clocked out using the rising edge of the PCMSCLK
- The PCM serial input data, PCMDIN, is clocked in on the falling edge of the PCMSCLK
- 8/16 bit sample data sizes supported
- DMA transfer mode supported
- Two FIFOs for transmit and receive respectively with 16 samples capacity in every direction

13.2 Pin Description

There are all 4 pins used to connect between PCM interface and an external device. They are listed and described in following table.

Table 13-1 PCM Interface Pins Description

Name	I/O	Description
PCMCLK	Input / Output	PCM Serial clock Line signal input/output
PCMSYN	Input / Output	PCM sync signal input/output
PCMDOUT	Output	PCM Serial data output
PCMDIN	Input	PCM Serial data input

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.3 Block Diagram

13.4 Registers

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

? - value unknown after reset

13.4.1 Registers Memory Map

Table 13-2 Registers Memory Map-Address Base

Name	Base	Description
PCM0	0x10071000	Address base of DMIC Controller.

Table 13-3 PCM Registers Description

Name	Description	Reset Value	Address offset	Access Size
PCMCTL0	PCM Control Register	0x00000000	0x000	32
PCMCFG0	PCM Configure Register	0x00000110	0x004	32
PCMDP0	PCM FIFO Data Port Register	0x00000000	0x008	32
PCMINTC0	PCM Interrupt Control Register	0x00000000	0x00c	32
PCMINTS0	PCM Interrupt Status Register	0x00000100	0x010	32
PCMDIV0	PCM Clock Divide Register	0x00000001	0x014	32

13.4.2 Register Description

13.4.2.1 PCM Control Register (PCMCTL)

	PCMCTL0																															BASE + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																						ERDMA	ETDMA	LSMP	ERPL	EREC	FLUSH	RST	SYNEN	CLKEN	PCMEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:10	Reserved	Writing has no effect, read as zero.	R
9	ERDMA	Receive DMA Enable. This bit is used to enable or disable the DMA during receiving audio data. ERDMA: Receive DMA 0: Disabled. 1: Enabled.	RW
8	ETDMA	Transmit DMA Enable. This bit is used to enable or disable the DMA during transmit audio data.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			ETDMA / **Transmit DMA** 0 / Disabled. 1 / Enabled.	
7	LSMP	Select between play last sample or play ZERO sample in TX FIFO underflow. ZERO sample means sample value is zero. **LSMP** / **CODEC used** 0 / Play ZERO sample when TX FIFO underflow. 1 / Play last sample when TX FIFO underflow.		RW
6	ERPL	Enable Playing Back function. This bit is used to disable or enable the audio sample data transmitting. **ERPL** / **Description** 0 / PCM Playing Back Function is Disabled. 1 / PCM Playing Back Function is Enabled.		RW
5	EREC	Enable Recording Function. This bit is used to disable or enable the audio sample data receiving. **EREC** / **Description** 0 / PCM Recording Function is Disabled. 1 / PCM Recording Function is Enabled.		RW
4	FLUSH	FIFO Flush. Write 1 to this bit flush transmit/receive FIFOs to empty. Writing 0 to this bit has no effect and this bit is always reading 0.		W
3	RST	Reset PCM. Write 1 to this bit reset PCM registers and FIFOs. Writing 0 to this bit has no effect and this bit is always reading 0.		W
2	Reserved	Writing has no effect, read as zero.		R
1	CLKEN	Enable the serial clock division logic. Must be HIGH for the PCM to operate.		RW
0	PCMEN	Enable PCM function. This bit is used to enable or disable the PCM function. **PCMENB** / **Description** 0 / Disable PCM Controller. 1 / Enable PCM Controller.		RW

13.4.2.2 PCM Configuration Register (PCMCFG)

	PCMCFG0																															BASE + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																	SLOT		ISS	OSS	IMSBPOS	OMSBPOS	RFTH				TFTH				PCMMOD
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0

Bits	Name	Description	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td>31:15</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>14:13</td><td>SLOT</td><td>Controls the amount of valid PCM timeslot in one PCMSYN frame.
<table>
<tr><th>SLOT</th><th>Number of slot in one frame</th></tr>
<tr><td>00</td><td>1 slot</td></tr>
<tr><td>01</td><td>2 slot</td></tr>
<tr><td>10</td><td>3 slot</td></tr>
<tr><td>11</td><td>4 slot</td></tr>
</table></td><td>RW</td></tr>
<tr><td>12</td><td>ISS</td><td>Input Sample Size. These bits reflect input sample data size to memory or register. The data sizes supported are: 8/16bits. The sample data is LSB-justified in memory/register.
<table>
<tr><th>ISS</th><th>Sample Size</th></tr>
<tr><td>0</td><td>8 bit</td></tr>
<tr><td>1</td><td>16 bit</td></tr>
</table></td><td>RW</td></tr>
<tr><td>11</td><td>OSS</td><td>Output Sample Size. These bits reflect output sample data size from memory or register. The data sizes supported are: 8/16 bits. The sample data is LSB-justified in memory/register.
<table>
<tr><th>OSS</th><th>Sample Size</th></tr>
<tr><td>0</td><td>8 bit</td></tr>
<tr><td>1</td><td>16 bit</td></tr>
</table></td><td>RW</td></tr>
<tr><td>10</td><td>IMSBPOS</td><td>Controls the position of the MSB bit in the serial input stream relative to the PCMSYN signal.
0: MSB is captured on the falling edge of PCMCLK during the same cycle that PCMSYNC is high
1: MSB is captured on the falling edge of PCMCLK during the cycle after the PCMSYNC is high</td><td>RW</td></tr>
<tr><td>9</td><td>OMSBPOS</td><td>Controls the position of the MSB bit in the serial output stream relative to the PCMSYN signal.
0: MSB sent during the same clock that PCMSYN is high
1: MSB sent on the next PCMSCLK cycle after PCMSYNC is high</td><td>RW</td></tr>
<tr><td>8:5</td><td>RFTH</td><td>Receive FIFO threshold for interrupt or DMA request. Determines when the RFS flags go active for the RXFIFO. When the sample number in receive FIFO, indicated by PCMINTS.RFL, is great than the threshold value, PCMINTS.RFS is set. Larger RFTH value provides lower DMA/interrupt request frequency but have more risk to involve receive FIFO overflow. The optimum value is system dependent.</td><td>RW</td></tr>
<tr><td>4:1</td><td>TFTH</td><td>Transmit FIFO threshold for interrupt or DMA request. Determines when the TFS flags go active for the TXFIFO. When the sample number in transmit FIFO, indicated by PCMINTS.TFL, is less than the threshold value, PCMINTS.TFS is set. Smaller TFTH value provides lower DMA/interrupt request frequency but have more risk to involve transmit FIFO underflow. The optimum value is system</td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		dependent.	
0	PCMMOD	PCM mode select. 0: Master mode; 1: Slave mode.	RW

13.4.2.3 PCM FIFO DATA PORT REGISTER (PCMDP)

	PCMDP0																															BASE + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	DATA	FIFO port. When write to it, data is push to the transmit FIFO. When read from it, data is pop from the receiving FIFO.	RW

13.4.2.4 PCM INTERRUPT CONTROL REGISTER (PCMINTC)

	PCMINTCR0																															BASE + 0x00c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																												ETFS	ETUR	ERFS	EROR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:4	Reserved	Writing has no effect, read as zero.	R
3	ETFS	Enable TFS Interrupt. This bit is used to control the TFS interrupt enable or disable. **ETFS** — **TFS Interrupt** 0 — Disabled. 1 — Enabled.	RW
2	ETUR	Enable TUR Interrupt. This bit is used to control the TUR interrupt enable or disable. **ETUR** — **TUR Interrupt** 0 — Disabled. 1 — Enabled.	RW
1	ERFS	Enable RFS Interrupt. This bit is used to control the RFS interrupt	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		enable or disable. ERFS: 0 – Disabled.; 1 – Enabled.	
0	EROR	Enable ROR Interrupt. This bit is used to control the ROR interrupt enable or disable. EROR: 0 – Disabled.; 1 – Enabled.	RW

13.4.2.5 PCM INTERRUPT STATUS REGISTER (PCMINTS)

	PCMINTS0																															BASE + 0x010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																	RSTS	TFL					TFS	TUR	RFL					RFS	ROR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:15	Reserved	Writing has no effect, read as zero.	R
14	RSTS	Soft reset / flush state. 0: Nothing / reset or flush operation has completed 1: reset or flush operation has not completed	R
13:9	TFL	Transmit FIFO Level. The bits indicate the amount of valid PCM data in Transmit FIFO.	R
8	TFS	Transmit FIFO Service Request. This bit indicates that transmit FIFO level exceeds TFL threshold which is controlled by PCMCFG.TFTH When TFS is 1, it may trigger interrupt or DMA request depends on the interrupt enable and DMA setting. TFS: 0 – Transmit FIFO level exceeds TFL threshold.; 1 – Transmit FIFO level at or below TFL threshold.	RW
7	TUR	Transmit FIFO Under Run. This bit indicates that transmit FIFO has or has not experienced an under-run. TUR: 0 – When read, indicates under-run has not been found.; When write, clear itself.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			1	When read, indicates data has even been read from empty transmit FIFO.		
				When write, not effects.		
6:2	RFL	Receive FIFO Level. The bits indicate the amount of valid PCM data in Receive FIFO.				R
1	RFS	Receive FIFO Service Request. This bit indicates that receive FIFO level is or not below RFL threshold which is controlled by PCMCFG.RFTH. When RFS is 1, it may trigger interrupt or DMA request depends on the interrupt enable and DMA setting.				RW
			RFS	**Description**		
			0	Receive FIFO level below RFL threshold.		
			1	Receive FIFO level at or above RFL threshold.		
0	ROR	Receive FIFO Over Run. This bit indicates that receive FIFO has or has not experienced an overrun.				RW
			ROR	**Description**		
			0	When read, indicates over-run has not been found.		
				When write, clear itself.		
			1	When read, indicates data has even been written to full receive FIFO.		
				When write, not effects.		

13.4.2.6 PCM CLOCK DIVIDE REGISTER (PCMDIV)

	PCMDIV0																															**BASE + 0x014**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																SYNL					SYNDIV					CLKDIV					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:27	Reserved	Writing has no effect, read as zero.	R
16:11	SYNL	Controls the length that the PCMSYN based upon the PCMCLK. The length of PCMSYN = (SYNL + 1) * PCMCLK cycle.	RW
10:6	SYNDIV	Controls the frequency of the PCMSYN signal based upon the PCMCLK. PCMSYN = PCMCLK / 8 (SYNDIV + 1).	RW
5:0	CLKDIV	PCMCLK clock divider value minus 1. Controls the divider used to create the PCMCLK based upon the CPM_PCM_SYSCLK. PCMCLK = CPM_PCM_SYSCLK / (CLKDIV + 1).	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.5 PCM Interface Timing

The following figures show the timing relationship for the PCM transfers, Note in all cases. In master mode, the PCMCLK is derived from dividing the input clock, CPM_PCM_SYSCLK, and the PCMSYN is divided depended on the PCMCLK. In slave mode, the PCMCLK and PCMSYN are input from the external device. Data is sampled on the falling edge of the PCMCLK and sent out on the rising edge of the PCMCLK. The PCMSYN signal determines when the next data sample is to be transferred between the controller and the external device. Also, the PCMSYN signal as seen in the figure can be one bit time or a long bit time controlled by PCMDIV.SYNL. The PCMSYN frequency controlled by PCMDIV.SYNDIV is usually the sample rate. There are some variations controlled by PCMCFG.ISS, PCMCFG.OSS and PCMCFG.SLOT to accommodate 8 / 16bit sample sizes and multi-slot transmission.

13.5.1 Short Frame SYN

Figure 13-1 Short Frame SYN Timing (Shown with 16bit Sample)

NOTE: Figure 13-1 shows a PCM transfer with the MSB configured to be coincident with the PCMSYN.

Figure 13-2 Short Frame SYN Timing (Shown with 16bit Sample)

NOTE: Figure 13-2 shows a PCM transfer with the MSB configured one shift clock after the PCMSYN.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.5.2 Long Frame SYN

Figure 13-3 Long Frame SYN Timing (Shown with 16bit Sample)

NOTE: Figure 13-3 shows a PCM transfer with the MSB configured one shift clock after the PCMSYN.

Figure 13-4 Long Frame SYN Timing (Shown with 16bit Sample)

NOTE: Figure 13-4 shows a PCM transfer with the MSB configured to be coincident with the PCMSYN.

13.5.3 Multi-Slot Operation

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 13-5 Multi-Slot Frame SYN Timing (Shown with two Slots and 8bit Sample)

13.6 PCM Operation

13.6.1 PCM Initialization

At power-on or other hardware reset (WDT and etc), PCM is disabled. Software must initiate PCM after power-on or reset.

For further details, see the corresponding register description sections.

13.6.2 Audio Replay

Outgoing audio sample data is written to PCM transmit FIFO from processor via store instruction or from memory via DMA. PCM then takes the data from the FIFO, serializes it, and sends it over the serial wire PCMDOUT to an external DEVICE.

The audio transmission is enabled automatically when the PCM is enabled by set PCMCTL.PCMEN. And PCMCTL.ERPL must be 1. If PCMCTL.ERPL is 0, value of zero is sent to external DEVICE even if there are samples in transmit FIFO. At least one audio sample data in the transmit FIFO. If the transmit FIFO is empty, value of zero or last sample depends on AICFR.LSMP, is send to external DEVICE even if PCMCTL.ERPL is 1.

Here is the audio replay flow:

1 Configure the external DEVICE as needed.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 Initialize PCM and configure the register.
3 Write 1 to PCMCTL.PCMEN and PCMCTL.CLKEN.
4 Fill sample data to the transmit FIFO. Repeat this till finish all sample data. In this procedure, please control the FIFO to make sure no FIFO under-run and other errors happen. When the transmit FIFO under-run, noise or pause may be heard in the audio replay, PCMINTS.TUR is 1, and if PCMINTC.ETUR is 1, PCM issues an interrupt. Please reference to 13.6.4 for detail description on FIFO.
5 Write 1 to PCMCTL.ERPL. It is suggested that at least a frame of PCM data is pre-filled in the transmit FIFO to prevent FIFO under-run flag (PCMINTS.TUR).
6 Waiting for PCMINTS.TFL change to 0. So that all samples in the transmit FIFO has been replayed, then we can have a clean start and write 0 to PCMCTL.ERPL.

13.6.3 Audio Record

Incoming audio sample data is received from PCMDIN serially and converted to parallel word and stored in PCM receive FIFO. Then the data can be taken from the FIFO to processor via load instruction or to memory via DMA.

The audio recording is enabled automatically when the PCM is enabled by set PCMCTL.PCMEN, And PCMCTL.EREC must be 1. If PCMCTL.EREC is 0, the received data is discarded even if there are rooms in the receive FIFO. At least one room left in the receive FIFO. If the receive FIFO is full, the received data is discarded even if PCMCTL.EREC is 1.

Here is the audio record flow:

1 Configure the external DEVICE as needed.
   a Initialize PCM and configure the register.
   b Write 1 to PCMCTL.PCMEN and PCMCTL.CLKEN.
2 Write 1 to PCMCTL.EREC. Make sure there are rooms available in the receive FIFO before set PCMCTL.EREC. Usually, it should empty the receive FIFO by fetch data from it before set PCMCTL.EREC.
3 Take sample data form the receive FIFO. Repeat this till the audio finished. In this procedure, please control the FIFO to make sure no FIFO over-run and other errors happen. When the receive FIFO over-run, same recorded audio samples will be lost, PCMINTS.ROR is 1, and if PCMINTC.EROR is 1, PCM issues an interrupt. Please reference to 13.6.4 for detail description on FIFO.
4 Write 0 to AICCR.EREC.
5 Take sample data from the receive FIFO until PCMINTS.RFL change to 0. So that all samples in the receive FIFO has been taken away, then we can have a clean start up next time. When the receive FIFO is empty, read from it returns zero.

13.6.4 FIFOs operation

PCM has two FIFOs, one for transmitting and one for receiving. The FIFOs are in 16 bits width and

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16 entries depth, one entry for keep one sample regardless of the sample size. PCMDP.DATA provides the access point for processor/DMA to write to transmit FIFO and read from receive FIFO. One time access to PCMDP.DATA process one sample. The sample data should be put in LSB (Least Significant Bit) in memory or processor registers. For transmitting, bits exceed sample are discarded. For receiving, these bits are set to 0. Following figure illustrates the FIFOs access.

Figure 13-6 Transmitting/Receiving FIFO access via APB Bus

The software and bus initiator must guarantee the right sample placement at the bus.
In case of DMA bus initiator, One 16 bits sample occupies one 16-bits half word in memory, so 16-bits width DMA must be used. One 8-bits sample occupies one byte in memory, and use 8-bits width DMA.

13.6.5 Data Flow Control

There are three approaches provided to control/synchronize the incoming/outgoing data flow.

13.6.5.1 Polling and Processor Access

PCMINTS.RFL and PCMINTS.TFL reflect how many samples exist in receiving and transmitting FIFOs. Through read these register fields, processor can detect when there are samples in receiving FIFO and then load them from the RxFIFO, and when there are rooms in transmitting FIFO and then store samples to the TxFIFO.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Polling approach is in very low efficiency and is not recommended.

13.6.5.2 Interrupt and Processor Access

Set proper values to PCMCFG.TFTH and PCMCFG.RFTH, the FIFO interrupts trig thresholds. Set PCMINTC.ETFS and/or PCMINTC.ERFS to 1 to enable transmitting and/or receiving FIFO level trigger interrupts. When the interrupt found, it means there are rooms or samples in the TX or RX FIFO, and processor can store or load samples to or from the FIFO.

Interrupt approach is more efficient than polling approach.

13.6.5.3 DMA Access

To enable DMA operation, set PCMCTL.ERDMA and PCMCTL.ETDMA to 1 for transmit and receive respectively. It also needs to allocate two channels in DMA controller for data transmitting and receiving respectively. Please reference to DMAC spec for the details.

The PCMCFG.TFTH and PCMCFG.RFTH are used to set the transmitting and receiving FIFO level thresholds, which determine the issuing of DMA request to DMA controller. To respond the request, DMAC initiator and controls the data movement between memory and TX/RX FIFO.

13.6.6 PCM Serial Clocks and Sampling Frequencies

Figure 13-7 PCMCLK and PCMSYN generation scheme

13.6.7 Interrupts

The following status bits, if enabled, interrupt the processor:

- Receive FIFO Service (PCMINTS.RFS). It's also DMA Request.
- Transmit FIFO Service (PCMINTS.TFS). It's also DMA Request.
- Transmit Under-Run (PCMINTS.TUR).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Receive Over-Run (PCMINTS.ROR).

For further details, see the corresponding register description sections.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14 Internal CODEC Interface

14.1 Overview

This chapter describes the embedded audio CODEC in the processor and related software **interfaces.**

This embedded CODEC is an I2S audio CODEC. AIC module is an interface to this CODEC for audio data replaying and recording. Several memory mapped registers **in AIC** are used to access this embedded CODEC, **and the CODEC's internal control and configure registers can be accessed by write/read these memory mapped registers using 12 MHz clock.**

14.2 Features

The following are internal CODEC features:

- 24 bits ADC and DAC
- Headphone load up to 16 Ohm
- Sample frequency supported: 8k, 11.025k, 12k, 16k, 22.05k, 24k, 32k, 44.1k, 48k, and 96k
- Stereo line input
- DAC to HP path: Power consumption: 17.6mW, THD: -65dB @17.6mW /16Ohm
- DAC to stereo line output path @10kOhm: SNR: 100dB A-Weighted, THD: -80dB @FS-1dB
- Line input to ADC path: SNR: 90dB A-Weighted, THD: -80dB @FS-1dB
- Separate power-down modes for ADC and DAC path with several shutdown modes
- Reduction of audible glitches systems: Pop Reduction system, Soft Mute mode
- Output short circuit protection
- Embedded low noise Linear Regulator
- 3 MIC in path or 3lin in path Maximum (total input share to 2 path at same time)

TBD = parameter or document section to be defined later on
TBC = parameter or document section subject to change
TO BE COMPLETED = section to be filled or subject to change

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.2.1 Signal Descriptions

CODEC has **maximum** 13 analog signal IO pins and 4 power pins on chip. They are listed and described in the following table.

Table 14-1 CODEC signal IO pin description

Pin Names	IO	Pin Description	Power
AIP1, AIN1	AI	Single-ended or differential analog input.	AVDCDC
AIP2	AI	Single-ended analog input.	AVDCDC
AIP3	AI	Single-ended analog input.	AVDCDC
MICBIAS1	AO	Microphone bias.	AVDCDC
AOLOP	AO	Differential line output, positive pin.	AVDCDC
AOLON	AO	Differential line output, negative pin.	AVDCDC
AOHPL	AO	Left headphone single-ended analog output.	AVDCDC
AOHPR	AO	Right headphone single-ended analog output.	AVDCDC
VCAP	AO	Voltage Reference Output. An 10μF ceramic or tantalum capacitor in parallel with a 0.1μF ceramic capacitor attached from this pin to AVSCDC eliminates the effects of high frequency noise.	AVDCDC
VREFP	AO	Internal nLR output	AVDCDC
AVSHP	P	Headphone amplifier ground.	-
AVDCDC	P	CODEC analog power, 3.3V, internal nLR input.	-
AVSCDC	P	CODEC analog ground, inter signal VREFN.	-
HPSENSE	AI	Headphone jack sense.	AVDCDC
DMIC_IN	DI	Digital microphone data input pin.	AVDCDC
DMIC_CLK	DO	Digital microphone clock output pin.	AVDCDC

NOTES:

1 AVDCDC= 3.3v (typ).

2 Internal signal VREFN is connected to AVSCDC, Internal power supply AVS is connected to AVSCDC.

3 Please refer to data sheet of the chip for details.

4 DMIC_IN is ‘GPIO : PB18’ , MIC_CLK is ‘GPIO : PB19’. Please refer to GPIO specification for these pins operating.

5 nLR: Internal low noise linear regulator.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.2.2 Block Diagram

Figure 14-1 CODEC block diagram

Figure 14-2 Internal CODEC works with AIC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.2.3 Application schematic

Note: 1. The Rhpdo value is 470 Ohm, it use to prevent pop-up noise.

2. The single-ended/differential input port AIP1/AIN1 and single-ended input port AIP2/AIP3 can be configure to microphone input or line input by software.

3. VREFP/VCAP/VREFP each of them requires connecting decoupling capacitors (0.1uF) between the pads VREFP/VCAP/VREFP and AVSCDC. This ceramic capacitor has to be kept as close as possible to IC package (closer than 0.2 inch)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.3 Mapped Register Descriptions

The internal CODEC software interface includes 2 registers. They are mapped **to** IO memory address space of AIC module so that program can access them to control the operations of the CODEC.

Table 14-2 Internal CODEC Mapped Registers Description (AIC Registers)

Name	Description	RW	Reset value	Address	Size
RGADW	Address, data in and write command for accessing to internal registers of internal embedded CODEC.	RW	0x00000000	0x100200A4	32
RGDATA	**The data read out** and interrupt request status of Internal registers data in the internal embedded CODEC.	R	0x00000000	0x100200A8	32

NOTES:

1 All these registers are AIC Registers, because they are mapped **to** AIC IO memory address.

2 RGADW contains data, address and write command to the internal registers of the internal CODEC.

3 RGDATA returns the internal register value of the internal CODEC and interrupt request status.

14.3.1 CODEC internal register access control (RGADW)

RGADW contains address, data and write command to the internal registers of the internal embedded CODEC.

	RGADW																																0x100200A4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICRST	Reserved														RGWR	Reserved	RGADDR							RGDIN							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	ICRST	If set ‘1’ to this bit internal codec will reset and then clear to ‘0’ to end reset.	RW
30:17	Reserved	Writing has no effect, read as zero.	R
16	RGWR	**Writing 1 to this bit means issuing a write command to CODEC’s internal register. This bit keeps 1 by AIC** before the current write process is finished. **A register read or a new register write** cannot be issued before the previous write process is finished. In another word,	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		RGADW should not be written before RGADW.RGWR becomes 0. A write process takes up to the sum of 0.17us and 1 PCLK cycle. Writing 0 to this bit is ignored.	
15	Reserved	Writing has no effect, read as zero.	R
14:8	RGADDR	**When it issues a writing command to CODEC's internal register, i.e.RGWR=1, this field specifies the corresponding CODEC register's address. In addition, this field also decides the address of the CODEC register's data out (The data is mapped to RGDATA. RGDOUT) at any time.**	RW
7:0	RGDIN	**When it issues a write command to CODEC's internal register**, i.e. RGWR=1, this field contains the data to be written to the CODEC register.	RW

NOTES:

1 **It is strongly suggested verifying the data (using read RGDATA below) after writing the data to internal register of CODEC. When verifying RGDATA. RGDOUT right (RGDATA. RGDOUT is the return data from internal CODEC register of the corresponding address), the writing process is finished.**

2 Please notice that AIC needs SYS_CLK (**please refer** to AIC spec), when write new value to or read from CODEC internal registers.

14.3.2 CODEC internal register data output (RGDATA)

RGDATA returns the register value of the internal embedded CODEC and interrupt request status.

	RGDATA																															0x100200A8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserve																							IRQ	RGDOUT							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:9	Reserved	Writing has no effect, read as zero.	R
8	IRQ	This field returns the internal embedded CODEC's interrupt request. IRQ: 0 — No CODEC's interrupt request found. IRQ: 1 — CODEC's interrupt request is pending.	R
7:0	RGDOUT	This field returns the value of the register in internal embedded CODEC. (RGADW.RGADDR field specifies the CODEC register's address)	R

NOTE: AIC needs SYS_CLK (**please refer** to AIC spec), when write new value to or read from

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

CODEC internal registers.

14.4 Operation

The internal embedded CODEC is controlled **through** its internal registers. These registers **data** can be accessed through memory-mapped registers, RGADW and RGDATA. AIC's BITCLK and SYNC are **from** the CODEC and is controlled by CKCFG.SELAD register. The audio data transferring, i.e. audio replaying and recording, is done by AIC. AIC still **plays** the role of I2S controller. We will refer to many AIC operations and registers in the following audio operation descriptions, please reference to AIC spec for the details.

This is a guide for software.

14.4.1 Access to internal registers of the embedded CODEC

The embedded CODEC is controlled through its internal registers. RGADW and RGDATA are used to write to and read from these registers. Here are some examples.

Example 1. Write to a CODEC internal register.
Step 1: RGADW.RGWR == 0.
Step 2: If not, go to step 1.
Step 3: Write to RGADW and make it.
RGADW.RGDIN = <data to be written to the register>.
RGADW.RGADDR = <the register's address >.
Step 4: Write to RGADW to commit the writing operation.
RGADW.RGWR = 1.

Example 2. Read from a CODEC internal register.
Step 1: RGADW.RGWR == 0.
Step 2: If not, go to step 1.
Step 3: write to RGADW and make it.
RGADW.RGWR = 0.
RGADW.RGDIN = <don't care>.
RGADW.RGADDR = <the register's address>.
Step 4: read RGDATA.DOUT, which returns the register's content.

14.4.2 CODEC controlling and typical operations

This section is **about** some typical operations. We assume the power supply of CODEC is on, and CODEC is in STANDBY mode.

Before **performing** any of these operations, make sure AIC is configured properly as list below:

1 Make AIC to use internal CODEC mode:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

AICFR.ICDC = 1; Use internal CODEC.
AICFR.AUSEL = 1; Use I2S mode.
AICFR.BCKD = 0; CODEC input BIT_CLK to AIC.
AICFR.SYNCD = 0; CODEC input SYNC to AIC.
I2SCR.AMSL = 1; Use I2S operation mode.
I2SCR.ESCLK = 1; Open SYS_CLK to internal CODEC. (if using PLL Clock)

2 Make sure AICCR.FLUSH = 0; AICFR.RST = 0; AICCR.ENLBF = 0.
3 Clear AICSR.ROR, AICSR.TUR, AICSR.RFS, AICSR.TFS to 0.
4 Set proper value to AICCR.M2S; AICCR.ENDSW; AICCR.ASVTSU.
5 Set AICFR.ENB to 1; Open AIC.

When using DMA mode, configure AICFR.RFTH, AICCR.RDMS or AICFR.TFTH, AICCR.TDMS.
Configure TX-FIFO and interrupt means setting proper value to AICFR.TFTH, clear AICCR.ETFS to 0, and clear AICCR.ETUR to 0.
Configure RX-FIFO and interrupt means setting proper value to AICFR.RFTH, clear AICCR.ERFS to 0 and clear AICCR.EROR to 0.
When configure interrupt, software must handle all the interrupts. So all interrupts are recommended disabled as shown above.
CODEC shares the interrupt with AIC module.

The register or register bit of CODEC will use the same form as the Mapped registers, but software should use the method in the section "Mapped Register Descriptions" to access this registers.

More details are listed in the CODEC guide.

14.4.3 Power saving

There are many power modes in CODEC. In every working mode, it should close **unused** stages (parts) of CODEC for saving power.

The power diagram is shown in "CODEC Power Diagram"; please refer to "CODEC Operating modes".

14.4.4 Pop noise and the reduction of it

Please refer to "Ramping system note" and "Anti-pop operation sequences" for details.

14.4.4.1 Reference open step

1 Init play.

Step 0: Open DMA and two AIC modules Clocks in CPM.CLKGR.
Step 1: Configure AIC as slave and using internal CODEC mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

AICFR.ICDC = 1; Use internal CODEC.

AICFR.AUSEL = 1; Use I2S mode.

AICFR.BCKD = 0; CODEC input BIT_CLK to AIC.

AICFR.SYNCD = 0; CODEC input SYNC to AIC.

I2SCR.AMSL = 1; Use I2S operation mode.

I2SCR.ESCLK = 1; Open SYS_CLK to internal CODEC.

Step 2: Configure DMA as slave mode using internal CODEC.

2 Open.

Step 0: Enable DMA Channel Clock.

Step 1: Configure AIC sample size and sample rate. Configure AIC Output FIFO Threshold.

Step 2: Configure DMA.

Step 3: Configure CODEC.

3 Write.

Step 0: Enable DMA Channel Clock.

Step 1: Configure AIC.

Step 2: Configure DMA.

Step 3: Configure CODEC.

4 Read.

Step 0: Enable DMA Channel Clock.

Step 1: Configure AIC.

Step 2: Configure DMA.

Step 3: Configure CODEC.

5 Close.

6 End.

NOTES:

1 SB_DAC Control the internal OBIT_CLK from CODEC to AIC, First turn it on when write data (replay).

2 SB_ADC Control the internal IBIT_CLK from CODEC to AIC, First turns it on when read data (record).

14.5 Timing parameters

Parameter	Condition	Min.	Typ.	Max.	Unit
Tsbyu	Cext = 10uF/100nF +/-20%		250	600	ms
Tshd_adc	Cext = 10uF/100nF +/-20%		200		ms
Tshd_dac	Cext = 10uF/100nF +/-20%		400	900	ms
Tr, Tf (all inputs)	All modes			5	ns
Tr, Tf (all outputs)	All modes			5	ns

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

1 Tsbyu is the reference wake-up time after complete power down.

2 Tshd_adc is the ADC wake-up time after sleep mode.

3 Tshd_dac is DAC wake-up time after sleep mode.

14.6 C parameters

Voltages:

AVSCDC is connected to analog ground.

AVDCDC= 3.3V (typ).

power consumption

mode	power consumption	Unit
Sleep mode	1200	uW
Playback stereo audio DAC only (capacitor coupled load configuration)	8.3	mW
Record mic stereo input only (audio ADC)	10.3	mW
Record mic mono input only (audio ADC)	6.3	mW
Bypass path (audio stereo line in to HP)	7.7	mW

Current value is at AVDCDC = 3.3 V.

Chip pin Name	MAX Current across I/O @ AVDCDC = 3.3 V
AVDCDC	< 200 mA in normal working mode
AVSCDC	< 20 mA in normal working mode
AVSHP	< 160 mA in normal working mode
	< 1400 mA in case of short circuit
VCAP	< 2 mA in normal working mode
AIP1,AIN1	< 2 mA in normal working mode
AIP2,AIN2	< 2 mA in normal working mode
AIP3,AIN3	< 2 mA in normal working mode
AIP4,AIN4	< 2 mA in normal working mode
MICBIAS1	< 5 mA in normal working mode
MICBIAS2	< 5 mA in normal working mode
AOHPL	< 80 mA in normal working mode
	< 1200 mA in case of short circuit
AOHPR	< 80 mA in normal working mode
	< 1200 mA in case of short circuit
AOLOP,AOLON	< 1 mA in normal working mode
HPSENSE	< 1 mA in normal working mode

The current in case of short circuit is the max value. This current is only sink or drawn until the short circuit detection system acts.

Please refer to Chip Datasheet for more details.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7 CODEC internal Registers

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

14.7.1 Registers Memory Map

Register Name	Function	Address	Reset value	Access Size
SR	Status Register	0x00	-	8
SR2	Status Register 2	0x01	-	8
MR	Mode status register	0x07	-	8
AICR_DAC	DAC Audio Interface Control Register	0x08	hD3	8
AICR_ADC	ADC Audio Interface Control Register	0x09	hD3	8
CR_LO	Differential line-out Control Register	0xB	h90	8
CR_HP	HeadPhone Control Register	0xD	hB0	8
CR_DMIC	Digital Microphone register	0x10	h00	8
CR_MIC1	Microphone1 Control register	0x11	hB0	8
CR_MIC2	Microphone2 Control register	0x12	h30	8
CR_LI1	Control Register for line1 inputs (for bypass function)	0x13	h10	8
CR_LI2	Control Register for line2 inputs (for bypass function)	0x14	h10	8
CR_DAC	DAC Control Register	0x17	h90	8
CR_ADC	ADC Control Register	0x18	h90	8
CR_MIX	Digital Mixer Control Register	0x19	h00	8
DR_MIX	Digital Mixer Data Register	0x1A	h00	8
CR_VIC	Control Register for the CODEC	0x1B	h03	8

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

CR_CK	Clock Control Register	0x1C	h00	8
FCR_DAC	DAC Frequency Control Register	0x1D	h00	8
SFCCR_DAC	Sample frequency coarse control register	0x1E	h00	8
SFFCR_DAC	Sample frequency fine control register	0x1F	h00	8
FCR_ADC	ADC Frequency Control Register	0x20	h00	8
CR_TIMER_MSB	MSB of programmable counter	0x21	h00	8
CR_TIMER_LSB	LSB of programmable counter	0x22	h00	8
ICR	Interrupt Control Register	0x23	h00	8
IMR	Interrupt Mask Register	0x24	hFF	8
IFR	Interrupt Flag Register	0x25	-	8
IMR2	Interrupt Mask Register 2	0x26	hFF	8
IFR2	Interrupt Flag Register 2	0x27	-	8
GCR_HPL	Left channel headphone Control Gain Register	0x28	h06	8
GCR_HPR	Right channel headphone Control Gain Register	0x29	h06	8
GCR_LIBYL	Left channel bypass line Control Gain Register	0x2A	h06	8
GCR_LIBYR	Right channel bypass line Control Gain Register	0x2B	h06	8
GCR_DACL	Left channel DAC Gain Control Register	0x2C	h00	8
GCR_DACR	Right channel DAC Gain Control Register	0x2D	h00	8
GCR_MIC1	Microphone 1 Gain Control Register	0x2E	h00	8
GCR_MIC2	Microphone 2 Gain Control Register	0x2F	h00	8
GCR_ADCL	Left ADC Gain Control Register	0x30	h00	8
GCR_ADCR	Right ADC Gain Control Register	0x31	h00	8
GCR_MIXDACL	DAC Digital Mixer Control Register	0x34	h00	8
GCR_MIXDACR	DAC Digital Mixer Control Register	0x35	h00	8
GCR_MIXADCL	ADC Digital Mixer Control Register	0x36	h00	8
GCR_MIXADCR	ADC Digital Mixer Control Register	0x37	h00	8
CR_ADC_AGC	Automatic Gain Control Register	0x3A	h00	8
DR_ADC_AGC	Automatic Gain Control Data Register	0x3B	h00	8

14.7.2 Register Description

14.7.2.1 SR: Status Register

Register Name: SR				Register Address: 0x0			
bit7-R-?	bit6-R-?	bit5-R-?	bit4-R-?	bit3-R-?	bit2-R-?	bit1-R-?	bit0-R-?
PON_ACK	IRQ_ACK	JACK	DAC_LOCKED	Reserved			

Bits	Field	Description
7	PON_ACK	Acknowledge status bit after power on. Read 0: reset value

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: codec is ready to operate
6	IRQ_ACK	Acknowledge status bit after IRQ sending. Read 0: reset value 1: codec has requested an interrupt (IRQ signal activated)
5	JACK	Output Jack plug detection status. Read 0: no jack 1: output jack present
4	DAC_LOCKED	Acknowledge status bit after Sample Rate Converter system is locked Read 0: SRC is not locked. Data from the audio interface are automatically muted. 1: The SRC is locked and operating normally.
3:0	Reserved	Writing has no effect, read as zero.

14.7.2.2 SR2: Status Register 2

Register Name: SR2				Register Address: 0x1			
bit7-R-?	bit6-R-?	bit5-R-?	bit4-R-?	bit3-R-?	bit2-R-?	bit1-R-?	bit0-R-?
Reserved			DAC_UNKNOWN_FS	Reserved			

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4	DAC_UNKOWN_FS	Read 0: Measured mean frequency of incoming audio data is within supported values 1: Measured mean frequency of incoming audio data is not within supported values
3:0	Reserved	Writing has no effect, read as zero.

14.7.2.3 MR: Status Register

Register Name: MR				Register Address: 0x7			
bit7-R-?	bit6-R-?	bit5-R-?	bit4-R-?	bit3-R-?	bit2-R-?	bit1-R-?	bit0-R-?
Reserved		ADC_MODE_UPD	ADC_MUTE_STATE		DAC_MODE_UPD	DAC_MUTE_STATE	

Bits	Field	Description
7:6	Reserved	Writing has no effect, read as zero.
5	ADC_MODE_UPD	Read 0: ADC operation mode is the programmed mode

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: detection of an operation mode update on the ADC
4:3	JADC_MUTE_ STATE	Read 00 : ADC not muted 01 : ADC being muted 10 : ADC leaving mute mode 11 : ADC in mute mode
2	DAC_MODE_UPD	Read 0: DAC operation mode is the programmed mode 1: detection of an operation mode update on the DAC
1:0	DAC_MUTE_ STATE	Read 00 : DAC not muted 01 : DAC being muted 10 : DAC leaving mute mode 11 : DAC in mute mode

14.7.2.4 AICR_DAC: Audio Interface Control Register

Register Name: AICR_DAC				Register Address: 0x8			
bit7-RW-1	bit6-RW-1	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-1	bit0-RW-1
DAC_ADWL		DAC_SLAVE	SB_AICR_DAC	Reserved		DAC_AUDIOIF	

Bits	Field	Description
7:6	DAC_ADWL	Audio Data Word Length for DAC path. Read / Write 00: 16-bit word length data 01: 18-bit word length data 10: 20-bit word length data 11: 24-bit word length data
5	SB_AICR_DAC	Read/Write: 0 = Audio interface in master mode 1 = Audio interface in slave mode
4	SB_AICR_DAC	Read/Write: 0 = DAC audio interface active 1 = DAC audio interface in power-down mode
3:2	Reserved	Writing has no effect, read as zero.
1:0	DAC_AUDIOIF	Audio Interface mode selection for DAC Read/Write: 00 = Parallel interface 01 = Left-justified interface 10 = DSP interface 11= I2S interface

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

1 **DAC_AUDIOIF should be configured to 11**

14.7.2.5 AICR_ADC: Audio Interface Control Register

Register Name: AICR_ADC				Register Address: 0x9			
bit7-RW-1	bit6-RW-1	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-1	bit0-RW-1
ADC_ADWL		Reserved	SB_AICR_ADC	Reserved		ADC_AUDIOIF	

Bits	Field	Description
7:6	ADC_ADWL	Audio Data Word Length for ADC path. Read / Write 00: 16-bit word length data 01: 18-bit word length data 10: 20-bit word length data 11: 24-bit word length data
5	Reserved	Writing has no effect, read as zero.
4	SB_AICR_ADC	Read/Write: 0 = ADC audio interface active 1 = ADC audio interface in power-down mode
3:2	Reserved	Writing has no effect, read as zero.
1:0	ADC_AUDIOIF	Audio Interface mode selection for ADC Read/Write: 00 = Parallel interface 01 = I2S interface 10 = DSP interface 11 = I2S interface

NOTES:

1 **ADC_AUDIOIF should be configured to 11**

14.7.2.6 CR_LO: differential line-out Control Register

Register Name: CR_LO				Register Address: 0xB			
bit7-RW-1	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
LO_MUTE	Reserved		SB_LO	Reserved	LO_SEL		

Bits	Field	Description
7	LO_MUTE	differential line out mute mode. Read/Write

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: mute inactive, Signal applied to line output 1: no signal on line output
6:5	Reserved	Writing has no effect, read as zero.
4	SB_LO	differential line out conditioning circuitry power-down mode. Read/Write 0: active 1: power-down
3:2	Reserved	Writing has no effect, read as zero.
1:0	LO_SEL	differential line-output Amplifier input selection Read/Write 000: DACL 001: DACR 010: DACL + DACR 011: reserved for future use 100: AILatt 101: AIRatt 110: AILatt + AIRatt 111: reserved for future use

14.7.2.7 CR_HP: HeadPhone Control Register

Register Name: CR_HP				Register Address: 0xD			
bit7-RW-1	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
HP_MUTE	Reserved		SB_HP	Reserved	HP_SEL		

Bits	Field	Description
7	HP_MUTE	HeadPhone output signal disabled. Read/Write 0: Signal applied to headphone outputs 1: no signal on headphone outputs, acts as a mute signal
6:5	Reserved	Writing has no effect, read as zero.
4	SB_HP	headphone output stage power-down mode. Read/Write 0: headphone output stage is active 1: power-down
3	Reserved	Writing has no effect, read as zero.
2:0	HP_SEL	Headphone Output Amplifier input selection Read/Write HP_SEL | HPL input | HPR input 000 | DACL | DACR 001 | DACL | DACL

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		010	AILatt	AIRatt
		011	AILatt	AILatt
		100	AIRatt	AIRatt
		others	reserved	reserved

14.7.2.8 CR_DMIC: Digital Microphone Control Register

Register Name: CR_DMIC				Register Address: 0x10			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
DMIC_CLKON	Reserved						

Bits	Field	Description
7	DMIC_CLKON	Digital microphone clock (DMIC_CLK) enable Read/Write 0= clock off 1= clock on, clock frequency varies with DMIC_RATE and MCLK
6:0	Reserved	Writing has no effect, read as zero.

14.7.2.9 CR_MIC1: Control Register for microphone1 inputs

Register Name: CR_MIC1				Register Address: 0x11			
bit7-RW-1	bit6-RW-0	bit5-RW-1	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
MICSTEREO	MICIDFF1	SB_MICBIAS1	SB_MIC1	Reserved			MIC1_SEL

Bits	Field	Description
7	MICSTEREO	Microphone input mode selection Read/Write 0 = Microphone mono inputs 1 = Microphone stereo inputs This signal affects IN_SEL
6	MICIDFF1	Microphone 1 input mode selection Read/Write 0= single-ended input 1= differential input
5	SB_MICBIAS1	Microphone 1 biasing buffer power-down Read/Write 0 = active 1 = power-down

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4	SB_MIC1	Microphone 1 power-down Read/Write 0 = active 1 = power-down
3:1	Reserved	Writing has no effect, read as zero.
0	MIC1_SEL	Microphone 1 input selection Read/Write 0 = selection of AIP1/AIN1 1 = selection of AIP2

14.7.2.10 CR_MIC2: Control Register for microphone2 inputs

Register Name: CR_MIC2				Register Address: 0x12			
bit7-RW-0	bit6-RW-0	bit5-RW-1	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved	MICIDFF2	SB_MICBIAS2	SB_MIC2	Reserved			MIC2_SEL

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	MICIDFF2	Microphone 2 input mode selection Read/Write 0= single-ended input 1= Reserved
5	SB_MICBIAS2	Microphone 2 biasing buffer power-down Read/Write 0 = active 1 = power-down
4	SB_MIC2	Microphone 2 power-down Read/Write 0 = active 1 = power-down
3:1	Reserved	Writing has no effect, read as zero.
0	MIC2_SEL	Microphone 2 input selection Read/Write 0 = selection of AIP3 1 = Reserved

14.7.2.11 CR_LI1: Control Register for line1 inputs (for bypass function)

Register Name: CR_LI1				Register Address: 0x13			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-1
Reserved	LIN1DIFF	Reserved	SB_LIBY1	Reserved			LI1_SEL

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	LIN1DIF	line 1 input mode selection Read/Write 0 = single-ended input 1 = differential input
5	Reserved	Writing has no effect, read as zero.
4	SB_LIBY1	Line 1 input power-down Read/Write 0 = active 1 = power-down
3:1	Reserved	Writing has no effect, read as zero.
0	LI1_SEL	Line 1 input selection Read/Write 0= selection of AIP1/AIN1 1= selection of AIP2

14.7.2.12 CR_LI2: Control Register for line2 inputs (for bypass function)

Register Name: CR_LI2				Register Address: 0x14			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-1
Reserved	LIN2DIFF	Reserved	SB_LIBY2	Reserved			LI2_SEL

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	LIN2DIF	line 2 input mode selection Read/Write 0 = single-ended input 1 = Reserved
5	Reserved	Writing has no effect, read as zero.
4	SB_LIBY2	Line 2 input power-down Read/Write 0 = active 1 = power-down
3:1	Reserved	Writing has no effect, read as zero.
0	LI2_SEL	Line 2 input selection Read/Write 0= selection of AIP3 1= Reserved

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.13 CR_DAC: Control Register for DAC

Register Name: CR_DAC				Register Address: 0x17			
bit7-RW-1	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
DAC_MUTE	Reserved	DAC_LEFT_ONLY	SB_DAC	Reserved			

Bits	Field	Description
7	DAC_MUTE	DAC soft mute mode. Read/Write 0: mute inactive, digital input signal transmitted to the DAC 1: puts the DAC in soft mute mode
6	Reserved	Writing has no effect, read as zero.
5	DAC_LEFT_ONLY	Left data only are considered. Read/Write 0: DAC right channel active 1: DAC left data are used for left and right channel **To avoid any audible pop, it is required to put the DAC in soft mute mode before modifying the DAC_LEFT_ONLY bit.**
4	SB_DAC	DAC power-down mode. Read/Write 0: active 1: power-down
3:0	Reserved	Writing has no effect, read as zero.

14.7.2.14 CR_ADC: Control Register for ADC

Register Name: CR_ADC				Register Address: 0x18			
bit7-RW-1	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
ADC_MUTE	ADC_DMIC_SEL	ADC_LEFT_ONLY	SB_ADC	Reserved		ADC_IN_SEL	

Bits	Field	Description
7	ADC_MUTE	ADC soft mute mode Read/Write 0 = mute inactive 1 = puts the ADC in soft mute mode
6	ADC_DMIC_SEL	digital filter input selection Read/Write 0= ADC 1= Digital microphone
5	ADC_LEFT_ONLY	Deactivation of ADC right channel

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Read/Write: 0= ADC right channel active 1= ADC right channel inactive
4	SB_ADC	ADC power down mode. Read/Write 0: active 1: power-down
3:2	Reserved	Writing has no effect, read as zero.
1:0	ADC_IN_SEL	selection of the signal converted by the ADC Read/Write: If MICSTEREO = 0 00 = Analog input 1 input to left and right channels (CODEC automatically considers that ADC_LEFT_ONLY equals '1' to optimize power consumption) 01 = Analog input 2 input to left and right channels (CODEC automatically considers that ADC_LEFT_ONLY equals '1' to optimize power consumption) 10 = Reserved for further use 11 = Reserved for further use Note that when digital microphone is selected, the ADC_LEFT_ONLY is not automatically modified. If MICSTEREO = 1 00 = Analog input 1 input to left channel and Analog input 2 input to right channel 01 = Analog input 2 input to left channel and Analog input 1 input to right channel 10 = Reserved for further use 11 = Reserved for test

NOTE: When ADC right channel is deactivated, right channel output is set to 0

14.7.2.15 CR_MIX: Control Register for digital mixer

Register Name: CR_MIX				Register Address: 0x19			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
MIX_EN	MIX_LOAD	MIX_ADD					

Bits	Field	Description
7	MIX_EN	digital mixer enable Read/Write 0: digital mixer off 1: digital mixer enabled
6	MIX_LOAD	Read/Write

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Refer to DR_MIX for MIX_LOAD description
1:0	DAC_MIX	Digital mixer control registers address Read/Write **MIX_ADD** / **Corresponding control register** 000000 / MIX_0 000001 / MIX_1 000010 / MIX_2 000011 / MIX_3

14.7.2.16 DR_MIX: Data Register for digital mixer

Register Name: CR_MIX				Register Address: 0x1A			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
MIX_DATA							

Bits	Field	Description
7:0	MIX_DATA	● Instructions for writing into a register The data must firstly be written into the DR_MIX register. The register address must then be written in CR_MIX (MIX_ADD[5:0]). The data is written into the register when MIX_LOAD is set to '1'. ● Instructions for reading the values in a register The address of the register to be read is written in CR_MIX (MIX_ADD[5:0]). The content of the corresponding register appears in DR_MIX when MIX_LOAD is set to '0'.

MIX_0: Digital mixer control register 0

Register Name: CR_MIX				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
AIDACL_SEL		AIDACR_SEL		Reserved			DAC_MIX

Bits	Field	Description
7:4	AIDACX_SEL	DAC input selection Read/Write 00: cross inputs 01: normal inputs 10: mixed inputs 11: 0 inputs
3:1	Reserved	Writing has no effect, read as zero.
0	DAC_MIX	Mixer mode on DAC Path

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Read/Write 0: Playback DAC only 1: Playback DAC + ADC

Note: AIDACX_SEL should be configured to 01 in normal mode.

MIX_1: Digital mixer control register 1

Register Name: CR_MIX				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
MIXDACL_SEL		MIXDACR_SEL		Reserved			

Bits	Field	Description
7:4	MIXDACX_SEL	Mixer input selection on DAC path Read/Write 00: normal inputs 01: cross inputs 10: mixed inputs 11: 0 inputs
3:0	Reserved	Writing has no effect, read as zero.

MIX_2: Digital mixer control register 2

Register Name: MIX_2				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
AIADCL_SEL		AIADCR_SEL		Reserved			MIX_REC

Bits	Field	Description
7:4	AIADCX_SEL	Read/Write 00: normal inputs 01: cross inputs 10: mixed inputs 11: 0 inputs
3:1	Reserved	Writing has no effect, read as zero.
0	MIX_REC	Mixer mode on ADC Path Read/Write 0: Record input only 1: Record input + DAC

MIX_3: Digital mixer control register 3

Register Name: CR_MIX				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
MIXADCL_SEL		MIXADCR_SEL		Reserved			

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Field	Description
7:4	MIXADCX_SEL	Read/Write 00: cross inputs 01: normal inputs 10: mixed inputs 11: 0 inputs
3:0	Reserved	Writing has no effect, read as zero.

Note: MIXADCX_SEL should be configured to 01 in normal mode.

14.7.2.17 CR_VIC: Control Register for the codec

Register Name: CR_VIC				Register Address: 0x1B			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-1	bit0-RW-1
Reserved						SB_SLEEP	SB

Bits	Field	Description
7:2	Reserved	Writing has no effect, read as zero.
1	SB_SLEEP	sleep mode. Read/Write 0: normal mode (active) 1: sleep mode
0	SB	complete power-down mode. Read/Write 0: normal mode (active) 1: complete power-down

14.7.2.18 CR_CK: Clock Control Register

Register Name: CR_CK				Register Address: 0x1C			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			SHUTDOWN_ CLOCK	CRYSTAL			

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4	SHUTDOWN_ CLOCK	Read/Write 0= normal mode (MCLK active) 1= MCLK is turned off (stuck to ‘0’)
3:0	CRYSTAL	Selection of the MCLK frequency Read/Write

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		The sampling frequency value is given in the CRYSTAL[3:0] table **CRYSTAL[3:0]** — **Master Clock Frequency** 0000 — 12 MHz 0001 — Reserved for further use 0010 — 13 MHz …. — Reserved for further use 1111 — Reserved for further use

NOTE: CRYSTAL should be configured to **0x0** for setting the internal 12MHz master clock SYS_CLK (default).

14.7.2.19 FCR_DAC: DAC Frequency Control Register

Register Name: FCR_DAC				Register Address: 0x1D			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved				DAC_FREQ			

Bits	Field	Description
7:4	Reserved	Writing has no effect, read as zero.
3:0	DAC_FREQ	Selection of the DAC sampling rate (Fs). Read/Write The sampling frequency value is given in the FREQ table.

NOTE: Please refer to section Sample frequency: FREQ.

14.7.2.20 SFCCR_DAC: SRC frequency adjustment register 1

Register Name: SFCCR_DAC				Register Address: 0x1D			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
DACFREQ_ VALID	DAC_FREQ_ADJ[14:8]						

Bits	Field	Description
7:4	DACFREQ_VALID	Enables the sampling frequency adjustment (DAC_FREQ_ADJ[14:0]) in master mode. Bits DAC_FREQ_ADJ[14:0] are split in both registers SFCCR_DAC and SFFCR_DAC.
3:0	DAC_FREQ_ADJ [14:8]	Frequency Tuning Control Word. DAC_FREQ_ADJ allows a fine tuning of DAC audio data sampling frequency in master mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.21 SFFCR_DAC: SRC frequency adjustment register 2

Register Name: SFFCR_DAC				Register Address: 0x1D			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
DAC_FREQ_ADJ[7:0]							

Bits	Field	Description
3:0	DAC_FREQ_ADJ [7:0]	Frequency Tuning Control Word. DAC_FREQ_ADJ allows a fine tuning of DAC audio data sampling frequency in master mode.

14.7.2.22 FCR_ADC: ADC Frequency Control Register

Register Name: FCR_ADC				Register Address: 0x20			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved	ADC_HPF	ADC_WNF		ADC_FREQ			

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	ADC_HPF	ADC High Pass Filter enable. Read/Write 0: inactive 1: enable the ADC High Pass Filter
5:4	ADC_WNF	ADC Wind Noise Filter frequency 00= inactive 01= enables the ADC Wind Noise Filter mode 1 10= enables the ADC Wind Noise Filter mode 2 11= enables the ADC Wind Noise Filter mode 3
3:0	ADC_FREQ	Selection of the ADC sampling rate (Fs). Read/Write The sampling frequency value is given in the FREQ table.

NOTE: Please refer to section Sample frequency: FREQ.

14.7.2.23 CR_TIMER_MSB: counter MSB

Register Name: CR_TIMER_MSB				Register Address: 0x21			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
COUNT[15:8]							

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Field	Description
7:0	COUNT[15:8]	MSB of counter programmable number. The counter programmable number register is split into both registers CR_TIMER_MSB and CR_TIMER_LSB.

14.7.2.24 CR_TIMER_LSB: counter LSB

Register Name: CR_TIMER_LSB				Register Address: 0x22			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
COUNT[7:0]							

Bits	Field	Description
7:0	COUNT[7:0]	LSB of counter programmable number The positive number decreases of ‘1’ every slow clock period (256 MCLK periods) and generates an IRQ (TIMER_END) when going from h0001 to h0000.

14.7.2.25 ICR: Interrupt Control Register

Register Name: ICR				Register Address: 0x23			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
INT_FORM		Reserved					

Bits	Field	Description
7:6	INT_FORM	Waveform and polarity of the IRQ signal Read/Write 00: The generated IRQ is a high level 01: The generated IRQ is a low level 10: The generated IRQ is a high level pulse with an 8 MC_CLK cycles duration when using 8-bit parallel control interface or 8 MCLK cycles duration when using SMB control interface 11: The generated IRQ is a low level pulse with an 8 MC_CLK cycles duration when using 8-bit parallel control interface or 8 MCLK cycles duration when using SMB control interface
5:0	Reserved	Writing has no effect, read as zero.

NOTE: INT_FORM should be configured to 00

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.26 IMR: Interrupt Mask Register

Register Name: IMR				Register Address: 0x24			
bit7-RW-1	bit6-RW-1	bit5-RW-1	bit4-RW-1	bit3-RW-1	bit2-RW-1	bit1-RW-1	bit0-RW-1
LOCK_MASK	SCLR_MASK	JACK_MASK	Reserved		ADC_MUTE_ MASK	DAC_MODE_ MASK	DAC_MUTE_ MASK

Bits	Field	Description
7	LOCK_MASK	Mask for the LOCK_EVENT flag Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)
6	SCLR_MASK	Mask for the SCLR flag. Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)
5	JACK_MASK	Mask for the JACK_EVENT flag. 0: interrupt enabled 1: interrupt masked (no IRQ generation)
4:3	Reserved	Writing has no effect, read as zero.
2	ADC_MUTE_ MASK	Mask for the ADC_MUTE_EVENT flag Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)
1	DAC_MODE_ MASK	Mask for the DAC_MODE_EVENT flag Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)
0	DAC_MUTE_ MASK	Mask for the DAC_MUTE_EVENT flag Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)

NOTES:

1 When an interrupt is masked, the event do not generates any change on the IRQ signal, but the corresponding flag value is set to ‘1’ in the IFR register.

2 When the IRQ signal is active on level, the IRQ signal is set to the inactive level while the bits IFR[7:0] & (!IMR[7:0]) equals ‘0’.

3 When the IRQ signal is a pulse, the IRQ signal is set to the inactive state until a new non-masked event occurs in IFR[7:0] or until a masked event is unmasked.

4 If the 8-bit parallel control interface is selected, MC_CLK must not be stopped in order to propagate IRQ signal.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.27 IFR: Interrupt Flag Register

Register Name: IFR				Register Address: 0x25			
bit7-RWC-0	bit6-RWC-0	bit5-RWC-0	bit4-R-0	bit3-R-0	bit2-RWC-0	bit1-RWC-0	bit0-RWC-0
LOCK_EVENT	SCLR	JACK_EVENT	Reserved		RDO	GUP	GDO

Bits	Field	Description
7	LOCK_EVENT	Lock status Read 0 = no event 1 = event detected Write 0 = no effect 1 = Reset of the flag
6	SCLR	Left or Right Output short circuit detection status. Read 0 : no event 1 : indicates that a short-circuit has been detected. Write 0 = no effect 1 = Reset of the flag
5	JACK_EVENT	Event on output Jack plug detection status. Read 0: no event 1: event detected (due to JACK flag change to ‘0’ or ‘1’). Write 0 = no effect 1 = Reset of the flag
4:3	Reserved	Writing has no effect, read as zero.
2	ADC_MUTE_ EVENT	Read 0 = no event 1 = indicates the beginning or the end of a soft mute on ADC data Write 0 = no effect 1 = Reset of the flag
1	DAC_MODE_ EVENT	Read 0 = no event 1 = event detected (due to DAC_MODE_UPD flag change to ‘0’ or ‘1’) Write 0 = no effect 1 = Reset of the flag
0	DAC_MUTE_	Read

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EVENT	0 = no event 1 = indicates the beginning or the end of a soft mute on DAC data Write 0 = no effect 1 = Reset of the flag

NOTE: The flags RUP, RDO, GUP and GDO can be changed after 4 cycles of SYS_CLK from codec being reset.

14.7.2.28 IMR2: Interrupt Mask Register 2

Register Name: IMR2				Register Address: 0x26			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			TIMER_END_MASK	Reserved			

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4	TIMER_END_MASK	Mask for the TIMER_END flag Read/Write 0: interrupt enabled 1: interrupt masked (no IRQ generation)
3:0	Reserved	Writing has no effect, read as zero.

14.7.2.29 IFR2: Interrupt Mask Register 2

Register Name: IFR2				Register Address: 0x27			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RWC-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			TIMER_END	Reserved			

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4	TIMER_END	Timer status Read 0 = no event 1 = event detected Write 0 = no effect 1 = reset of the flag
3:0	Reserved	Writing has no effect, read as zero.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.30 GCR_HPL: left channel headphone Gain Control Register

Register Name: GCR_HPL				Register Address: 0x28			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-0
LRGO	Reserved		GOL				

Bits	Field	Description
7	LRGO	HP amplifier gain coupling. Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6:5	Reserved	Writing has no effect, read as zero.
4:0	GOL	Left channel HP amplifier gain programming value.

NOTE: Please refer to section "Programmable attenuation: GO" for more details.

14.7.2.31 GCR_HPR: right channel headphone Gain Control Register

Register Name: GCR_HPR				Register Address: 0x29			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-0
Reserved			GOR				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GOR	Right channel HP amplifier gain programming value.

NOTE: Please refer to section "Programmable attenuation: GO" for more details.

14.7.2.32 GCR_LIBYL: left channel bypass line Gain Control Register

Register Name: GCR_LIBYL				Register Address: 0x2A			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-0
LRGI	Reserved		GIL				

Bits	Field	Description
7	LRGI	analog bypass gain coupling. Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6:5	Reserved	Writing has no effect, read as zero.
4:0	GIL	Left channel Line in gain programming value.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: Please refer to section "Programmable Bypass path attenuation: GI" for more details.

14.7.2.33 GCR_LIBYR: right channel bypass line Gain Control Register

Register Name: GCR_LIBYR				Register Address: 0x2B			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-0
Reserved			GIR				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GIR	Right channel Line in gain programming value.

NOTE: Please refer to section "Programmable Bypass path attenuation: GI" for more details.

14.7.2.34 GCR_DACL: Left channel DAC Gain Control Register

Register Name: GCR_DACL				Register Address: 0x2C			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
LRGOD	Reserved		GODL				

Bits	Field	Description
7	LRGOD	DAC digital gain coupling. Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6:5	Reserved	Writing has no effect, read as zero.
4:0	GODL	Left channel DAC digital gain programming value.

NOTE: Please refer to section "Programmable digital attenuation: GOD" for more details.

14.7.2.35 GCR_DACR: right channel DAC Gain Control Register

Register Name: GCR_DACR				Register Address: 0x2D			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			GODR				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GODR	Right channel DAC digital gain programming value.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: Please refer to section "Programmable digital attenuation: GOD" for more details.

14.7.2.36 GCR_MIC1: Microphone 1 Gain Control Register

Register Name: GCR_MIC1				Register Address: 0x2E			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved					GIM1		

Bits	Field	Description
7:3	Reserved	Writing has no effect, read as zero.
2:0	GIM1	Microphone 1 boost stage gain programming value.

NOTE: Please refer to section "Programmable boost gain: GIM".

14.7.2.37 GCR_MIC2: Microphone 2 Gain Control Register

Register Name: GCR_MIC2				Register Address: 0x2F			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved					GIM2		

Bits	Field	Description
7:3	Reserved	Writing has no effect, read as zero.
2:0	GIM2	Microphone 2 boost stage gain programming value.

NOTE: Please refer to section "Programmable boost gain: GIM".

14.7.2.38 GCR_ADCL: Left channel ADC Gain Control Register

Register Name: GCR_ADCL				Register Address: 0x30			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
LRGID	Reserved	GIDL					

Bits	Field	Description
7	LRGID	ADC digital gain coupling. Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6	Reserved	Writing has no effect, read as zero.
5:0	GIDL	Left channel ADC digital gain programming value.

NOTE: Please refer to the section "Programmable input attenuation amplifier: GID".

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.39 GCR_ADCR: Right channel ADC Gain Control Register

Register Name: GCR_ADCR				Register Address: 0x31			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved		GIDR					

Bits	Field	Description
7:6	Reserved	Writing has no effect, read as zero.
5:0	GIDR	Right channel ADC digital gain programming value.

NOTE: Please refer to the section "Programmable input attenuation amplifier: GID".

14.7.2.40 GCR_MIXDACL: DAC Left Digital Mixer Control Register

Register Name: GCR_MIXDACL				Register Address: 0x34			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
LRGOMIX	Reserved		GOMIXL				

Bits	Field	Description
7	GOMIXL	Mixer gain coupling for DAC path Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6:5	Reserved	Writing has no effect, read as zero.
4:0	GOMIXL	Mixer gain for DAC path. Read/Write 00000 : 0dB 00001 : -1dB …by step of 1dB 11111 : -31dB

14.7.2.41 GCR_MIXDACR: DAC Right Digital Mixer Control Register

Register Name: GCR_MIXDACR				Register Address: 0x35			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			GOMIXR				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4:0	GOMIXR	Mixer gain for DAC path. Read/Write 00000 : 0dB 00001 : -1dB …by step of 1dB 11111 : -31dB

14.7.2.42 GCR_MIXADCL: ADC Left Digital Mixer Control Register

Register Name: GCR_MIXADCL				Register Address: 0x36			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
LRGIMIX	Reserved		GIMIXL				

Bits	Field	Description
7	LRGIMIX	Mixer gain coupling for input path Read/Write 0: Left and right channels gains are independent 1: Left and right channels gain track left channel gain
6:5	Reserved	Writing has no effect, read as zero.
4:0	GIMIXL	Mixer gain for input path. Read/Write 00000 : 0dB 00001 : -1dB … by step of 1dB 11111 : -31dB

14.7.2.43 GCR_MIXADCR: ADC Right Digital Mixer Control Register

Register Name: GCR_MIXADCR				Register Address: 0x37			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			GIMIXR				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GIMIXR	Mixer gain for input path. Read/Write 00000 : 0dB 00001 : -1dB … by step of 1dB 11111 : -31dB

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.7.2.44 CR_ADC_AGC: Automatic Gain Control Register

Register Name: CR_ADC_AGC				Register Address: 0x3A			
bit7-RW-0	bit6-RW-0	bit5-RW-1	bit4-RW-1	bit3-RW-0	bit2-RW-1	bit1-RW-0	bit0-RW-0
AGC_EN	ADC_AGC_LOAD	ADC_AGC_LOAD					

Bits	Field	Description
7	AGC_EN	Enable of the AGC system. Read/Write 0 : inactive 1 : enable the automatic level control
6	ADC_AGC_LOAD	Read/Write Refer to DR_ADC_AGC for ADC_AGC_LOAD description
5:0	ADC_AGC_ADD	Address of AGC control registers Read/Write:

ADC_AGC_ADD[5:0]	Corresponding control register
000000	ADC_AGC_0
000001	ADC_AGC _1
000010	ADC_AGC _2
000011	ADC_AGC _3
000100	ADC_AGC _4

NOTE: Please refer to section "AGC system guide" for more details.

14.7.2.45 DR_ADC_AGC: Automatic Gain Control Data Register

Register Name: DR_ADC_AGC				Register Address: 0x3B			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-1
ADC_AGC_DATA							

Bits	Field	Description
7:0	ADC_AGC_DATA	data loaded in AGC control registers Read/Write ● Instructions for writing into a register The data must firstly be written into the DR_ADC_AGC register. The register address must then be written in CR_ADC_AGC (ADC_AGC_ADD[5:0]). The data is written into the register when ADC_AGC_LOAD is set to '1'. ● Instructions for reading the values of a register The address of the register to be read is written in CR_ADC_AGC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		(ADC_AGC_ADD[5:0]). The content of the corresponding register appears in DR_ADC_AGC when ADC_AGC_LOAD is set to '0'.

ADC_AGC_0: AGC control register 0

Register Name: **ADC_AGC_0**				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved	AGC_STEREO	TARGET				Reserved	

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	AGC_STEREO	selection of the AGC system Read/Write 0 =same gain applied to Left and Right channel 1 = different gains applied to Left and Right channel
5:1	TARGET	Target output level of the ADC Read/Write: 0000 = -6 dB 0001 = -7.5 dB … by step of 1.5 dB 1111 = - 28.5 dB
0	Reserved	Writing has no effect, read as zero.

ADC_AGC_1: Automatic Gain Control Register 2

Register Name: **ADC_AGC_1**				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
NG_EN	NG_THR			HOLD			

Bits	Field	Description
7	NG_EN	Selection of the Noise Gate system. Read/Write 0: inactive 1: enables the noise gate system
6:4	NG_THR	Noise Gate Threshold value. Input level (dB) < Noise Gate Level (dB). Read/Write 000: -72 dB 001: -66 dB …by step of 6dB 111: -30 dB

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3:0	HOLD	Hold time before starting AGC adjustment to the TARGET value. Read/Write 0000: 0ms 0001: 2 ms 0010: 4 ms … Time Step x2 1111: 32.768s

NOTE: Please refer to section "AGC system guide" for more details.

ADC_AGC_2: AGC control register 2

Register Name: **ADC_AGC_2**				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
ATK				DCY			

Bits	Field	Description
7:4	ATK	Attack Time - Gain Ramp Down. Read/Write 0000: 32 ms 0001: 64 ms …by step of 32 ms 1111: 512 ms
3:0	DCY	Decay Time - Gain Ramp up. Read/Write 0000: 32 ms 0001: 64 ms …by step of 32 ms 1111: 512 ms

NOTES:

1 DCY and ATK registers values are delays between each step of gain.

2 Please refer to section "AGC system guide" for more details.

ADC_AGC_3: AGC control register 3

Register Name: **ADC_AGC_3**				Register Address:			
Bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			AGC_MAX				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	AGC_MAX	Maximum Gain Value to apply to the ADC path.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

1 Please refer below table for AGC_MAX setup.

2 Please refer to section "AGC system guide" for more details.

AGC_MAX	Gain Value	AGC_MAX	Gain Value	AGC_MAX	Gain Value	AGC_MAX	Gain Value
00000	0	01000	12	10000	23	11000	32
00001	1.5	01001	13.5	10001	23	11001	33.5
00010	3	01010	15	10010	23	11010	35
00011	4.5	01011	16.5	10011	24.5	11011	36.5
00100	6	01100	18	10100	26	11100	38
00101	7.5	01101	19.5	10101	27.5	11101	39.5
00110	9	01110	21	10110	29	11110	41
00111	10.5	01111	22.5	10111	30.5	11111	42.5

ADC_AGC_4: AGC control register 4

Register Name: **ADC_AGC_4**				Register Address:			
bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved			AGC_MIN				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	AGC_MIN	Maximum Gain Value to apply to the ADC path.

NOTES:

1 Please refer to below table for AGC_MIN setup.

AGC_MIN	Gain Value	AGC_ MIN	Gain Value	AGC_ MIN	Gain Value	AGC_ MIN	Gain Value
00000	0	01000	12	10000	23	11000	32
00001	1.5	01001	13.5	10001	23	11001	33.5
00010	3	01010	15	10010	23	11010	35
00011	4.5	01011	16.5	10011	24.5	11011	36.5
00100	6	01100	18	10100	26	11100	38
00101	7.5	01101	19.5	10101	27.5	11101	39.5
00110	9	01110	21	10110	29	11110	41
00111	10.5	01111	22.5	10111	30.5	11111	42.5

2 Please refer to section "AGC system guide" for more details.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.8 Programmable gains

This section helps you to configure the programmable gain amplifier in the CODEC.
Internal signal VREFP is connected to AVDCDC Pin and internal signal VREFN is connected to AVSCDC Pin.

In this section, VREF equals to (VREFP – VREFN).

14.8.1 Programmable boost gain: GIM

The following table gives the relation between the gain and the input level for the microphone input amplifier **when GID = 0.**

GIM[2]	GIM[1]	GIM[0]	Gain value (dB)	Maximum input amplitude (Vp-p differential) (FS)
0	0	0	0	0.85*VREF
0	0	1	4	0.536*VREF
0	1	0	8	0.338*VREF
0	1	1	12	0.213*VREF
1	0	0	16	0.134*VREF
1	0	1	20	0.085*VREF
1	1	0	20	0.085*VREF
1	1	1	20	0.085*VREF

NOTES:

1 **Analog input amplitude value is not more than 'Maximum input amplitude' for Vp-p differential.**

2 This maximum analog input amplitude is referenced as Full Scale (FS). After conversion, the corresponding digital code of the output value varies from 0x7FFF down to 0x8000 for a 16-bit word. When the analog input amplitude value is greater than FS, the dynamic characteristics are not guaranteed.

3 When a change occurs on GIDi inputs, data are valid on the digital output after about 64 sample periods. If the HPF is activated, data are valid after about 64 sample periods but the offset cancellation is not still completed at this time due to its internal time constant.

14.8.2 Programmable input gain amplifier: GID

The digital gain of ADC path may be programmed through the registers bits GIDL and GIDR.
The value of the gain is programmable from 0 to 23dB with a pitch of 1dB.

The gain and input levels are obtained according to the following table:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

GID						Decimal decoded value	Gain value (dB)	Maximum input amplitude (Vp-p. Differential) (FS)
0	0	0	0	0	0	0	0	0.85*VREF
0	0	0	0	0	1	1	1	0.757*VREF
0	0	0	0	1	0	2	2	0.6021*VREF
…								
x	y	z	t	u	v	i	i	0.85/{10^(i/20)}*VREF
…								
0	1	0	1	1	1	23	23	0.06*VREF
0	1	1	0	0	0	24	24	0.06*VREF
…								
1	0	1	0	1	0	43	43	0.006*VREF

NOTE: The last column of the table shows the maximum analog input to **be applied to** the MICi inputs. The value is given in Vp-p differential. These values refer to the external voltage reference VREF **equal** to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

14.8.3 Programmable digital attenuation: GOD

The attenuation of DAC output amplifier may be programmed independently for the both channels through the registers bits GODL and GODR.

The value of the gain GODL/R is programmable from +0 to –31dB with 1 dB pitch. The gain and output levels are obtained according to the following table:

GOD					Decimal decoded value	Gain Value (dB)
0	0	0	0	0	0	0
0	0	0	0	1	1	-1
					…	
0	0	1	1	0	6	-6
					…	
1	1	1	1	0	30	-30
1	1	1	1	1	31	-31

14.8.4 Programmable attenuation: GO

The attenuation of Headphone output amplifier may be programmed independently for the both

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

channels through the registers bits GOL and GOR.

The value of the gain GOL/R is programmable from +6 to –25dB with 1 dB pitch. The gain and output levels are obtained according to the following table:

GO					Decimal decoded value	Gain Value (dB)	Maximal PGAT input amplitude (Vp-p)	Maximal PGAT output amplitude (Vp-p)
0	0	0	0	0	0	+6	0.425*VREF	0.85*VREF
0	0	0	0	1	1	+5	0.478*VREF	0.85*VREF
					…		…	…
0	0	1	0	1	5	+1	0.757*VREF	0.85*VREF
0	0	1	1	0	6	0	0.85*VREF	0.85*VREF
0	0	1	1	1	7	-1	0.85*VREF	0.757*VREF
					…		…	…
1	1	1	1	0	30	-24	0.85*VREF	0.054*VREF
1	1	1	1	1	31	-25	0.85*VREF	0.048*VREF

NOTES:

1 When headphone driver is loaded by a 16 Ohm load, setting GOL/R = 0 is possible. However, set GOL/R to 9 at the maximum to preserve dynamic performances. The output stage is sized to support a 70mA current at most.

2 The last column of the table shows the analog output voltage in the condition of a digital input at FS (Full Scale). The value is given in Vp-p single-ended.

3 These values refer to the external voltage reference VREF equal to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

14.8.5 Programmable Bypass path attenuation: GI

The analog input gain may be programmed through GIL/R.

The value of the gain is programmable from +6 to -25dB with a pitch of 1dB. The gain and input levels are obtained according to the following table:

GI					Decimal decoded value	Gain value (dB)	Maximum input amplitude (Vp-p) (FS)
0	0	0	0	0	0	+6	0.425*VREF
0	0	0	0	1	1	+5	0.478*VREF
0	0	0	1	0	2	+4	0.536*VREF

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

x	y	z	t	u	i	6 - i	0.85/{10^((6 - i)/20)}*VREF
0	0	1	1	0	6	0	0.85*VREF
					…		0.85*VREF
1	1	1	1	1	31	-25	0.85*VREF

The last column of the table gives the maximum analog input **voltage**. The value is given in Vp-p. These values refer to the external voltage reference VREF **equal** to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

14.8.6 Programmable digital mixer gain: GIMIX and GOMIX

The following table shows the relation between the gain and GIMIX/GOMIX.

GIMIX or GOMIX	Gain value (dB)
00000	0
00001	-1
00010	-2
00011	-3
…	…
11101	-29
11110	-30
11111	-31

14.8.7 Gain refresh strategy

GI* and GO* gains are controlled through the control interface. To avoid sound artifacts, the gain increases or decreases each time the gain stage output crosses the zero value. **Tcrossout** time-out counter forces the gain to be updated if a zero crossing event doesn't occur. After each gain step, zero crossing events are ignored during at least **Tcrossmin**.

In case that the gain coupling between both left and right channels is active (LRGi different from RLGi), gain stepping of each channel is independent from the other depending on zero crossing event occurrence.

The duration of Tcrossout and Tcrossmin are given below:

MCLK (MHz)	Tcrossout (ms)	Tcrossmin (ms)
12	21.8	0.171
13	20.2	0.158

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.9 Configuration of the headphone output stage

In cap-coupled connection, codec uses the LOAD register bit to control the ramping duration. Inappropriate setting will lead to a too long or too fast ramping and will create audio artifacts.

To prevent pop-up noise generation due to floating nodes when no load is plugged in the jack connector, it is required to add some resistor devices that act as pull down function (named Rhpl and Rhpr in section “Headphone connnection” and section “Required external components”).

Its value has to be determined as following:

Working Mode	Load resistor and bypass capacitor values	LOAD value	Rhpdo value
Driving Headphone	16 Ohm / 220uF	0	470 Ohm typ.
Driving Lineout	10k Ohm / 1uF	1	4.7k Ohm max.

14.10 Out-of-band noise filtering

An internal analog Low Pass Filter at the DAC output is designed to remove the out-of-band noise generated by the delta sigma modulation (Noise Shaper). The internal LPF reduces the amount of energy contained in the wide band part (> 24 kHz) of the output signal. The out-of-band noise, when not removed, can be damageable in some high quality applications.

This filter is always working and does not need configure.

14.11 Output short-circuit protection (headphone output)

Analog short-circuit protection in the output stage has been implemented to prevent excessive current from flowing through AOHPL, AOHPR output pins. This prevents the output stage from over-heating.

The system detects the following **case**:

- Abnormal headphone load.

14.11.1 Indication of the short circuit detection

When such an overload is detected on one of AOHPL, AOHPR output pins,

- An interrupt is sent on the IRQ pin and the SCMC flag in the IFR register is set to ‘1’.
- Internally to codec:

  Automatic power-down of the 2 output amplifiers (AOHPL, AOHPR signals) when a short-circuit is detected on AOHPL or AOHPR pin (SCMC flag is set to ‘1’).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.11.2 Reset of short circuit detection

The following sequence has to be to apply:

1 Mask the interrupt by writing ‘1’ in the Interrupt Control Register.

2 Handle the cause of short-circuit according to the events presented in following paragraphs (Capacitor-coupled headphone connection).

3 Update the short-circuit flag by writing ‘1’ in the Interrupt Flag Register.

4 Check the reset of flag by reading the Interrupt Flag Register. The bit must be equal to ‘0’. If it remains at '1', that means that short-circuit is not resolved.

5 Enable the interrupt by writing ‘0’ in the Interrupt Control Register.

14.11.3 Capacitor-coupled headphone connection

It is up to the application to put the output stage in power down mode (SB_HP = ‘1’), to put codec in sleep or complete power-down mode, to reset it.

The short-circuit will be solved by the following events:

– Removal of the inserted jack. (needs the use of HPSENSE pins)
– Reset of codec. (NRST signal)
– Putting the output stage in power-down mode. (SB_HP=1)
– Putting codec in sleep mode. (SB_SLEEP=1)
– Putting codec in complete power-down mode. (SB=1)

14.12 Sampling frequency: FREQ

The sampling frequency value is given in the FREQ table below.

FREQ[3:0]	Sampling Rate (Fs)
0000	8 kHz
0001	11.025 kHz
0010	12 kHz
0011	16 kHz
0100	22.05 kHz
0101	24 kHz
0110	32 kHz
0111	44.1 kHz
1000	48 kHz
1001	88.2 kHz
1010	96 kHz
1011	176.4 kHz
1100	192 kHz

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1101	Reserved
1110	Reserved
1111	Reserved

14.13 Programmable data word length

The Data Word Length block (DWL) allows selecting the length of the input data and of the output data between 24-/20-/18-/16-bit **to AICR_DAC. DAC_ADWL and AICR_ADC.ADC_ADWL** (respectively for the DAC and ADC paths) in accordance with the following table:

ADWL	Word length
0 0	16-bit word length data
0 1	18-bit word length data
1 0	20-bit word length data
1 1	24-bit word length data

The size of the buses is always 24 bits, but the input/output data only use the number of MSB programmed with ADWL. The LSB are considered as ‘0’ in input and set to ‘0’ in output.

The capability to use a data word length of 16 bits is kept for compatibility with standard applications.

14.14 Ramping system note

An internal mechanism is used to reduce output glitches when the headphone stage enters or leaves the power-down mode.

When the SB_HP is set to ‘1’, the headphone output voltages (AOHPL, AOHPR) are slowly decreased in the same time from /2 down to 0.

When the SB_HP is set to ‘0’, the headphone output voltages (AOHPL, AOHPR) are slowly increased in the same time from 0 to VREFP/2.

After power supplies ramp up, the CODEC starts its internal initialization sequence and SR. PON_ACK register is changed when the initialization sequence is complete.
An interrupt request is sent when the ramp completes.

Do not change the level of SB_HP as long as the **initialization** sequence is not complete. If the change occurs, the success of the **initialization** sequence is not **guaranteed.**
In order to prevent audible glitch, it is required to power-down the output stage (SB_HP=1) before changing the output load with CR1.LOAD.

Please refer to “Anti-pop operation sequences” for details.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.15 AGC system guide

For the microphone input to ADC path, an Automatic Gain Control (AGC) system allows to optimize the signal swing at the input of the ADC.

The AGC circuit compares the output of the ADC to a level and increases or decreases the gain of the microphone preamplifier and the digital gain to compensate. The full dynamic range of the ADC can be used automatically if the audio from the microphone is to be output digitally through the ADC.

The AGC_EN register bit enables the AGC system, in this case IN_SEL must be equal to “00” or “01”.

AGC Block Diagram:

The AGC system is used at the MIC input.
The HPF filter characteristics: Cut Frequency =300 Hz.
In the in AGC mode, the system of gain control will directly assign the values of the gains GIDL, GIDR and GIM1 (or GIM2).

14.15.1 AGC operating mode

TARGET sets the desired ADC output range level. The AGC system adapts the gain stages (GID and GIM) in order to best reach this target. AGC_MAX and AGC_MIN fix the limits of the gain variation.
Please refer to “CODEC Operating modes” for the AGC System diagram in the “CODEC Power Diagram”.

In order that the AGC system should not alter the dynamic content of the signal (voice “tonic” for instance) by continuously adapting the gain to fit the target level, the time between two consecutives gain adjustments is modifiable by the HOLD register value.

After this delay:

– If the output level is lower than TARGET, the gain is increased step by step in accordance to the DCY register value.

– If the output level is higher than TARGET, the gain is decreased step by step in accordance

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

to the ATK register value.

The following figure illustrates the behavior of AGC system:

Figure 14-3 AGC adjusting waves

A noise-gating feature, enabled by the NG_EN register bit, prevents gain increases when no signal or small signal is present at the input. The noise gate threshold is set by the NG_THR register value. The following graph shows a more detailed application.

The following graph summarizes the operations and shows more details.

Figure 14-4 AGC adjust areas

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The areas from A to E are **different working areas** of AGC system, which is listing below:

A: If the signal level is in this critical area: the AGC system decreases quickly the gain at the input of the ADC until the signal goes under the critical threshold.

B: If the signal level remains in this area after the HOLD delay: the AGC system decreases the gain at the input of the ADC until the signal reaches the target area with a slope defined by AGC3.ATK register value.

C: If the signal level is in this area: the AGC system does not perform gain adjustment.

D: If the signal level remains in this area after the HOLD delay: the AGC system increases gain at the input of the ADC until the signal reach the target area with a slope defined by AGC3.DCY register value.

E: If the signal level is in this range: the AGC system considers the signal as noise and does not perform gain adjustment.

14.16 Digital Mixer description

CODEC includes a digital mixer which provides a loopback of the ADC output to the DAC and Headphone output and a loopback of the mixer output to the record path.

Two gains GIMIX[4:0] and GOMIX[4:0] are added to each input of the mixer to adapt the amplitude of the mixed signal. A zero-crossing detection is included on each gain stage to minimize the zipper noise.

When ADC_LEFT_ONLY = '1', the right output of stereo ADC is set to '0' (ADC_INR = '0').

A digital multiplexer allows choosing between the ADC signal and the mixer output signal on the

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

record path.

AIDAC*SEL, AIADC*SEL, MIXDAC*SEL and MIXADC*SEL functionality are applied between the Digital Mixer and the Data interface. The behavior is the following:

AIDACLSEL1	AIDACLSEL0	I_ DAC_INL	AIDACRSEL1	AIDACRSEL0	I_ DAC_INR
0	0	DAC_INR	0	0	DAC_INL
0	1	DAC_INL	0	1	DAC_INR
1	0	(DAC_INL + DAC_INR)/2	1	0	(DAC_INL + DAC_INR)/2
1	1	0	1	1	0

AIADCLSEL1	AIADCLSEL0	DOL	AIADCRSEL1	AIADCRSEL0	DOR
0	0	I_DOR	0	0	I_DOL
0	1	I_DOL	0	1	I_DOR
1	0	(I_DOL+I_DOR)/2	1	0	(I_DOL+I_DOR)/2
1	1	0	1	1	0

MIXDACLSEL1	MIXDACLSEL 0	I_DILMIX	MIXDACRSEL1	MIXDACRSEL0	I_DIRMIX
0	0	DIL	0	0	DIR
0	1	DIR	0	1	DIL
1	0	(DIL+DIR)/2	1	0	(DIL+DIR)/2
1	1	0	1	1	0

MIXADCLSEL1	MIXADCLSEL 0	I_ADC_INL	MIXADCRSEL1	MIXADCRSEL 0	I_ADC_INR
0	0	ADC_INR	0	0	ADC_INL
0	1	ADC_INL	0	1	ADC_INR
1	0	(ADC_INL+ADC_INR)/2	1	0	(ADC_INL+ADC_INR)/2
1	1	0	1	1	0

14.17 Digital microphone interface

CODEC accepts bitstream from digital microphone and converts it into audio data at the sample rate (Fs) selected in FCR_ADC register. CODEC provides a clock (DMIC_CLK) and receives data on DMIC_IN at the same frequency. DMIC_CLK frequency depends on MCLK frequency selection in CCR register.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 14-5 Digital microphone interface connection

After conversion, the corresponding digital code of the output value varies from 0x7FFF down to 0x8000 for a 16-bit word, coded in 2's complement.

CODEC can receive simultaneously data from two digital microphones.

14.17.1 Timing Diagram

Figure 14-6 Digital microphone timing diagram at MCLK = 12 MHz

(DMIC_CLK = 3 MHz) and MCLK = 13 MHz (DMIC_CLK = 3.25 MHz)

14.17.2 Timings

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Parameter	Symbol	Min	Typ	Max	Unit
DMIC_CLK frequency	F_{dmic_clk}	3	-	3.25	MHz
DMIC_CLK duty cycle	D_{dmic_clk}	0.4	0.5	0.6	-
DMIC_IN setup time	T_{dmic_setup}	T_{MCLK} + 10	-	-	ns
DMIC_IN hold time	T_{dmic_hold}	0	-	-	ns

14.17.3 Noise template (TBC)

Figure 14-7 Digital microphone modulation noise reference spectrum

(with FFT resolution = 20 Hz and 7 terms Blackman-Harris windowing)

If the noise at the digital microphone output is higher than the reference in the [20 Hz – 20 kHz] bandwidth, the SNR will be limited by the digital microphone in-band noise.

If the noise at the digital microphone output is higher than the reference for frequencies beyond 20 kHz, the SNR will be limited by the aliasing of the digital microphone quantization noise.

14.18 CODEC I2S interface DAC part slave mode.

The mode is used for eliminated the frequency skew when internal codec transfer with other audio interface, such as PCM. When this mode, need set AICFR.CDC_SLAVE = 1'b1 \ AICFR.ICDC = 1'b0 \ AICFR.BCKD = 1'b1 \ AICFR.SYNCD = 1'b1 \ AICR_DAC.DAC_SLAVE = 1'b1. Set AIC and PCM clock from the same source, such as the same PLL.
Note: CODEC I2S interface ADC part always in master mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.19 CODEC Operating modes

Different operating modes are available:

– Power-up mode: During power on time, CODEC is in this mode.

– Reset mode: When NRST is low, CODEC is in this mode.

– Soft mute mode: When DAC_MUTE is 1, CODEC is in this mode.

– Complete Power-down mode: After RESET, CODEC is in this mode.

– SLEEP modes: When SB_SLEEP is 1, CODEC is in this mode.

– Normal mode: When CODEC is not in above mode, it is in this mode. This mode has three modes: RECORD mode, REPLAY mode, RECORD_REPLAY mode.

The power diagram is shown below.

Figure 14-8 CODEC Power Diagram

There are many power parts of CODEC. Any part could be powered down independently.

14.19.1 Power-On mode and Power-Off mode

When the power supply ramps up, CODEC enters the power-on mode. During the reset, the CODEC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

is put in stand-by in order to reduce audible pops.

The CODEC doesn't handle the power supply ramp down on itself. The software has to turn the CODEC in complete stand-by mode before the power supply starts to ramp down.

14.19.2 RESET mode

The reset input signal is asynchronous; the reset minimum duration is one SYS_CLK cycle.
During the power-up mode and system reset, the CODEC goes into Reset mode.
After system reset the CODEC will exit Reset mode and go to STANDBY mode.

NOTES:

1 Except during the power-up mode, do NOT perform any reset in order to avoid audible pops.
2 Resetting the CODEC during normal operating mode will turn instantaneously the CODEC in STANDBY mode. This will lead to generate a large audible pop.

14.19.3 STANDBY mode

CODEC goes to STANDBY mode when the SB register bit equals 1, and all functions including ADC path, DAC path and analog references will stop and whole CODEC is shutdown for saving power. CODEC is complete down in this mode.

During the STANDBY mode, the power consumption is reduced to a minimum, so it is also called Complete Power-Down mode. When SB is set to '0', CODEC leaves the STANDBY mode. It is necessary to wait some time before the CODEC references settle. This time is called Tsbyu. When CODEC reference settled, it is in SLEEP mode.

Please refer to the section "Timing parameters" for the Tsbyu Value.

14.19.4 SLEEP mode

When SB_SLEEP is set to 1, CODEC enters in sleep mode. The logical part and the analog functions, except the voltage and biasing references, enter in power-down mode. So, the power consumption is reduced without penalizing the start-up time.

When SB_SLEEP falls, CODEC leaves the corresponding STANDBY mode; it is necessary to wait some time before the CODEC reaches the normal mode. Depending on the selected mode, this time is either called Tshd_adc (SB_ADC=0) for the ADC path or Tshd_dac (SB_DAC=0) for the DAC path.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Please refer to the section "Timing parameters" for the Tshd_adc and Tshd_dac Value.

14.19.5 Initial all the gain

When CODEC turn into STANDBY mode, need initial all the gain:

- Firstly set a gain that different you wished (such as '1') to GIM1 / GIM2 / GIDL / GIDR / GIL / GIR / GOL / GOR / GODL / GODR.
- Secondly wait about 10ms and than set a gain you wished (such as '0') to GIM1 / GIM2 / GIDL / GIDR / GIL / GIR / GOL / GOR / GODL / GODR.

After the step, you can change the gain you wished at any times

14.19.6 Soft Mute mode

Soft Mute mode is used in order to reduce audible parasites when before the DAC enters or after leaves the Normal mode. Set the DAC_MUTE register bit to 1, it will go to Soft Mute mode.

Set DAC_MUTE to 1 puts the DAC in Soft Mute mode. The CODEC decreases progressively the digital gain from 0dB to -∞. When the gain down sequence is completed, the signal of the DAC is equal to 0 whatever the value of the digital input data is. **The CODEC sets IFR.GDO register bit to 1, then generates an interrupt if IMR.GDO_MASK is 0.**

During Soft Mute mode, the DAC is still converting but the output final voltages (**AOLOP, AOHP**) are equal to VREF/2, so the differential of the Headphone voltage is zero that cause no sound output.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 14-9 DAC Gain up and gain down sequence

In the opposite, when DAC_MUTE is set to 0, the DAC leaves the Soft Mute mode by increasing

Figure 14-10 ADC Gain up and gain down sequence

progressively the digital gain from -∞ to 0dB. When the gain up sequence is completed, the DAC returns in Normal mode. **The CODEC sets IFR.GUP register bit to 1, then generates an interrupt if IMR.GUP_MASK is 0.**

After exiting Soft Mute mode, the DAC output will flow the DAC input data, and there is sound in the Headphone.

The duration of gain down and gain up sequences are nearly independent of Fs and is about 23ms.

NOTES:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1 **Do NOT change the value of DAC_MUTE while the effect of the previous change is not reached, or the working is not guaranteed.**
2 **Do NOT enter in stand-by mode while the gain sequence is not completed, or the working is not guaranteed.**

14.19.7 Power-Down mode and ACTIVE mode

Twelve stand-by inputs allow putting independently the different parts of CODEC into Power-Down mode.

When all SB_*=1 except SB=0 and SB_SLEEP=0, the CODEC is in ACTIVE mode, it's ready for play sound or record sound. But still need follow the anti-pop start or stop sequence. Please refer to "Start up sequence" and "Shutdown sequence".

14.19.8 Working modes summary

Different working modes are sum-up in the following table (non exhaustive table):

Mode	SB	SB_SLEEP	SB_DAC	SB_HP	SB_LO	SB_ADC	SB_MICBIAS1	SB_MICBIAS2	SB_MIC1	SB_MIC2	SB_LIBY1	SBLIBY2	ADC_IN_SEL	HP_SEL	LO_SEL	MICSTEREO	DAC_LEFT_ONLY	ADC_LEFT_ONLY	DAC_MUTE	HP_MUTE	LO_MUTE
Reset mode (complete power-down mode)	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1	1
Complete power-down mode	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sleep mode	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Mono Record Mode, Line input to ADC	0	0	X	X	X	0	X	X	0 1	10	X	X	00 01	X	X	0	X	1	X	X	X
Stereo Record Mode, Line input to ADC	0	0	X	X	X	0	X	X	0	0	X	X	00	X	X	1	X	0	X	X	X
Stereo Playback mode, DAC to HP	0	0	0	0	X	X	X	X	X	X	X	X	X	000	X	X	0	X	0	0	X
Stereo Playback mode, DAC to LO	0	0	0	X	0	X	X	X	X	X	X	X	X	X	010	X	0	X	0	X	0
Bypass mode, stereo Line to HP	0	0	X	0	X	X	X	X	X	X	0	0	X	010	XX	X	X	X	X	0	X

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.20 SYS_CLK turn-off and turn-on

The main clock of CODEC is called SYS_CLK, which is generated in CPM module and called MCLK. During the SLEEP mode and the complete power-down mode, the main clock SYS_CLK may be stopped to reduce the power consumption to the leakage currents only. In other modes, the main clock SYS_CLK must not be stopped.

The main clock SYS_CLK must not be stopped until CODEC has reached the complete power-down mode and must be restarted before leaving the power-down mode.

After SYS_CLK restarts, it is required to wait 4 SYS_CLK cycles before reading or writing the registers.

When SYS_CLK is turned off (SB_SLEEP=1 or SB=1), writing on register values are not taken into account, register values are not up to date when read and interrupts not generated until SYS_CLK turns on.

14.21 Requirements on outputs and inputs selection and power-down modes

The following rules must be respected in order not to damage performances and to keep the functionality:

- If SB_MIC1 is set to 1, MICSTEREO must be equal to 0, IN_SEL, HP_SEL and LO_SEL must not be equal to ‘00’.
- If SB_MIC2 is set to 1, MICSTEREO must be equal to 0, IN_SEL, HP_SEL and LO_SEL must not be equal to ‘01’.
- If SB_LINE is set to 1, IN_SEL must not be equal to ‘10’.
- If SB_LIBY is set to 1, HP_SEL and LO_SEL must not be equal to ‘10’.
- If SB_DAC is set to 1, HP_SEL and LO_SEL must not be equal to ‘11’.

14.22 Anti-pop operation sequences

The main idea of this section is to describe the sequences to perform to minimize the audible pop to the minimum for the headphone output.

Due to the large number of stand-by combinations and to be the most flexible, the handling of the sequence from one working mode to another is left to the software. So for helping the software designer in this task, some specific sequences are automatically performed by CODEC and an interrupt mechanism (IRQ signal and associated registers) warns the application when these sequences end.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.22.1 Initialization and configuration

To use the embedded CODEC with AIC, the following AIC registers should be set up before start the CODEC:

AICFR.ICDC = 1
AICFR.AUSEL = 1
AICFR.BCKD = 0
AICFR.SYNCD = 0
I2SCR.AMSL = 0
I2SCR.ESCLK = 1

14.22.2 Start up sequence (DAC)

This sequence is from Power-on mode to CODEC REPLAY mode.

The output sound is **drive** by DAC.

The intent of the following sequence is to prevent for large audible glitches due to the system start-up with the CODEC.

Before this sequence, setup the AIC properly.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 14-11 Start up sequence

SEQUENCE:

A : Once the sleep mode is left, the application turns on the DAC (SB_DAC register bit) and after 0.5 ms switches the analog mute signal to ‘0’ (MUTE_HP); then the application sets the HP gain stage to the maximum value (+6 dB)

B: Ramp-up cycle starts: after waiting 1 ms, the application turns on the headphone output stage (SB_HP = 0). The codec sets DAC_MODE_UPD to ‘1’ and then generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to ‘1’.

C: Ramp-up cycle ends: once the ramp-up cycle is started, the application handles the interrupt, resets the DAC_MODE_EVENT flag by writing ‘1’ on it. Once the ramp ends, the Codec sets DAC_MODE_UPD to ‘0’ and then generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to ‘1’.

D: IRQ handling and gain-up cycle: the application handles the interrupt, resets the DAC_MODE_EVENT flag by writing ‘1’ on it. The application first sets the wished gain on

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Headphone and then releases the DAC mute (DAC_MUTE register bit). In the meantime, the application sends valid audio data to the Codec DAC.

E: IRQ generation: once the gain-up cycle starts, DAC_MUTE[1:0] is set to '10' (cf register 4.2.2.8) and the Codec generates an interrupt by setting DAC_MUTE_EVENT flag to '1'. Then the application resets the DAC_MUTE_EVENT flag by writing '1' on it.

F: IRQ handling and active mode: once the gain-up cycle ends, DAC_MUTE[1:0] is set to '00' (cf register 4.2.2.8) and the Codec generates an interrupt by setting DAC_MUTE_EVENT flag to '1'. The application handles the interrupt and resets the DAC_MUTE_EVENT flag by writing '1' on. The Codec DAC path is now fully activated.

14.22.3 Shutdown sequence (DAC)

This sequence is from CODEC REPLAY mode to STANDBY mode.
The output sound is driving by DAC.

The intent of the following sequence is to prevent for large audible glitches due to the system shutdown with the CODEC.

Figure 14-12 Shutdown sequence

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SEQUENCE:

F: Initial state: the DAC path is activated.

G: Gain-down cycle starts: the application activates the mute of the DAC (DAC_MUTE register bit). Once the gain-down cycle starts, DAC_MUTE[1:0] is set to '01' (cf register 4.2.2.8) and the Codec generates an interrupt on IRQ by setting DAC_MUTE_EVENT flag to '1'. Then the application handles the interrupt and resets the DAC_MUTE_EVENT flag by writing '1' on it.

H: Gain-down cycle ends: once the gain-down cycle completes, DAC_MUTE[1:0] is set to '11' (cf register 4.2.2.8) and the Codec generates an interrupt on IRQ by setting DAC_MUTE_EVENT flag to '1'. The application handles the interrupt stopping sending audio data and resets the DAC_MUTE_EVENT flag by writing '1' on it. Then the application sets the HP gain stage to the maximum value (+6 dB).

I: Ramp-down cycle starts: the application turns off the headphone output stage (SB_HP = 1). The Codec sets DAC_MODE_UPD to '1' and then generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to '1'. The application handles the interrupt, resets the DAC_MODE_EVENT flag by writing '1' on it.

J: Ramp-down cycle ends: once the ramp down cycle ended, the CODEC sets DAC_MODE_UPD to '0' and then generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to '1'.

K: IRQ handling: the application handles the interrupt and resets the DAC_MODE_EVENT flag by writing '1' on it. Then, the application can activate the analog mute (MUTE_HP). Finally, the application turns off the DAC path (SB_DAC register bit) to be in sleep mode or turn off the Codec (SB_SLEEP, SB register bits).

14.22.4 Start up sequence (Line input)

Headphone port when driven by Line Input

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SEQUENCE:

A: initial state: playback mode is active. Valid analog audio signals are available at the input of the switch matrix

B: initializing output port: the application first sets the line-in gain to its minimum value and headphone gain stage to its maximum value (+6 dB). This setting is taken into account after a few clocks cycles. Then the application sets MUTE_HP to ‘0’.

C: ramp-up cycle: the application turns on the headphone output stage (SB_HP = ‘0’). The CODEC sets DAC_MODE_UPD to ‘1’ and then generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to ‘1’. Once the ramp-up cycle starts, the application handles the interrupt, resets the DAC_MODE_EVENT flag by writing ‘1’ on it.

D: IRQ generation: once the ramp-up cycle ends, the CODEC sets DAC_MODE_UPD to ‘0’ and generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to ‘1’. The application resets the DAC_MODE_EVENT flag by writing ‘1’ on it.

E: Headphone stage gain-up: the application sets HP gain stage to the targeted value (TBD). The maximum duration of the gain ramping equals (number of gain steps * Tcrossout). Refer to section 4.6 for the value of Tcrossout.

F: Line-in stage gain-up: the application sets the line-in gain stage to the targeted value. The maximum duration of the gain ramping equals (number of gain steps * Tcrossout).

G: Active mode: the signal path is now fully active.

14.22.5 Shutdown sequence (Line input)

Stopping sequence from playback mode to sleep mode

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SEQUENCE:

G: Active mode: the bypass mode is on.

H: Line-in stage gain-down: the application sets the line-in gain stage to the minimum value. The maximum duration of the gain ramping equals (number of gain steps * Tcrossout). Refer to section 4.6 for the value of Tcrossout.

I: Headphone stage gain-up (TBD): the application sets the headphone gain to the maximum value (+6 dB). The maximum duration of the gain ramping equals (number of gain steps * Tcrossout).

J: Ramp-down cycle: the application can now turn off the headphone output stages (SB_HP register bit). The CODEC sets DAC_MODE_UPD to '1' and generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to '1'. The application resets the DAC_MODE_EVENT flag by writing '1' on it.

K: IRQ generation: once the ramp down cycle ends, the CODEC sets DAC_MODE_UPD to '0' and generates an interrupt on IRQ by setting DAC_MODE_EVENT flag to '1'.

L: IRQ handling: the application handles the interrupt and resets the DAC_MODE_EVENT flag by writing '1' on it. The application can then activate the analog mute (MUTE_HP). The signal path is now off.

14.23 Circuits design suggestions

This section lists a few PCB design suggestions with **different** using mode.

14.23.1 Avoid quiet ground common currents

14.23.1.1 References pins

To work properly, CODEC requires **a** few additional external components.

CODEC includes an internal voltage reference. To insure a correct common mode biasing of the internal components, an additional voltage VCAP is used. This requires connecting two decoupling capacitors (Cext) between the pin VCAP and AVSCDC. One 10uF low ESR (ceramic or tantalum) and one 100nF ceramic have to be used. The ceramic capacitor has to be kept as close as possible to IC package (closer than 0.2 inch).

14.23.1.2 Power supply pins

CODEC analog power supplies require external decoupling capacitors.

For each power supply pin, one 100nF ceramic capacitor has to be used. This ceramic capacitor has to be kept as close as possible to IC package (closer than 0.2 inch). One low ESR (ceramic or tantalum) capacitor has to be used to decouple the analog power supply provided to the CODEC. Its value depends on the power supply generator; its typical value is between 1uF and 10uF. Ideally use separate ground planes for analog and digital parts.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Connect all ground pins with thick traces to power plane in order to ensure lowest impedance connections.

AVSCDC must be connected to the PCB analog single point reference (star connection) ground (AGND).

14.23.2 Headphone connection (Capacitor-coupled)

Capacitor-coupled headphone and line connection

Figure 14-13 Capacitor-coupled connection

The AOHPL and AOHPR pins are connected to the headphone through an external bypass capacitor which is a DC blocking capacitors.

This capacitor is called C_L. When the headphone resistance R_L is 16 Ohm, The tantalum blocking capacitor C_L is 220 uF.

The DC value of the signal AOHPL or AOHPR equals to AVDCDC/2.

The ground of the headphone is connected to AGND, which is the PCB analog single point reference (star connection) ground.

14.23.3 Microphone connection

The optimal performance for the SNR is obtained in differential Microphone inputs with a FS input level

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

corresponding to peak-to-peak amplitude of the signal is 0.2125V when GIM = 20 (Gain value = 20).

We recommend customer to use differential MIC input for better performance.

Application schematic with differential MIC input (using MICBIAS pin):

Application schematic with differential MIC input (Vmicbias generated on board):

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Application schematic with single-ended MIC input (using MICBIAS pin):

Application schematic with single-ended MIC input (Vmicbias generated on board):

In single-ended connection, one external resistor (R) has to be used to bias the electret microphone. In differential connection, a pair of external resistor (R1, R2) has to be used to bias the electret microphone. The resistors value relation between them is R1 = R2 = R/2.

Specific value of resistor (R, commonly from 2.2k Ohm to 4.7k Ohm) and Vmicbias (if generated on board, usually from 1 to 2V or more) depends on the selected EC (Electret Condenser) microphone. The 1nf decoupling capacitance used in MICBIAS pin removes high frequency noise of the chip.

Setting SB_MIC1/SB_MIC2 to 1 will close microphone input path for saving power, also setting SB_MICBIAS to 1 will close MICBIAS stage and the MICBIAS output voltage will be zero.

MICBIAS output voltage scales with AVDCDC, equals to 5/6*AVDCDC (typical 2.08V).
MICBIAS output current is 4mA max.
MICBIAS output noise is 40uVrms max.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

This configuration is better suited for microphone with single wire + shielding.

The AVSCDC Pin is connected the analog quiet reference ground in the chip (refers to Grounds and analog signal references). So the ground of MIC must be connected to AVSCDC using a star connection.

14.23.4 Description of the connections to the jack

When the jack is inserted, “sense” and “ground” are disconnected.

No jack plugged: the switch acts as a short-circuit.

Jack plugged: the switch acts as an open circuit.

14.23.4.1 Grounds and analog signal references

In order to limit the parasitic disturbances from the AVSHP output power supplies to inter VREFN analog quiet ground (which is using AVSCDC pin), should use the following principle to distribute the grounds.

Figure 14-14 Ground distributing

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Minimize the values of the connections parasitic resistance Rpar1, Rpar2.

Take a special care for Rpar1 in order to limit the disturbance from the output stages (AVSHP) to the signal reference (VREFN).

The reference of the input signals must be connected to VREFN (internal quiet ground which using the AVSCDC pin) using a star connection.

14.23.5 PCB considerations

To work properly, CODEC analog power supplies require external decoupling capacitors.

In the VCAP pin, one 10uF low ESR (ceramic or tantalum) called C2 and one 100nF ceramic called C1 have to be used. The ceramic capacitor has to be kept as close as possible to IC package (closer than 0.2 inch).

For each power supply pin, one 100nF ceramic capacitor has to be used. The capacitor used in AVDCDC pin is called C4, the capacitor used in AVDHP pin is C6. These ceramic capacitors have to be kept as close as possible to IC package (closer than 0.2 inch).

One low ESR (ceramic or tantalum) capacitor called C3 has to be used to decouple the analog power supply provided to the CODEC. Its value depends on the power supply generator; its typical value is between 1uF and 10uF. Ideally use separate ground planes for analog and digital parts.

C1, C2, C3, C4, C5, C6 are defined in section "Required external components".

The reference PCB design is shown below:

Figure 14-15 the bottom corner of chip PCB Layer

This is just an example reference diagram. You should change and select the PCB layer and route

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

with your design constraints.

14.23.5.1 Required external components

The following table summarizes the external components required for a proper working of CODEC, except those used for the analog input and output signals.

Name	Description	Typical Value	Unit
C1	Ceramic reference decoupling capacitor. Cext.	100	nF
C2	Tantalum reference decoupling capacitor. Cext.	10	uF
C3	Tantalum analog power supply decoupling capacitor.	1 to 10	uF
C4	Ceramic AVDCDC decoupling capacitor.	100	nF
C5	Ceramic inter signal VREFP decoupling capacitor (can be shared with C4).	100	nF
C6	Ceramic AVDHP decoupling capacitor. Not shown in section PCB considerations.	100	nF
C8	MICBIAS decoupling capacitor, Refer to section "Microphone connection".	1	nF
C9, C10	External bypass capacitor, for DC blocking, Refer to section "Headphone connection (Capacitor-coupled)".	220	uF
Rhpl, Rhpr	Headphone jack pull-down resistors, Refer to section "Headphone connection (Capacitor-coupled)".	470 or 4.7K	Ohm
R	In single-ended connection, one external resistor (R) has to be used to bias the electret microphone. Refer to section "Headphone connection (Capacitor-coupled)".	2.2K ~ 4.7K	Ohm

14.24 Analog characteristics

14.24.1 Line input to audio ADC path

Measurement conditions:
T = 25°C, AVDCDC = 3.3 V, input sine wave with a frequency of 1 kHz, Fmclk = 12 MHz, Fs = 8 to 96 kHz,measurement bandwidth 20 Hz – 20 kHz, unless otherwise specified.

Parameter	Test conditions	Min.	Typ	Max.	Unit
Input level (2)	Full Scale, Gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 20 dB	0.189	0.212	0.239	Vp-p
SNR	A-weighted, 1 kHz sine wave @ Full Scale and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 0 dB	85 (TBC)	90 (TBC)		dB

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	A-weighted, 1 kHz sine wave @ Full Scale and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 20 dB	75 (TBC)	80 (TBC)		dB
THD	1 kHz sine wave @ Full Scale -1 dB and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 20 dB		-70 (TBC)	-60 (TBC)	dB
	1 kHz sine wave @ Full Scale -1 dB and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 0 dB		-80 (TBC)	-70 (TBC)	dB
Dynamic range	A-weighted, 1 kHz sine wave @ Full Scale -60 dB and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 0 dB (note 1)	85 (TBC)	90 (TBC)		dB
	A-weighted, 1 kHz sine wave @ Full Scale -60 dB and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 20 dB (note 1)	75 (TBC)	80 (TBC)		dB
PSRR	100 mVp-p 1 kHz sine wave is applied to AVDCDC, input data is 0 and gain GIDL, GIDR = 0 dB, boost gain GIM1, GIM2 = 20 dB		90 (TBC)		dB
Gain boost	Boost gain GIM1, GIM2 when activated	0		20	dB
Gain boost step	GIM1, GIM2 @1 kHz		4		dB
Gain boost accuracy	GIM1, GIM2 @1 kHz	-1 (TBC)		+1 (TBC)	dB
Input resistance	Boost gain GIM1, GIM2 = 0 dB	63	80	96	kOhm
	Boost gain GIM1, GIM2 = 20 dB	10	12.5	15	
Input capacitance	Includes 10 pF for ESD, bonding and package pins capacitances			25	pF
Input bypass capacitor	Cbyline		1		uF

Note 1: The specified value is extrapolated by adding 60 dB to the measured SNR

Note 2: The Full Scale input voltage scales with LDO output: VREFP_CODEC.

14.24.2 Audio DAC to headphone output path

Measurement conditions: T = 25°C, AVDCDC = 3.3 V, input sine wave with a frequency of 1 kHz, Fmclk = 12 MHz, Fs = 8 to96 kHz, measurement bandwidth 20 Hz – 20 kHz, unless otherwise specified.					
Parameter	**Test conditions**	**Min.**	**Typ**	**Max.**	**Unit**
FOutput level (3)	Full Scale, Gain GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load	1.89	2.12	2.39	Vp-p

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Full Scale, Gain GOL, GOR = -3 dB, GODL, GODR = 0 dB, 16 Ohm load (TBC)	1.33	1.5	1.69	Vp-p
Maximum output power	Rl = 16 Ohm		17.6 (TBC)		mW
SNR	A-weighted, 1 kHz sine wave @ Full Scale, Gain GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load	95 (TBC)	100 (TBC)		dB
Idle Noise	A-weighted with no signal and gain GOL, GOR= -10 dB, GODL, GODR = 0 dB, 16 Ohm load		-103 (TBC)	-98 (TBC)	dBV
THD, THD+N	1 kHz sine wave @ Full Scale -1 dB, GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load		-85 (TBC)	-75 (TBC)	dB
	1 kHz sine wave @ Full Scale -1 dB and gain GOL, GOR = -3 dB, GODL, GODR = 0 dB, 16 Ohm load		-70 (TBC)	-65 (TBC)	dB
Dynamic range	A-weighted, 1 kHz sine wave @ Full Scale -60 dB, Gain GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load (note 1)	95 (TBC)	100 (TBC)		dB
Wide Band Noise	1 kHz sine wave @ Full Scale and gain GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load, measurement bandwidth 20 kHz – 100 kHz		65		dB
PSRR	100 mVp-p 1 kHz sine wave is applied to AVDCDC, input data is 0 and gain GOL, GOR = 0 dB, GODL, GODR = 0 dB, 10 kOhm load		90 (TBC)		dB
	100 mVp-p 1 kHz sine wave is applied to AVDCDC, input data is 0 and gain GOL, GOR = -25 dB, GODL, GODR = 0 dB, 16 Ohm load		90 (TBC)		dB
PuN	Active <-> inactive, 10 kOhm load		-60 (TBC)		dBVp
	Active <-> inactive, 16 Ohm load		-60 (TBC)		dBVp
Output resistance	Rl	16			Ohm
Output bypass capacitor	Cl (Rl = 10 kOhm)			1	uF
	Cl (Rl = 16 Ohm)			220	uF
Gain range	GOL, GOR 5-bit programmable range, digital control, @1 kHz	-25		+6	dB

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	GODL, GODR 5-bit programmable range, digital control, @1 kHz	-31		0	
Gain step	GOL, GOR and GODL, GODR monotonic		1		dB
Gain accuracy	GOL, GOR, GODL, GODR @1 kHz	-0.5		+0.5	dB
Output capacitance (note 2)	Cp			200	pF
Polarity	DIL,R to AOHPL,R		+1		

Note 1: The specified value is extrapolated by adding 60 dB to the measured SNR with a FS-60 dB input signal.

Note 2: Output may oscillate above specified load capacitances. The capacitance is equivalent to a 2-meter cable.

Note 3: The Full Scale input voltage scales with the nLR output: VREFP_CODEC.

14.24.3 Audio DAC to mono line output path

Measurement conditions:
T = 25°C, AVDCDC = 3.3 V, input sine wave with a frequency of 1 kHz, Fmclk = 12 MHz, Fs = 8 to 96kHz, measurement bandwidth 20 Hz – 20 kHz, unless otherwise specified.

Parameter	Test conditions	Min.	Typ	Max.	Unit
Output level (2)	Full Scale, Gain GODL, GODR = 0 dB	3.78	4.25	4.78	Vp-p
SNR	A-weighted, 1 kHz sine wave @ Full Scale, Gain GODL, GODR = 0 dB	90 (TBC)	95 (TBC)		dB
THD+N	1 kHz sine wave @ Full Scale -1 dB, Gain GODL, GODR = 0 dB		-85 (TBC)	-75 (TBC)	dB
Dynamic range	A-weighted, 1 kHz sine wave @ Full Scale – 60 dB, Gain GODL, GODR = 0 dB (note 1)	90 (TBC)	95 (TBC)		dB
PSRR	100 mVp-p 1 kHz sine wave is applied to AVDCDC, Gain GODL, GODR = 0 dB, input data is 0		90 (TBC)		dB
Output resistance	Rl	10			kOhm
Output capacitance	Cp			100	pF

Note 1: The specified value is extrapolated by adding 60 dB to the measured SNR

Note 2: The Full Scale input voltage scales with the nLR output: VREFP_CODEC.

14.24.4 Line input to headphone output path (analog bypass)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Measurement conditions: T = 25°C, AVDCDC = 3.3 V, input sine wave with a frequency of 1kHz, Fmclk = 12 MHz, Fs = 8 to 96 kHz,measurement bandwidth 20 Hz – 20 kHz, unless otherwise specified.					
Parameter	**Test conditions**	**Min.**	**Typ**	**Max.**	**Unit**
Input level	Full Scale	1.89	2.12	2.39	Vp-p
Input resistance		10k			Ohm
Output level (2)	Full Scale, Gain GOL, GOR = 0 dB, GIL, GIR = 0 dB, 10 kOhm load (note 1)	1.89	2.12	2.39	Vp-p
	Full Scale, Gain GOL, GOR = -3 dB, GIL, GIR = 0 dB, 16 Ohm load (TBC)	1.33	1.5	1.69	Vp-p
SNR	A-weighted, 1 kHz sine wave @ Full Scale, Gain GOL, GOR = 0 dB, GIL, GIR = 0 dB	90 (TBC)	95 (TBC)		dB
THD, THD+N	1 kHz sine wave @ Full Scale -1 dB, Gain GOL, GOR = 0 dB, GIL, GIR = 0 dB, 10 kOhm load		-85 (TBC)	-75 (TBC)	dB
	1 kHz sine wave @ Full Scale -1 dB and gain GOL, GOR = -3 dB, GIL, GIR = 0 dB, 16 Ohm load		-70 (TBC)	-65 (TBC)	dB
Dynamic range	A-weighted, 1 kHz sine wave @ Full Scale -60 dB, Gain GOL, GOR = 0 dB, GIL, GIR = 0 dB, 10 kOhm load (note 1)	90 (TBC)	95 (TBC)		dB
PSRR	100 mVp-p 1 kHz sine wave is applied to AVDCDC, input data is 0 and gain GOL, GOR = 0 dB, GIL, GIR = 0 dB, 10 kOhm load		90 (TBC)		dB
	100 mVp-p 1 kHz sine wave is applied to AVDCDC, input data is 0 and gain GOL, GOR = -25 dB, GIL, GIR = 0 dB, 16 Ohm load		90 (TBC)		dB
PuN	Active <-> inactive, 10 kOhm load		-60 (TBC)		dBVp
	Active <-> inactive, 16 Ohm load		-60 (TBC)		dBVp
Output resistance	Rl	16			Ohm
Gain range	GIL, GIR 5-bit programmable range, digital control, @1 kHz	-25		+6	dB
Gain step	GIL, GIR monotonic		1		dB
Gain accuracy	GIL, GIR, @1 kHz	-0.5		+0.5	dB
Input capacitance	Includes 10 pF for ESD, bonding and package pins capacitances			25	pF

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Input bypass capacitor	Cbyline		1		uF
Polarity	AIL,R to AOL,R		+1		

Note 1: The specified value is extrapolated by adding 60 dB to the measured SNR

Note 2: The Full Scale input voltage scales with the nLR output: VREFP_CODEC

14.24.5 Micbias and reference

Measurement conditions: T = 25°C, AVDCDC=3.3 V, input sine wave with a frequency of 1 kHz, Fmclk = 12 MHz, Fs = 8to 96 kHz, measurement bandwidth 20 Hz – 20 kHz, unless otherwise specified.					
Parameter	**Test conditions**	**Min.**	**Typ**	**Max.**	**Unit**
Micbias output level		2.35 (TBC)	2.5 (TBC)	2.65 (TBC)	V
Micbias output current				4	mA
Micbias output noise	A-weighted		30 (TBC)	40 (TBC)	uVrms
Micbias decoupling capacitor	Cmic	0.75	1	1.25	nF
VCAP voltage			2 (TBC)		V
VREFP_CODEC (also called VREF)		2.35 (TBC)	2.5 (TBC)	2.65 (TBC)	V

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15 DMIC Controller

15.1 Overview

The DMIC Controller is used to convert the signal of Digital Microphone to PCM signal.

Digital microphone relative to the analog MEMS microphone is more advanced technology, because of its high sensitivity, high signal to noise ratio, high noise reject capability, low distortion, low power consumption and other characteristics, widely used in higher recording requirements on new equipment.

15.1.1 Features

1. Low power DMIC Controller
2. 16 bits data interface and 24bit precision internal controller.
3. SNR: 90dB, THD: -90dB @ FS -20dB
4. Linear high pass filter include. Attenuation: -2.9dB@100Hz, -22dB@27Hz. -36dB@10Hz
5. Low power voice trigger when waiting to start talking.
6. Mono and stereo digital MIC support.
7. Support voice data pre-fetch when trigger enable and the data interface disable, but do not increase the power dissipation.
8. Sample frequency supported: 8k, 16k.
9. Support low power mode, user for decrease DMIC sensor and DMIC Controller power dissipation.

15.1.2 Pin Description

There are 2 or 3 Pins used to connect between DMIC Controller and DMIC sensor. They are listed and described in following table.

Table 15-1 DMIC Controller Pins Description

Name	I/O	Description
DMIC0_CLK	Output	DMIC sensor clock.
DMIC0_IN	Input	DMIC sensor data. Support left and right channel wire and when share mode①, only left channel when split mode①.
DMIC1_IN	Input	DMIC sensor data. Support right channel when split mode.

Note①: When the split mode, DMIC_CR0.SPLIT_DI need be set to ‘1’. When the share mode, DMIC_CR0.SPLIT_DI need be set to ‘0’.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.1.3 DMIC Controller Block Diagram

Figure 15-1 DMIC Controller block diagram 1 of 2

Figure 15-2 DMIC Controller block diagram 2 of 2

15.2 Register

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute

   R - Read only

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

W - Write only
RW - read and write
RCW - read and write, but clear to 0 by read
RSW - read and write, but set to 1 by read
RWC - read and write, clear to 0 by write 1, write 0 has no effect
RWS - read and write, set to 1 by write 1, write 0 has no effect
RC - read only, and clear to 0 by read
RS - read only, and set to 1 by read
SPEC - special access method, relate to its description

3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

15.2.1 Registers Memory Map

Table 15-2 Registers Memory Map-Address Base

Name	Base	Description
DMIC	0x10021000	Address base of DMIC Controller.

Table 15-3 DMIC Controller Registers Description

Name	Description	Reset Value	Address offset	Access Size
DMIC_CR0	DMIC Control register 0	0x000000c8	**0x000**	32
DMIC_GCR	DMIC Gain Control Register	0x00000000	**0x004**	32
DMIC_IMR	DMIC Interrupt Mask Register	0x0000001f	**0x008**	32
DMIC_ICR	DMIC Interrupt Control Register	0x00000000	**0x00C**	32
TRI_CR	DMIC Trigger Control Register	0x00030000	**0x010**	32
THR_H	DMIC Trigger High Threshold	0x00020000	**0x014**	32
THR_L	DMIC Trigger Low Threshold	0x0000127a	**0x018**	32
TRI_M_MAX	DMIC Trigger number M Max value	0x000007e8	**0x01C**	32
TRI_N_MAX	DMIC Trigger number N Max value	0x00000003	**0x020**	32
DMIC_DR	DMIC Data Register	0x????????	**0x030**	32
DMIC_FCR	DMIC FIFO Control register	0x00000040	**0x034**	32
DMIC_FSR	DMIC FIFO State Register	0x000?00??	**0x038**	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.2.2 Register Description

15.2.2.1 DMIC Control register 0 (DMIC_CR0)

	DMIC_CR0																															BASE + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RESET	RESET_TRI	Reserved																UNPACK_MSB	UNPACK_DIS	SW_LR	SPLIT_DI	MCLK_24	PACK_EN	SR		STEREO	DMIC_R	LP_MODE	HPF1_EN	TRI_EN	DMIC_EN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0

Bits	Name	Description	RW
31	RESET	If set this bit to '1' will reset all DMIC (Include FIFO \ data interface \ trigger block but without software configuration registers).	RWS
30	RESET_TRI	If set this bit to '1' will reset DMIC TRI block. (Without software configuration registers).	RWS
29:12	Reserved	Writes to these bits have no effect and always read as 0.	R
13	UNPACK_MSB	Use Big-Endian format at DMIC_DR when PACK_EN = 1 and UNPACK_DIS = 1.	RW
12	UNPACK_DIS	Disable unpack function, when pack mode.	RW
11	SW_LR	Switch left and right channel when data input. Set this bit to '1' will enable the function.	RW
10	SPLIT_DI	Split the left and right data at DMIC0_IN and DMIC1_IN else left and right data will share in DMIC0_IN. Set this bit to '1' will enable the function.	RW
9	Reserved_0	Inner usage, keep to 0.	RW
8	PACK_EN	Pack the 16-bit data to FIFO aim to economize the FIFO. Set this bit to '1' will enable the function.	RW
7:6	SR	DMIC interface output sample rate: 0: 8k 1: 16k 3: 48k(test only) 4. Disable the DMIC interface.	RW
5	STEREO	STEREO mode, left and right channel output at the same time. Set this bit to '1' will enable the function.	RW
4	Reserved_0	Inner usage, keep to 0.	RW
3	LP_MODE	Low power mode. When this mode the DMIC_CLK frequency is 1.2MHz else the DMIC_CLK frequency is 2.4MHz. Set this bit to '1' will enable the function.	RW
2	HPF1_EN	HPF1(High pass filter) enable. Set this bit to '1' will enable the function.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	TRI_EN	Trigger mode enable. Set this bit to '1' will enable the function.	RW
0	DMIC_EN	DMIC enable. Set this bit to '1' will enable the function.	RW

15.2.2.2 DMIC Gain Control Register (DMIC_GCR)

	DMIC_GCR																															BASE + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											DGAIN				
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:4	Reserved	Writes to these bits have no effect and always read as 0.	R
3:0	DGAIN	Digital gain register. The gain of PGA equal to DGAIN * 3 dB.	RW

15.2.2.3 DMIC Interrupt Mask Register (DMIC_IMR)

	DMIC_IMR																															BASE + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											WAKEUP_MASK	EMPTY_MASK	FULL_MASK	PRE_READ_MASK	TRI_MASK
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

Bits	Name	Description	RW
31:5	Reserved	Writes to these bits have no effect and always read as 0.	R
4	WAKEUP_MASK	Enable WAKEUP Interrupt.	RW
3	EMPTY_MASK	Enable EMPTY Interrupt.	RW
2	FULL_MASK	Enable FULL Interrupt.	RW
1	PRE_READ_ MASK	Enable PRE_READ Interrupt.	RW
0	TRI_MASK	Enable TRI Interrupt.	RW

Note: 1. The pre-read is a function used to control FIFO over run. When FIFO over run, FIFO will pre-read a data to ensure the data of FIFO and the later data is continuously, but the data is discontinuously with before.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.2.2.4 DMIC Interrupt Control Register (DMIC_ICR)

	DMIC_ICR																															BASE + 0x00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											WAKEUP_FLAG	EMPTY_FLAG	FULL_FLAG	PRE_READ_FLAG	TRI_FLAG
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:5	Reserved	Writes to these bits have no effect and always read as 0.	R
4	WAKEUP_FLAG	When this bit is '1' indicate that wakeup of voice trigger was generate. Write this bit '1' will clear it.	RWC
3	EMPTY_FLAG	When this bit is '1' indicate that empty of FIFO was generate. Write this bit '1' will clear it.	RWC
2	FULL_FLAG	When this bit is '1' indicate that full of FIFO was generate. Write this bit '1' will clear it.	RWC
1	PRE_READ_ FLAG	When this bit is '1' indicate that pre-read of FIFO was generate. Write this bit '1' will clear it.	RWC
0	TRI_FLAG	When this bit is '1' indicate that voice trigger was triggered. Write this bit '1' will clear it.	RWC

Note: 1. The pre-read is a function used to control FIFO over run. When FIFO over run, FIFO will pre-read a data to ensure the data of FIFO and the later data is continuously.

15.2.2.5 DMIC Trigger Control Register (TRI_CR)

	TRI_CR																															BASE + 0x010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												TRI_MODE				Reserved											TRI_DEBUG	HPF2_EN	PREFETCH		TRI_CLR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writes to these bits have no effect and always read as 0.	R
19:16	TRI_MODE	Trigger mode: 0. If the signal great than the THR_L level, trigger generate.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1. If the signal peak is great than THR_L and less than THR_H, trigger generate. 2. If the signal great than the THR_L level and continue num N times, trigger generate. 3. If the signal peak is great than THR_L and less than THR_H and continue num N times, trigger generate. 4. Same as mode 2, but use THR_L and THR_H to Schmitt mode. 5~15. Reserved.	
15:5	Reserved	Writes to these bits have no effect and always read as 0.	R
4	TRI_DEBUG	Voice Trigger module Debug mode enable. Set this bit to '1' will transmit the data of Voice Trigger module input to FIFO.	RW
3	HPF2_EN	HPF2(High pass filter) enable. Set this bit to '1' will enable the function.	RW
2:1	PREFETCH	Auto open the DMIC interface and prefetch the data to FIFO and ignore the SR when trigger. 0. Prefetch function disable. 1. Prefetch function use 8k mode. 2. Prefetch function use 16k mode. 3. Reserved.	RW
0	TRI_CLR	If write this bit '1' will clear trigger.	RWS

15.2.2.6 DMIC Trigger High Threshold (THR_H)

	THR_H																															BASE + 0x014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												THR_H																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	Reserved	Writes to these bits have no effect and always read as 0.	R
19:0	THR_H	DMIC Trigger High Threshold.	RW

15.2.2.7 DMIC Trigger Low Threshold (THR_L)

	THR_L																															BASE + 0x018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												THR_L																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	1	1	0	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:20	Reserved	Writes to these bits have no effect and always read as 0.	R
19:0	THR_L	DMIC Trigger Low Threshold.	RW

15.2.2.8 DMIC Trigger number M Max value (TRI_M_MAX)

	TRI_M_MAX																															BASE + 0x01C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																M_MAX															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writes to these bits have no effect and always read as 0.	R
15:0	M_MAX	The num m max value used for the time out. After M cycle, if the trigger have not generate and the generate times of trigger less than N_MAX, the counter of N will be clear.	RW

15.2.2.9 DMIC Trigger number N Max value (TRI_N_MAX)

	TRI_N_MAX																															BASE + 0x020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																N_MAX															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:16	Reserved	Writes to these bits have no effect and always read as 0.	R
15:0	N_MAX	The num n max value used for the time. If trigger generated N times, Interrupt will be generate.	RW

15.2.2.10 DMIC Data Register (DMIC_DR)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DMIC_DR																															BASE + 0x030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFO_DATA																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	FIFO_DATA	FIFO data register. This register is connect to the FIFO. The left channel data laid at the bit 0 ~ bit 15, the right channel data laid at the bit 0 ~ bit 31. Note: When trigger debug mode,	R

15.2.2.11 DMIC FIFO Control register (DMIC_FCR)

	DMIC_FCR																															BASE + 0x034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDMS	Reserved																								FIFO_THR						
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bits	Name	Description	RW
31	RDMS	DMIC receive FIFO request enable. Set this bit to '1' will enable the function.	
30:7	Reserved	Writes to these bits have no effect and always read as 0.	R
6:0	FIFO_THR	FIFO threshold. When FIFO level is great than the FIFO threshold the DMIC request will generate.	RW

15.2.2.12 DMIC FIFO State Register (DMIC_FSR)

	DMIC_FSR																															BASE + 0x038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												FULL_S	TRIG_S	PRERD_S	EMPTY_S	Reserved									FIFO_LVL						
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:20	Reserved	Writes to these bits have no effect and always read as 0.	R
19	FULL_S	FIFO FULL state.	R
18	TRIG_S	FIFO voice trigger state.	R
17	PRERD_S	FIFO pre-read mode state.	R
16	EMPTY_S	FIFO empty state.	R
15:7	Reserved	Writes to these bits have no effect and always read as 0.	R
6:0	FIFO_LVL	FIFO level state.	R

15.3 DMIC Controller Timing

The following figures show the timing relationship for the DMIC transfers. The DMIC_CLK is generated from DMIC Controller to DMIC (Digital Microphone sensor). DMIC Controller samples the DMIC0_IN Pin use the different edge to two channels.

Figure 15-3 The DMIC signal Timing

15.4 DMIC Controller Operation

15.4.1 Turn on data path flow

1. Set all configuration registers. Such as SR \ STEREO \ LP_MODE \ IMR \ ICR \ HPF1_EN \ GCR.
2. Set DMIC_CR0.RESET = ‘1’ and then wait for this bit read ‘0’.
3. Set DMIC_EN = ‘1’, DMIC Controller will transfer the data to FIFO.

15.4.2 Turn on trigger path flow

10. Set all configuration registers. Such as SR \ STEREO \ LP_MODE \ IMR \ ICR \ HPF1_EN \ GCR.
11. Set DMIC_CR0.RESET = ‘1’ and then wait for this bit read ‘0’.
12. Set TRI_EN = ‘1’ and DMIC_EN = ‘1’, DMIC Controller will transfer the data to FIFO.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.4.3 Trigger application flow.

1. Set all configuration registers. Such as DMIC_CR0 \ SR \ STEREO \ LP_MODE \ IMR \ ICR \ HPF1_EN \ GCR.
2. Set DMIC_CR0.RESET = '1' and then wait for this bit read '0'.
3. Set DMIC_EN = '1', DMIC Controller will transfer the data to FIFO.
4. Set all Trigger configuration registers. Such as TRI_CR \ THR_H \ THR_L \ TRI_M \ TRI_N.
5. Set SR = '3' use for close DMIC data path, the data level of FIFO no longer increases.
6. Set RESET_EN = '1' and RESET_TRI = '1', and then wait for these bit read '0'.
7. Wait trigger interrupt.
8. When the interrupt is generated, set SR to target value, then DMIC Controller will transfer the data to FIFO.
9. set DMIC_EN = '0' to turn off the trigger part.

Note : If PREFETCH = '1', the data transfer will start at trigger interrupt generate.

Trigger level configuration

1. Set LP_MODE = '1' or '0' (In accordance with recode and trigger mode configuration).
   Set PACK_EN = '0', set UNPACK_DIS = '0', set SR = '3'.
   Set HPF2_EN = '1' or '0' (In accordance with trigger mode configuration).
   Set TRI_DEBUG = '1'.
2. Set DMIC_CR0.RESET = '1' and then wait for this bit read '0'.
3. Set DMIC_EN = '1', and set TRI_EN = '1'.
4. Read FIFO use by 32-bit signed format (The most 24-bit is valid data).
5. Identification the noise level and voice level, used to make sure the trigger level.
6. For example trigger mode 1. The THR_L = trigger result (the Trigger level of the previous step)(Normalization: range -1 ~ 1) * 2^19.
7. Set the THR_L value.
8. Run the trigger application follow.

15.4.4 Wakeup by DMIC Trigger flow

When CPU is in sleep mode, If DMIC controller is active (please see CPM specification sleep mode) and the voice meet the requirements (Trigger configure registers), DMIC will generate a interrupt to INTC and then wakeup CPU. The DMIC wakeup flow is:

1. Configure DMIC controller for Trigger application flow.
2. configure CPM to mask INTC's interrupt
3. disable CPU's IE(interrupt enable)
4. configure CPM to make system turn into sleep, idle deep sleep or deep idle state. (See CPM sleep mode)
5. When the voice meet the requirements (Trigger configure registers), DMIC will generate a interrupt to INTC and then wakeup CPU.
6. Set step 2 \ 3 \ 4's configuration to normal state
7. Interrupt handler Configure DMIC controller for Turn on data path flow.
8. Check the voice data from DMIC controller by voice recognition software. If the voice is the

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

keyword of wakeup, boot the operating system.

15.4.5 FIFO operations

DMIC controller has a 32*128 bit FIFO. For optimum to use the FIFO, it can be set to 3 store mode and 3 APB bus read mode.

The 3 store mode:

31 … 16	15 … 0
0x0	Data31
…	…
0x0	Data2
0x0	Data1
0x0	Data0

Figure 15-4 Mono Non-pack mode

31 … 16	15 … 0
Data31	Data30
…	…
Data5	Data4
Data3	Data2
Data1	Data0

Figure 15-5 Mono pack mode

31 … 16	15 … 0
Data_R31	Data_L31
…	…
Data_R2	Data_L2
Data_R1	Data_L1
Data_R0	Data_L0

Figure 15-6 Stereo mode

The 3 APB bus read mode:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 15-7 APB read when unpack function enable

Figure 15-8 APB read when unpack function disable and Big-Endian mode (UNPACK_MSB) enable

Figure 15-9 APB read when unpack function disable and non Big-Endian mode (UNPACK_MSB) disable

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.4.6 Trigger operations:

Voice trigger of DMIC controller has 5 trigger modes.

15.4.6.1 Single level trigger mode.

When this mode, the trigger interrupt generate condition is that voice signal was higher than the trigger level (THR_L).

Figure 15-10 One word voice signal

Figure 15-11 Magnifying the above figure 15-10

15.4.6.2 Multiple level trigger mode.

Some times, the noise can higher than the trigger level too. Such as bump noise or knocking noise. But the sound has a law that the signal duration is very short. We can use the multiple

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

level trigger mode to avoid the false trigger.

When this mode, the trigger interrupt generate condition is that voice signal was higher than the trigger level (THR_L) multi-times (TRI_N_MAX).

Figure 15-12 A bump noise before one word voice signal

Figure 15-13 Magnifying the bump part of above figure 15-12

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 15-14 Magnifying the voice part of above figure 15-12

15.4.6.3 Single limited level trigger mode.

Some times, the noise can higher than the trigger level even and voice level. Such as bump noise or knocking noise. We can use the multiple limited level trigger mode to avoid the false trigger.

Figure 15-15 A bump noise before one word voice signal

15.4.6.4 Multiple limited level trigger mode.

This mode focus the advantages of the above two modes that Multiple level trigger mode and Single limited level trigger mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.4.6.5 Multiple Schmitt trigger mode.

If the noise value is close to the voice level, noise will make the voice signal in the vicinity of the threshold trigger repeatedly. But perhaps the real voice is much less than the trigger cycles, it will make the purpose of setting the number (TRI_N_MAX and TRI_M_MAX) of inaccuracies. We can use the multiple Schmitt trigger mode to filter the noise of superimposed.

Figure 15-16 Magnifying the one word voice signal use by multiple level trigger mode

Figure 15-17 Magnifying the one word voice signal use by multiple Schmitt trigger mode

15.4.7 DMIC controller clocks set

DMIC controller needs a work clock from CPM. Set I2SCDR of CPM to generate a 12M Hz clocks for DMIC controller. The clock of I2SCDR generated is shared by AIC. DMIC controller generates a 1.2MHz (low power mode) or 2.4MHz DMIC clock for DMIC sensor use. If DMIC_EN was set to ‘0’, the DMIC clock will be gated

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 4

MEMORY INTERFACE

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16 DDR Controller

16.1 Overview

DDRC (DDR Controller) is a general IP which provides an interface to DDR2, DDR3, mobile DDR(LPDDR) and LPDDR2 memory. The DDRC IP is designed for SOC usage and it is configurable, scalable to meet the requirement of various SOC.

Features:

- Multi-port Architecture and asynchronous interface to all port
- Support clock-stop mode
- Support auto-refresh and self-refresh
- Support power-down mode and deep-power-down mode
- Programmable DDR timing parameters
- Programmable DDR row and column address width
- Programmable remapping from Logic address to Physical address

16.1.1 Supported DDR SDRAM Types

- DDR2
- DDR3
- LPDDR (mobile DDR)
- LPDDR2

Row address width less than 16-bit & Column width less than 12-bit are supported.

16.1.2 Block Diagram

Following figure shows the functional block diagram of DDR system.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 16-1 DDR system block diagram

16.1.3 Revision History

16.2 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute

   R - Read only

   W - Write only

   RW - read and write

   RCW - read and write, but clear to 0 by read

   RSW - read and write, but set to 1 by read

   RWC - read and write, clear to 0 by write 1, write 0 has no effect

   RWS - read and write, set to 1 by write 1, write 0 has no effect

   RC - read only, and clear to 0 by read

   RS - read only, and set to 1 by read

   SPEC - special access method, relate to its description
3. Reset Value

   1 - reset to 1

   0 - reset to 0

   ? - value unknown after reset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2.1 Registers Memory Map

The registers in DDR are divided into 3 groups base on ways to be accessed:

Table 16-1 Registers Memory Map-Address Base

Name	Base	Description
DDRC_AHB	*0x134F0000*	Address base of controller registers which are accessed by AHB
DDRC_APB	*0x13012000*	Address base of controller registers which are accessed by APB
DDRP_APB	*0x13011000*	Address base of PHY registers which are accessed by APB

Note: the DDRC_AHB is strictly forbidden to access during data retention mode.

Table 16-2 DDR Controller(DDRC) Registers Description

Name	Description	Reset Value	Base group	Address offset	Access Size
DSTATUS	DDRC Status Register	0x00000001	DDRC_AHB	0x000	32
DCFG	DDRC Configure Register	0x00000000	DDRC_AHB	0x004	32
DCTRL	DDR General Control Register	0x00800000	DDRC_AHB	0x008	32
DLMR	DDR Load -Mode-Register	0x00000000	DDRC_AHB	0x00C	32
DTIMING1	DDR Timing Configure Register 1	0x01050505	DDRC_AHB	0x060	32
DTIMING2	DDR Timing Configure Register 2	0x02050505	DDRC_AHB	0x064	32
DTIMING3	DDR Timing Configure Register 3	0x25050205	DDRC_AHB	0x068	32
DTIMING4	DDR Timing Configure Register 4	0x0F220551	DDRC_AHB	0x06C	32
DTIMING5	DDR Timing Configure Register 5	0xFF050505	DDRC_AHB	0x070	32
DTIMING6	DDR Timing Configure Register 6	0x10280505	DDRC_AHB	0x074	32
DREFCNT	Auto-Refresh Counter	0x00FF0000	DDRC_AHB	0x018	32
DMMAP0	DRAM CS0 Map Configure Register	0x0000FF00	DDRC_AHB	0x024	32
DMMAP1	DRAM CS1 Map Configure Register	0x0000FF00	DDRC_AHB	0x028	32
DDLP	DFI low power handshake register	0x0000F003	DDRC_AHB	0x0BC	32
DREMAP1	DDR address remapping register1	0x03020100	DDRC_AHB	0x09C	32
DREMAP2	DDR address remapping register2	0x07060504	DDRC_AHB	0x0A0	32
DREMAP3	DDR address remapping register3	0x0B0A0908	DDRC_AHB	0x0A4	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

DREMAP4	DDR address remapping register4	0x0F0E0D0C	DDRC_AHB	0x0A8	32
DREMAP5	DDR address remapping register5	0x13121110	DDRC_AHB	0x0AC	32
WCMDCTRL1	Write command reorder & grouping (Performance control)	0x40408040	DDRC_AHB	0x100	32
RCMDCTRL0	Read Channel mode control (Performance control)	0x08084848	DDRC_AHB	0x104	32
RCMDCTRL1	Read Channel mode control (Performance control)	0x00004808	DDRC_AHB	0x108	32
WDATTHD0	Wdata Channel mode control (Performance control)	0x08084948	DDRC_AHB	0x114	32
WDATTHD1	Wdata Channel mode control (Performance control)	0x00004808	DDRC_AHB	0x118	32
IPORTWPRI	Configuration of Internal Priority for write channel (Performance control)	0x00000000	DDRC_AHB	0x240	32
IPORTRPRI	Configuration of Internal Priority for read channel (Performance control)	0x00000000	DDRC_AHB	0x244	32
CHxWDOS	WQoS configure for each channel (Performance control)	0x80030100	DDRC_AHB	0x200 0x204 0x208 0x20C 0x210 0x214 0x218	32
CHxRDOS	RQoS configure for each channel (Performance control)	0x80030100	DDRC_AHB	0x220 0x224 0x228 0x22C 0x230 0x234 0x238	32
AUTOSR_CNT	Auto Self Refresh counter configuration	0x00000100	DDRC_AHB	0x308	32
AUTOSR_EN	Auto Self Refresh enable	0x00000000	DDRC_AHB	0x304	32
EFPRB_CFG	Bus efficiency probe configuration	0x00000000	DDRC_AHB	0x30C	32
EFPRB_VALUE	Bus efficiency data		DDRC_AHB	0x310	32
CLKSTP_CFG	Auto Clock Gating		DDRC_APB	0x068	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	configuration				
DDRC_STATUS	DDR status	0x0000c000	DDRC_APB	0x06C	32
PHYRET_CFG	Retention Mode Configuration	0x00002202	DDRC_APB	0x034	32
EPD_HADDR	EPD wake up configure	0x00000000	DDRC_APB	0x070	32
EPD_LADDR	EPD wake up configure	0x00000000	DDRC_APB	0x074	32
PHYRST_CFG	DDR Phy reset control	0x00000000	DDRC_APB	0x080	32
CKC_CFG0	Clock change Configuration	0x00400040	DDRC_APB	0x084	32
CKC_CFG1	Clock change Configuration	0x00000A10	DDRC_APB	0x088	32

16.2.2 Registers and Fields Description

16.2.2.1 DDRC Status Register (DSTATUS)

	DSTATUS																															DDRC_AHB + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								ENDIAN	MISS	DPDN	PDN	AREF	SREF	CKE1	CKE0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bits	Name	Description	RW
31	DFI_INIT	DFI (PHY interface) initial status: 0: DFI initial not completed, PHY is not ready for access 1: DFI initial completed, PHY is ready for access	R
30:8	Reserved	Writes to these bits have no effect and always read as 0	R
7	ENDIAN	Endian status: 0: Little data Endian. 1: Big data Endian.	R
6	MISS	Indicate the accessed address is out of memory mapping range 0: All operated address is falling in DRAM mapping range 1: At least one operated address is out of DRAM mapping range	RC
5	DPDN	Indicate the deep-power-down status of DDR memory 0: DDR memory is NOT in deep-power-down state 1: DDR memory is in deep-power-down state	R
4	PDN	Indicate the power-down status of DDR memory 0: DDR memory is NOT in power-down state 1: DDR memory is in power-down state	R
3	AREF	Indicate the auto-refresh status of DDR memory. 0: DDR memory is NOT in auto-refresh state 1: DDR memory is in auto-refresh state	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2	SREF	Indicate the self-refresh status of DDR memory 0: DDR memory is NOT in self-refresh state 1: DDR memory is in self-refresh state	R
1	CKE1	Indicates the status of CKE1 pin (only usable after DDR PHY initialized) 0: CKE1 is low 1: CKE1 is high	R
0	CKE0	Indicates the status of CKE0 pin (only usable after DDR PHY initialized) 0: CKE0 is low 1: CKE0 is high	R

16.2.2.2 DDRC Configure Register (DCFG)

Configure the external memory, static configuration only. i.e. This register can NOT be changed on-the-fly.

	DCFG																															DDRC_AHB + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		ROW1			COL1			BA1	IMBA	BSL	Reserved	TYPE			ODTEN	MISPE	Reserved	RAW0			COL0			CS1EN	CS0EN	CL				BA0	DW
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	
29:27	ROW1	Row Address width. Specify the row address width of external DDR. 000: 12-bit row address 001: 13-bit row address 010: 14-bit row address 011: 15-bit row address 100: 16-bit row address	RW
26:24	COL1	Column Address width. Specify the Column address width of external DDR 000: 8-bit Column address. 001: 9-bit Column address 010: 10-bit Column address 011: 11-bit Column address 100: 12-bit Column address	RW
23	BA1	0: 4 bank device, Pin ba[1:0] is valid, ba[2] un-used. 1:8 bank device, Pin ba[2:0] is valid.	RW
22	IMBA	0: CS0, CS1 connected 2 memory chips which have same ROW, COL, BA configuration. In this mode, both chips are configured by ROW0,COL0, BA0 .	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		ROW1,COL1,BA1 does not effect the system. 1: CS0, CS1 connected 2 memory chips which have different ROW, COL, BA configuration. Chip0 is configured by ROW0, COL0, BA0; Chip1 is configured by ROW1, COL1, BA1;	
21	BSL	Burst length for DDR chips 1: 8 burst 0: 4 burst	RW
20	Reserved	Writing has no effect, read as zero.	
19:17	TYPE	Select external memory device type. Please refer below table for more detail description 000 :Normal SDR (Single-Data-Rate) SDRAM(Not support). 001 : Mobile SDR(Not support). 010 : Normal DDR1 (Not support) 011 : Mobile DDR(LPDDR). 100 : Normal DDR2. 101 : LPDDR2 110 : Normal DDR3 111 : LPDDR3 (Not support).	RW
16	ODTEN	whether enable ODT 0:not enable ODT 1:enable ODT	RW
15	MISPE	Missing CS protection Enable . Set 1 to enable. If software read (or write) a memory space which is not selected by any CS, controller will return random data to a read operation (or mask write operation) to avoid system bus being locked. A CS missing flag will set in DSTATUS.	RW
14	Reserved	Reserved. Writing has no effect, read as zero.	
13:11	RAW0	Row Address width. Specify the row address width of external DDR. 000: 12-bit row address 001: 13-bit row address 010: 14-bit row address 011: 15-bit row address 100: 16-bit row address	RW
10:8	COL0	Column Address width. Specify the Column address width of external DDR 000: 8-bit Column address. 001: 9-bit Column address 010: 10-bit Column address 011: 11-bit Column address 100: 12-bit Column address	RW
7	CS1EN	DDR Chip-Select-1 Enable, If an DDR memory chip is connected to ddr pin cs1, set CS1EN=1.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: DDR Pin CS1 is not in use 1; DDR Pin CS1 is in use.	
6	CS0EN	DDR Chip-Select-0 Enable. If DDR memory is connected to ddr pin cs0, set CS0EN=1. 0: DDR Pin CS0 is not in use. 1: DDR Pin CS0 is in use.	RW
5:2	CL	CAS Latency. Not in use for this version	RW
1	BA0	Bank Address width of DDR memory. 0: 4 bank device, Pin ba[1:0] is valid, ba[2] un-used. 1: 8 bank device, Pin ba[2:0] is valid.	RW
0	DW	External DDR Memory Data Width. Specify the external DDR memory data width 0: External memory data width is 16-bit. 1: External memory data width is 32-bit.	RW

16.2.2.3 DDR Control Register (DCTRL)

On the positive edge of START, the command selected by CMD field will be performed.

	DCTRL																															DDRC_AHB + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								DFI_RST	DLL_RST	CTL_RST	CFG_RST	Reserved		KEEPSR	Reserved	ACTPD	PDT			ACTSTP	Reserved				DPD	SR	UNALIGN	ALH	Reserved	CKE	RESET
RST	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Reserved. Writing has no effect, read as zero.	R
23	DFI_RST	Reserved, please keep the default value.	RW
22	DLL_RST	Reserved, please keep the default value.	RW
21	CTL_RST	Reserved, please keep the default value.	RW
20	CFG_RST	Reserved, please keep the default value.	RW
19:18	Reserved	Reserved. Writing has no effect, read as zero.	
17	KEEPSR	inner usage, keep to 0.	RW
16	Reserved	Reserved. Writing has no effect, read as zero.	
15	ACTPD	Active Power-Down. Some SDRAM devices support Active-Power-Down. By default, ACTPD=0, hardware will precharge all active banks before entering Power-Down mode, so called Precharge-Power-Down. By setting ACTPD=1, hardware drives SDRAM into	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Power-Down mode without precharge all active banks, some banks are still active in Power-Down mode, so called Active-Power-Down. 0: Precharge all banks before entering power-down. 1: Do not precharge all banks before entering power-down.	
14:12	PDT	Power-Down Timer. When there's no access to DDR memory for a period of time, hardware drives DDR into power-down mode to save power consumption. Hardware can exit Power-Down mode automatically when new access arrives. If PDT=0, power-down function is disabled. If enable power-down mode, it is recommended to be set after DDR initialization. 000: power-down disabled, hardware never drive SDRAM into power-down mode. 001: Enter power-down after 8 tCK idle. 010: Enter power-down after 16 tCK idle. 011: Enter power-down after 32 tCK idle. 100: Enter power-down after 64 tCK idle. 101: Enter power-down after 128 tCK idle. 110 - 111: Reserved.	RW
11	ACTSTP	Active Clock-Stop. 0: Clock can be stopped only after all banks be precharged 1: Clock can be stopped after some bank's row active	RW
10:7	Reserved	Reserved. Writing has no effect, read as zero.	
6	DPD	Software drives external Mobile DDR device entering Deep-Power-Down mode. Software set DPD = 1 drives external Mobile DDR device entering Deep-Power-Down mode instead of Power-Down mode, when there's no access to DDR memory for a period of time. So you must firstly enable Power-Down mode (refer to PDT). Software needs to reset DDR controller and re-do a complete initial process to exit Deep-Power-Down mode. When external device goes to Deep-Power-Down mode, it lose all data store in memory and registers. The memory chip will cut off inner power supply for power saving.	RW
5	SR	Software drive external DDR device entering Self-Refresh mode. Software set SR=1 drive external DDR device entering self-refresh mode; Software set SR=0 drive external DDR device exiting	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		self-refresh mode; In this mode, the CK to external DDR device would be stopped during self-refresh period; But the clock supplied to ddr_controller logic would not stop. Software can read & write ddr_controller registers in this mode. Software can NOT read or write memory data in this mode. **NOTE:** Software must guarantee that there's no memory access during self-refresh mode. Otherwise, software can NOT exit this mode, **system would be hang-up!!** 0: Drive external DDR device entering self-refresh mode. 1: Drive external DDR device exiting self-refresh mode.	
4	UNALIGN	Enable unaligned transfer on AXI BUS. Not use in this version	RW
3	ALH	Advanced Latency Hiding. This is a test-oriented register. Some latency is reduced in special cases. 0: Disable ALH. 1: Enable ALH.	RW
2	Reserved	Reserved. Writing has no effect, read as zero.	
1	CKE	Control the status of CKE pin. Write CKE=1 can set CKE pin to HIGH state. Write CKE=0 would be ignored. The default value of CKE Pin is low; CKE0,1 Pins status is represented by DDR_STATUS register. **Caution:** This register is used only for DDR initializing sequence; software should NOT update this register when DDR memory is in normal working mode. 0: Not set CKE Pin High. 1: Set CKE Pin to HIGH.	RW
0	RESET	Module reset for ddr_controller. Software is able to reset ddr_controller by setting RESET bit high and ends the reset by clear this bit. 0: End resetting ddr_controller. 1: Resetting ddr_controller.	RW

16.2.2.4 DDR Memory Initial Register (DLMR)

DLMR register is used for initializing the DDR SDRAM memory device.

On the positive edge of START, the command selected by CMD field will be performed.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

| | DLMR | | |

		1: On the positive edge of START, perform a command defined by CMD field	

16.2.2.5 DDR Timing Configure Register (DTIMING1,2,3,4,5,6)

The timing parameters are identical to the JEDEC DDR Specification.

	DTIMMING1																															DDRC_AHB + 0x060
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		tRTP						Reserved		tWTR						Reserved		tWR						Reserved		tWL					
RST	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1

If the DDR clock period are tCK.

All these registers' value equal to the number tCK cycle. For example, if tRTP=2, it means 2 tCK between READ and PRECHARGE;

Bits	Name	Description	RW
31:30	Reserved	Reserved. Writing has no effect, read as zero.	
29:24	tRTP	READ to PRECHARGE command period.	RW
23:22	Reserved	Reserved. Writing has no effect, read as zero.	
21:16	tWTR	WRITE to READ command delay.	RW
15:14	Reserved	Reserved. Writing has no effect, read as zero.	
13:8	tWR	WRITE Recovery Time defined by register MR of DDR2 DDR3 memory.	RW
7:6	Reserved	Reserved. Writing has no effect, read as zero.	
5:0	tWL	Write latency, please notice that this version only support AL=0.	RW

	DTIMMING2																															DDRC_AHB + 0x064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		tCCD						Reserved		tRAS						Reserved		tRCD						Reserved		tRL					
RST	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1

Bits	Name	Description	RW
31:30	Reserved	Reserved. Writing has no effect, read as zero.	
29:24	tCCD	CAS# to CAS# command delay	RW
23:22	Reserved	Reserved. Writing has no effect, read as zero.	
21:16	tRAS	ACTIVE to PRECHARGE command period. tRAS defines the ACTIVE to PRECHARGE command	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		period to the same bank.	
15:14	Reserved	Reserved. Writing has no effect, read as zero.	
13:8	tRCD	ACTIVE to READ or WRITE command period. tRCD defines the ACTIVE to READ/WRITE command period to the same bank.	RW
7:6	Reserved	Reserved. Writing has no effect, read as zero.	
5:0	tRL	Read latency, please notice that this version only support AL=0.	RW

	DTIMMING3																															DDRC_AHB + 0x068
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved	ONUM				tCKSRE			Reserved		tRP						Reserved		tRRD						Reserved		tRC					
RST	0	0	1	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1

Bits	Name	Description	RW
31	Reserved	Reserved. Writing has no effect, read as zero.	
30:27	ONUM	Keep to 4 in this version.	RW
26:24	tCKSRE	Valid clock after enter self-refresh(tCKSRX = tCKSRE in this version). Delay time is tCKSRE*8 (tCK)	RW
23:22	Reserved	Reserved. Writing has no effect, read as zero.	
21:16	tRP	PRECHARGE command period tRP defines the PRECHARGE to next command period to the same bank.	RW
15:14	Reserved	Reserved. Writing has no effect, read as zero.	
13:8	tRRD	ACTIVE bank A to ACTIVE bank B command period. tRRD defines the ACTIVE to ACTIVE command period to different banks.	RW
7:6	Reserved	Reserved. Writing has no effect, read as zero.	
5:0	tRC	ACTIVE to ACTIVE command period. tRC defines the ACTIVE to ACTIVE command period to the same bank.	RW

	DTIMING4																															DDRC_AHB + 0x06C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		tRFC						tEXTRW			tRWCOV		tCKE			Reserved				tMINSR				Reserved	tXP			Reserved		tMRD	
RST	0	0	0	0	1	1	1	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:30	Reserved	Reserved. Writing has no effect, read as zero.	
29:24	tRFC	AUTO-REFRESH command period. tRFC defines the minimum delay after an AUTO-REFRESH command. During tRFC period, no command can be issued to DDR memory. Delay Time = **2 * (tRFC + 1).** 000000: 1 tCK. 000001: 3 tCK. 000010: 5 tCK. 000011: 7 tCK. **… … : … 2 * (tRFC + 1) …** 111101: 125 tCK. 111110: 127 tCK. 111111: 129 tCK. e: tCK – one DDR memory clock cycle, typical tCK value is 7.5 ns (133MHz clock).	RW
23:21	tEXTRW	keep the default value.	RW
20:19	tRWCOV	keep the default value.	RW
18:16	tCKE	minimum CKE pulse width tCKE define the minimum CKE pulse width, include high level and low level. 000: 1 tCK. 001: 2 tCK. 010: 3 tCK. 011: 4 tCK. 100: 5 tCK. 101: 6 tCK. 110: 7 tCK. 111: 8 tCK.	RW
15:12	Reserved	Reserved. Writing has no effect, read as zero.	
11:8	tMINSR	Minimum Self-Refresh / Deep-Power-Down time. After DDR memory turns into Self-Refresh or Deep-Power-Down mode, it will NOT exit until tMINSR condition meets. Delay Time = tMINSR * 8 + 1. 0000: 1*8 + 1 tCK. 0001: 2*8 + 1 tCK. 0010: 3*8 + 1 tCK. 0011: 4*8 + 1 tCK. **… …: … tMINSR * 8 + 1 …** 1101: 14*8 + 1 tCK.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1110: 15*8 + 1 tCK. 1111: 16*8 + 1 tCK.	
7	Reserved	Reserved. Writing has no effect, read as zero.	
6:4	tXP	EXIT-POWER-DOWN to next valid command period. tXP defines the EXIT-POWER-DOWN to next valid command period to all banks. 000: 1 tCK. 001: 1 tCK. 010: 2 tCK. 011: 3 tCK. 100: 4 tCK. 101: 5 tCK. 110: 6 tCK. 111: 7 tCK.	RW
3:2	Reserved	Reserved. Writing has no effect, read as zero.	
1:0	tMRD	Load-Mode-Register to next valid command period tMRD defines the Load-Mode-Register to next valid command period 00: 1 tCK. 01: 2 tCK. 10: 3 tCK. 11: 4 tCK	RW

	DTIMMING5																																DDRC_AHB + 0x070
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tCTLUPD								Reserved		tRTW						Reserved		tRDLAT						Reserved		tWDLAT					
RST	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1

Bits	Name	Description	RW
31:24	tCTLUPD	Inner usage. Not need to change.	RW
23:22	Reserved	Reserved. Writing has no effect, read as zero.	
21:16	tRTW	Read to Write latency.	RW
15:14	Reserved	Reserved. Writing has no effect, read as zero.	
13:8	tRDLAT	tRL-2. (when use LPDDR2, set tRDLAT=tRL).	RW
7:6	Reserved	Reserved. Writing has no effect, read as zero.	
5:0	tWDLAT	tWL-1(when use LPDDR2, set tWDLAT=tWL).	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DTIMMING6																															DDRC_AHB + 0x074
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tXSRD								Reserved		tFAW						Reserved		tCFGW						Reserved		tCFGR					
RST	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1

Bits	Name	Description	RW
31:24	tXSRD	Exit self-refresh to READ delay.	RW
23:22	Reserved	Reserved. Writing has no effect, read as zero.	
21:16	tFAW	4-active command window.	RW
15:14	Reserved	Reserved. Writing has no effect, read as zero.	
13:8	tCFGW	Write PHY configure registers to other commands delay. Not need to change.	RW
7:6	Reserved	Reserved. Writing has no effect, read as zero.	
5:0	tCFGR	Ready PHY configure registers to other commands delay. Not need to change.	RW

16.2.2.6 DDR Auto-Refresh Counter (DREFCNT)

	DREFCNT																															DDRC_AHB + 0x018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								CON								CNT								Reserved				CLK_DIV			REF_EN
RST	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Reserved. Writing has no effect, read as zero.	
23:16	CON	A constant value used to compare with the CNT value After reset, CON=0xFF and CNT=0x00; It is not recommended to set CON=0x00	RW
15:8	CNT	8-bit counter; When the value of CNT match the value of CON, flag bit EQU is set high and an auto-refresh command will be issued to DDR memory. READ only.	RW
7:4	Reserved	Reserved. Writing has no effect, read as zero.	
3:1	CLK_DIV	Clock Divider. Divide the dclk to generate a lower frequency of clock to drive the auto-refresh counter. This helps to save power consumption. Set CLK_DIV=0 can disable the clock to auto-refresh counter. When the DDR memory is in self-refresh mode or in	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		deep-power-down mode, disable the clock of auto-refresh counter can save power consumption. Future more, the module clock to DDRC can also be stopped. dclk is CKO clock, When ddr work in 500Mbps, dclk is 250Mhz. 000: dclk / 16. 001: dclk / 32. 010: dclk / 64. 011: dclk / 128. 100: dclk / 256. 101: dclk / 512. 110: dclk / 1024. 111: --	
0	REF_EN	Enable Refresh Counter. Software set REF_EN=1 right after initialize ddr memory. 0: Enable auto-refresh counter. 1: Disable auto-refresh counter.	RW

16.2.2.7 DDR Memory Map Configure Register (DMMAP0,1)

The physical base address and size of external DDR Memory can be configured by DMMAP register.

The size of external DDR Memory must be: **2^(24+n), n=0, 1, 2, 3, ….**

When the following equation is met:

(AXI_BUS_Address[31:24] & MASK[7:0]) == BASE

The DDR Memory is selected.

	DMMAP0, **DMMAP1**																																**DDRC_AHB + 0x024** **DDRC_AHB + 0x028**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																BASE								MASK							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	
15:8	BASE	base address.	RW
7:0	MASK	address mask Examples: **1:**DDR address space in system memory : 0x2000_0000 ~	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0x2FFF_FFFF (256MB) BASE=0x20 MASK=0xF0. **2:**DDR address space in system memory : 0x5000_0000 ~ 0x57FF_FFFF (128MB) BASE=0x50 MASK=0xF8.	

16.2.2.8 DDR DFI low power handshake control register (DDLP)

	DDLP																															DDRC_AHB + 0x0BC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													Only	UAR	UZQ	SLP				FLP				Reserved					FSR	FPD	LPEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:19	Reserved	Reserved. Writing has no effect, read as zero.	
18	Only	Inner usage, keep to 0	RW
17	UAR	Inner usage, keep to 0	RW
16	UZQ	Inner usage, keep to 0	RW
15:12	SLP	slow LP handshake and fast LP handshake timing register, Just keep it as reset value.	RW
11:8	FLP	slow LP handshake and fast LP handshake timing register, Just keep it as reset value.	RW
7:3	Reserved	Reserved. Writing has no effect, read as zero.	
2	FSR	0: Use slow lp handshake process during self-refresh. 1: Use fast lp handshake process during self-refresh.	RW
1	FPD	0: Use slow lp handshake process during power down. 1: Use fast lp handshake process during power down.	RW
0	LPEN	DFI low power(lp) interface enable	RW

16.2.2.9 DDR Address Remapping Register (DREMAP1,2,3,4,5)

	DREMAP1																															DDRC_AHB + 0x09C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			BIT3MP					Reserved			BIT2MP					Reserved			BIT1MP					Reserved			BIT0MP				
RST	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DREMAP2																															DDRC_AHB + 0x0A0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			BIT7MP					Reserved			BIT6MP					Reserved			BIT5MP					Reserved			BIT4MP				
RST	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0

	DREMAP3																															DDRC_AHB + 0x0A4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			BIT11MP					Reserved			BIT10MP					Reserved			BIT9MP					Reserved			BIT8MP				
RST	0	0	0	0	1	0	1	1	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0

	DREMAP4																															DDRC_AHB + 0x0A8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			BIT15MP					Reserved			BIT14MP					Reserved			BIT13MP					Reserved			BIT12MP				
RST	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	0	0	0	0	0	1	1	0	1	0	0	0	0	1	1	0	0

	DREMAP5																															DDRC_AHB + 0x0AC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			BIT19MP					Reserved			BIT18MP					Reserved			BIT17MP					Reserved			BIT16MP				
RST	0	0	0	1	0	0	1	1	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0

DREMAP1~5 are used to define address mapping in DDRC. The “BITnMP” represents address[31:20]. The low 4KB(bit [11:0]) address are reserved without any change. The high 20 bits are configurable. For example, If you want to remap address between address[27:25] and address[14:12], you can set BIT12MP as 0, BIT13MP as 1, BIT14MP as 2, BIT0MP as 12, BIT1MP as 13 and BIT2MP as 14.

Original address:

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

Remapped address

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

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2.2.10 Performance wcmd reorder & grouping (WCMDCTRL1)

Terminology:

- Priority: priority of command, the command with higher priority will be processed prior to that with lower priority. The priority is generated from:
  1. on-port control signals (hpri, ARPRI, AWPRI)
  2. iport_pri (register configurable for each port)
  3. QoS counter
- Page-hit-conflit:

  Page-hit: wpage-hit

  Page-conflict: the page of write command is conflict with the page of read command. In this case, to guarantee the read-write coherency, the write command must be process prior to read command.

	WCMDCTRL1																															DDRC_AHB + 0x100
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wcmd_lock_thd					wpage_hit_max								pend_qos_cnt												wram_in_use					disable_reorder	msk_empty_wfifo
RST	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bits	Name	Description	RW
[31:27]	wcmd_lock_thd	Indicates how many wcmd (write commands) are grouped together. Higher value improves total bandwidth of ddr, but leads to longer read command latency.	RW
[26:19]	wpage_hit_max	Wpage-hit affects the reorder of wcmd, the wcmd with same page should be grouped for higher performance. This register indicates the maximum number of wcmd to be bonded. Higher value improves total bandwidth of ddr, but leads to longer read maximum latency.	RW
[18:7]	pend_qos_cnt	Once the wcmd is accepted, a counter indicates how long the command is pending, when the pending time is exceed Pend_QoS once, the priority of this command will increase 1, up to 3, which increases the priority to be processed. This mechanism avoids the wcmd with lower priority pending too long.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Range: 0x001 ~0xFFF **NOTE**: configure step: 1. set d*isable_reorder* to 0. 2. execute *step1* for 14 times 3. set wcmd_pend_qos to expect value 4. set disable_reorder to 1, (Optional) Higher value improves total bandwidth of ddr, but leads to longer maximum latency for write command with lower priority. **Dynamic configuration is not supported.**	
[6:2]	wram_in_use	0~12. Ram to be used in write command reorder. Reduction of this number leads to poor performance. **Configuration is not recommended**	RW
[1]	disable_reorder	0: wcmd will be reordered based on priority and page-hit-conflict 1: wcmd will not be reordered, the commands will be process with the order of being received. **Dynamic configuration is not recommended.**	RW
[0]	msk_empty_wfifo	0: wcmd is involved in arbitration immediately. 1: wcmd is not involved in arbitration until receives enough wdata.	RW

16.2.2.11 Performance rcmd request control (RCMDCTRL0)

Terminology:

rfifo_thd: read fifo threshold. Read command is not involved in arbitration until the vacancy in rfifo is higher than rfifo threshold. This should improves the ddr performance when processing accessing discrete pages. However, for page-access (commands access the same page continuously) the performance will be lagged as the threshold affects the command grouping.

rcmd_igr_cflit: guarantee coherency between wcmd and rcmd. If the coherency if not required in some application (such as LCD), ignoring of confliction would improve ddr read performance.

	RCMDCTRL0																															**DDRC_AHB + 0x104**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Ch3_rcmd_igr_cflit	Ch3_rfifo_thd_en	Ch3_rfifo_thd						Ch2_rcmd_igr_cflit	Ch2_rfifo_thd_en	Ch2_rfifo_thd						Reserved								Ch0_rcmd_igr_cflit	Ch0_rfifo_thd_en	Ch0_rfifo_thd					
RST	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
[31]	ch3_rcmd_igr_cflit	1: enable. Consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[30]	ch3_fifo_thd_en	0: disable rfifo_thd (recommanded) 1: enable rfifo_thd	RW
[29:24]	ch3_rfifo_thd	If Ch3 rfifo level threshold higher than this value, rcmd request is masked	RW
[23]	Ch2_rcmd_igr_cflit	1: enable. Consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[22]	Ch2_fifo_thd_en	0: disable rfifo_thd (recommanded) 1: enable rfifo_thd	RW
[21:16]	Ch2_rfifo_thd	If Ch3 rfifo level threshold higher than this value, rcmd request is masked	RW
[15]	Reserved	Writes to these bits have no effect and always read as 0	R
[14]	Reserved	Writes to these bits have no effect and always read as 0	R
[13:8]	Reserved	Writes to these bits have no effect and always read as 0	R
[7]	Ch0_rcmd_igr_cflit	1: enable. Consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[6]	Ch0_fifo_thd_en	0: disable rfifo_thd (recommanded) 1: enable rfifo_thd	RW
[5:0]	Ch0_rfifo_thd	If Ch3 rfifo level threshold higher than this value, rcmd request is masked	RW

16.2.2.12 Performance rcmd request control (RCMDCTRL1)

	RCMDCTRL1																															DDRC_AHB + 0x108
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								Ch6_rcmd_igr_cflit	Ch6_rfifo_thd_en	Ch6_rfifo_thd						Ch5_rcmd_igr_cflit	Ch5_rfifo_thd_en	Ch5_rfifo_thd						Ch4_rcmd_igr_cflit	Ch4_rfifo_thd_en	Ch4_rfifo_thd					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0

Bits	Name	Description	RW
[31:24]	Reserved	Writes to these bits have no effect and always read as 0	R
[23]	Ch6_rcmd_igr_cflit	0: disable rfifo_thd (recommended) 1: enable rfifo_thd	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

[22]	Ch6_fifo_thd_en	1: enable. consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[21:16]	Ch6_rfifo_thd	If Ch1 rfifo level threshold higher than this value, rcmd request is masked	RW
[15]	Ch5_rcmd_igr_cflit	0: disable rfifo_thd (recommended) 1: enable rfifo_thd	RW
[14]	Ch5_fifo_thd_en	1: enable. consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[13:8]	Ch5_rfifo_thd	If Ch1 rfifo level threshold higher than this value, rcmd request is masked	RW
[7]	Ch4_rcmd_igr_cflit	0: disable rfifo_thd (recommended) 1: enable rfifo_thd	RW
[6]	Ch4_fifo_thd_en	1: enable. consistence between wcmd and rcmd will not be guaranteed. 0: disable.	RW
[5:0]	Ch4_rfifo_thd	If Ch1 rfifo level threshold higher than this value, rcmd request is masked	RW

16.2.2.13 performance wcmd request control (WDATTHD0)

Terminology:

Wfifo_thd: wdata fifo threshold. In axi port, as the wdata might be far behind of wcmd. The threshold guarantees the wcmd will not be involved in arbitration until receive enough wdata. This improves ddr performance for axi write access which wdata does not continuously transfer on bus. But for continuous transfer, the performance would be worse.

	WDATTHD0																															DDRC_AHB + 0x114
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved	Ch3_wfifo_thd_en	Ch3_wfifo_thd						Reserved	Ch2_wfifo_thd_en	Ch2_wfifo_thd						Reserved									Ch0_wfifo_thd_en	Ch0_wfifo_thd					
RST	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	0	0

Bits	Name	Description	RW
31	Reserved	Writes to these bits have no effect and always read as 0	R
30	Ch3_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[29:24]	Ch3_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

[22]	Ch2_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[21:16]	Ch2_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW
[14]	Reserved	Writes to these bits have no effect	RW
[13:8]	Reserved	Writes to these bits have no effect	RW
[6]	Ch0_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[5:0]	Ch0_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW

16.2.2.14 performance wcmd request control (WDATTHD1)

	WDATTHD1																															DDRC_AHB + 0x118
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved									Ch6_wfifo_thd_en	Ch6_wfifo_thd							Ch5_wfifo_thd_en	Ch5_wfifo_thd						Reserved	Ch4_wfifo_thd_en	Ch4_wfifo_thd					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0

Bits	Name	Description	RW
[22]	Ch6_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[21:16]	Ch6_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW
[14]	Ch5_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[13:8]	Ch5_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW
[6]	Ch4_wfifo_thd_en	1: Enable 1: Disable (recommended)	RW
[5:0]	Ch4_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	RW

16.2.2.15 performance priority control (IPORTPRI)

Terminology:

Priority affects on arbitration between ports. Transfers with higher priority will be grant first

Port_pri: priority provided by masters.

Iport_pri: register configurable which applies to each ports. The finial priority is the larger value of port_pri and iport_pri.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

IPORTWPRI																																DDRC_AHB + 0x240
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				Ch6_iport_wpri_en	Reserved	Ch6_iport_wpri		Ch5_iport_wpri_en	Reserved	Ch5_iport_wpri		Ch4_iport_wpri_en	Reserved	Ch4_iport_wpri		Ch3_iport_wpri_en	Reserved	Ch3_iport_wpri		Ch2_iport_wpri_en	Reserved	Ch2_iport_wpri		Reserved				Ch0_iport_wpri_en	Reserved	Ch0_iport_wpri	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPORTRPRI																																DDRC_AHB + 0x244
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				Ch6_iport_rpri_en	Reserved	Ch6_iport_rpri		Ch5_iport_rpri_en	Reserved	Ch5_iport_rpri		Ch4_iport_rpri_en	Reserved	Ch4_iport_rpri		Ch3_iport_rpri_en	Reserved	Ch3_iport_rpri		Ch2_iport_rpri_en	Reserved	Ch2_iport_rpri		Reserved				Ch0_iport_rpri_en	Reserved	Ch0_iport_rpri	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[27]	Ch6_iport_wpri_en Ch6_iport_rpri_en	Enable iport_pri (iport_wpri is write channel priority, Iport rpri is read channel priority) 0: disable, use external priority 1: enable, use the max priority of external and internal priority	RW
[25:24]	Ch6_iport_wpri Ch6_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW
[23]	Ch5_iport_wpri_en Ch5_iport_rpri_en	Enable iport_pri (iport_wpri is write channel priority, Iport rpri is read channel priority) 0: disable, use external priority 1: enable, use the max priority of external and internal priority	RW
[21:20]	Ch5_iport_wpri Ch5_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW
[19]	Ch4_iport_wpri_en Ch4_iport_rpri_en	Enable iport_pri 0: disable, use external priority	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: enable, use the max priority of external and internal priority	
[17:16]	Ch4_iport_wpri Ch4_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW
[15]	Ch3_iport_wpri_en Ch3_iport_rpri_en	Enable iport_pri 0: disable, use external priority 1: enable, use the max priority of external and internal priority	RW
[13:12]	Ch3_iport_wpri Ch3_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW
[11]	Ch2_iport_wpri_en Ch2_iport_rpri_en	Enable iport_pri 0: disable, use external priority 1: enable, use the max priority of external and internal priority	RW
[9:8]	Ch2_iport_wpri Ch2_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW
[7:4]	Reserved	Writes to these bits have no effect	RW
[3]	Ch0_iport_wpri_en Ch0_iport_rpri_en	Enable iport_pri 0: disable, use external priority 1: enable, use the max priority of external and internal priority	RW
[1:0]	Ch0_iport_wpri Ch0_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	RW

16.2.2.16 performance QoS control (CHxQOS0,1,2,3,4,5)

Terminology:

Qos: Generally speaking, the Qos controller is a timer, it indicates the how long the command is pending. Once the timer exceed predefined threshold (con), the priority increases by 1, up to top-limit (max). The higher priority is, the sooner it will be processed. This register affects the latency.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	CHxWQOS0,1,2,3,4,5,6																DDRC_AHB + 0x200,204,208,20C,210,214,218															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Qos_en	Outstanding_en	Reserved												max		con															
RST	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

	CHxRQOS0,1,2,3,4,5,6																DDRC_AHB + 0x220,224,228,22C,230,234,238															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Qos_en	Outstanding_en	Reserved												max		con															
RST	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31]	Qos_en	Enable QoS control for each channel	RW
[17:16]	Max	Up-limit of priority	RW
[15:0]	Con	Threshold of timer	RW

16.2.2.17 Auto-Self-Refresh Enable (AUTOSR_EN)

	AUTOSR_EN																DDRC_AHB + 0x304															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															AUTOSR_EN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[19:0]	BUS_IDLE_EN	0: disable auto self-refresh 1: enable auto self-refresh	RW

16.2.2.18 Auto-Self-Refresh Counter (AUTOSR_CNT)

Terminology:

AUTOSR: auto self-refresh. When the DDRC bus interface is idle for some time, the DDRC can issue a command to put the DRAM into self-refresh mode. This will same much power in off-chip DRAM.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	AUTOSR_CNT																															DDRC_AHB + 0x308
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												BUS_IDLE_CNT																			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31:20]	Reserved	Writes to these bits have no effect	R
[19:0]	BUS_IDLE_CNT	Timer Cnt: when the bus interface is idle(all transfer are finished), the internal timer is going to run, after the Timer Cnt is meet, DDR issues Self-Refresh command.	RW

16.2.2.19 Bus Efficient Probe Configure (BUSEFF_CFG)

	BUSEFF_CFG																															DDRC_AHB + 0x30C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved			enable	reserved																window											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31:29]	reserved	Writes to these bits have no effect, read as 0	R
[28]	enable	Enable: if set, the bus efficiency will be monitored.	RW
[27:12]	reserved	Writes to these bits have no effect, read as 0	R
[11:0]	window	Sample window: define the how long the period that bandwidth value refresh. Valid value: 1~1023.	RW

16.2.2.20 Bus Efficient Probe Value (BUSEFF_VAL)

	BUSEFF_VAL																															DDRC_AHB + 0x310
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved																				value											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
[31:12]	reserved	Writes to these bits have no effect, read as 0.	R
[11:0]	value	Indicates the valid data transfer on DRAM interface during cycles of BUSEFF_CFG.window. It is average value of adjacent window counting.	RW

16.2.2.21 Auto Clock Stop Configure Register (CLKSTP_CFG)

Description:

Under certain condition, the DDRC input clock can be stopped for power saving. For example, when the SDRAM is in self-refresh mode, the clock can be stop.

	CLKSTP_CFG																															DDRC_APB + 0x068
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	clkstp_en	idle_cond_en	lp_cond_en	sr_cond_en	Reserved																dly_timer											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31]	clkstp_en	Function enable	RW
[30]	idle_cond_en	Enable judge condition: ddrc idle	RW
[29]	lp_cond_en	Enable judge condition: low-power mode (DCTRL.PDT)	RW
[28]	sr_cond_en	Enable judge condition: self-refresh	RW
[27:12]	Reserved	Writes to these bits have no effect, read as 0	R
[11:0]	dly_timer	Specify the time before shutting down the clock after all condition met.	RW

Please Note that:

The CLKSTP will affect the function of lp(low power mode), auto-sr(auto self-refresh)mode. If the Clock is gating, the timer used for lp/auto-sr will stop running. As a result, if the Auto-self-refresh enabled, the bit [28] should be set.

16.2.2.22 DDRC Status Register2 (DDRC_STATUS2)

Description:

Read-Only register indicates the ddrc status. This register can be accessed even in DDR retention mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DDRC_STATUS																															DDRC_APB + 0x06C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													ckc_reject	ckc_ack	ckc_req	Reserved		ddr_sr_mode	ddr_lp_mode	ddrc_idle	ch6_bus_idle	ch5_bus_idle	ch4_bus_idle	ch3_bus_idle	ch2_bus_idle	ch1_bus_idle	ch0_bus_idle	Reserved			clkstp_req
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31:19]	Reserved	Writes to these bits have no effect and always read as 0	
[18]	ckc_reject	1: Clock change request is rejected	R
[17]	ckc_ack	1: Clock change request is acknowledged	R
[16]	ckc_req	1: Clock change request is raised from CPM	R
[15]	Reserved	Read as 1	R
[14]	Reserved	Read as 1	R
[13]	ddr_sr_mode	0: DRAM is not in self-refresh mode 1: DRAM is in self-refresh mode	R
[12]	ddr_lp_mode	0: ddr PHY is not in low-power mode 1: ddr PHY is in low-power mode	R
[11]	ddrc_idle	Indicates the ddrc control logic is idle	R
[10:4]	chx_bus_idle	Indicates the bus interface for each channel is idle	R
[3:1]	Reserved	Writes to these bits have no effect and always read as 0	
[0]	clkstp_req	Indicates the ddrc is requiring CPM gating ddrc clock	R

16.2.2.23 DDR PHY Retention Configure Register (PHYRET_CFG)

Description:

DDR PHY can be set into retention mode, which allow power cutting-down. This can save the leakage power in highest level.

However, when use this function, please flow the retention flow.

	PHYRET_CFG																															DDRC_APB + 0x034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																			cfg_iso_en	Reserved			phy_iso_en	Reserved			clk_iso_en	Reserved		ret_en_n	ret_en
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
[12	cfg_iso_en	PHY output prdata isolation enable	RW
[8]	phy_iso_en	PHY output dfi signal isolation enable (high-active)	RW
[4]	clk_iso_en	PHY output clock isolation enable	RW
[1]	ret_en_n	Retention enable: low-active	RW
[0]	ret_en	Retention enable: High-active (make sure: ret_en = ~ret_en_n)	RW

16.2.2.24 DDR PHY Reset Configure Register (PHYRST_CFG)

Soft Reset of DDRC

	PHYRST_CFG																															DDRC_APB + 0x080
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								dfi_reset	dll_reset	ctl_reset	cfg_reset																				phy_reset
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[23]	dfi_reset	Soft reset of dfi	RW
[22]	dll_reset	Soft reset of dll	RW
[21]	ctl_reset	Soft reset of ctl	RW
[20]	cfg_reset	Soft reset of cfg	RW
[0]	phy_reset	Soft reset of PHY	RW

16.2.2.25 EPD detecting address high end (EPD_HADDR)

This register with EPD_LADDR is designed for EPD monitoring whether the address within an range is accessed by any channel, if it was, an flag *detect_epd_wr* will be informed to EPD controller.

	EPD_HADDR																															DDRC_APB + 0x070
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	haddr																Reserved														clock_en	fast_detect
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bits	Name	Description	RW
[31:16]	haddr	High end of monitored addr range	RW
[15:2]	Reserved	Writing has no effect, read as 0	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

[1]	clock_en	Backdoor for enable clock for monitor logic (keep to 0 unless the function does not work)	RW
[0]	fast_detect	1: ignore asynchronous handshake between channels, will raise an fast flag when address detected. 0: raise the flag slow but with security. (keep to 0 unless the function does not work)	RW

16.2.2.26 EPD detecting address low end (EPD_LADDR)

	EPD_LADDR																															DDRC_APB + 0x074
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	laddr																Reserved															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
[31:16]	laddr	High end of monitored addr range	RW
[15:0]	Reversed	Writing has no effect, read as 0	R

16.2.2.27 Clock Change Configure 0 (CKC_CFG0)

	CPM_DRCG																															DDRC_APB + 0x084
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					Reserved																OPP_TOUT											
RST	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bits	Name	Description	RW
[27:16]	Reserved	Inner usage, keep to 0x40	RW
[11:0]	OPP_TOUT	Clock Change Opportunity time counter , if times out, clock change will be rejected.	RW

see Fig1.2.24 Clock-Change-Timing for more details.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2.2.28 Clock Change Configure 1 (CKC_CFG1)

	CPM_DRCG																DDRC_APB + 0x088															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														OPP_MODE					POST_TOUT								PRE_TOUT					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0

Bits	Name	Description	RW
[18:16]	OPP_MODE	Defines the opportunity for Clock Change 000: Disable Clock Change, the request from CPM will never be responded (neither acknowledged nor rejected). In LPDDR2: 001: When CKE == 0, clock change request from CPM will be responded (either acknowledged or rejected) 010: When CKE == 1, clock change request will be responded. 011: When DRAM in self-refresh mode, clock change request will be responded. 100~111: invalid value, treat as 000 In Other DRAM mode: 001~100: Only in self-refresh mode, the request from CPM will be responded.	RW
[13:8]	POST_TOUT	Clock Change ACK post time	RW
[5:0]	PRE_TOUT	Clock Change ACK prepare time	RW

The register CKC_CFG0 and CKC_CFG1 define how the DDR responds to request of clock frequency change from CPM.

After CKC request raised, the DDRC will check when the opportunity is met, if it was, DDRC acknowledge the CKC request, after the acknowledge, CPM can change the clock frequency. After clock changing, CPM should dessert the CKC request. After POST_TOUT, ACK from DDRC will be dessert. After this handshake, system is safe to access DDR.

Please Note that, the DLL might need to be soft reset if DLL was in use.

In case of CKC request being acknowledged:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

In case of CKC request being rejected:

Fig1.2.24 Clock-Change-Timing

16.2.2.29 CPM_DRCG

	CKC_CFG0																															0xB00000D0	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														DLLRESET	CFGCLKEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bits	Name	Description	RW
[31:2]	Reserved	Inner usage, keep in default value	
[1]	DLLRESET	1: reset the DLL in DDR PHY 0: disable the reset.	RW
[0]	CFGCLKEN	Inner usage, keep to 1	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.3 Functional Description

16.3.1 DDRC and DDR2 Memory Initialization Sequence

The following content is just an example for initialization, the sequence is already implementation in our BSP。

Example

One 512Mb x16 DDR2 device connected on CS0;
No memory device connected on CS1;
DCK = 133 MHz, CL = 3.

1 After system reset, wait system clock stable before initialize ddrc.
2 Configure the Clock-Control module for ddrc clocks.
3 DDR Memory device need at least 200us initialization time after power-on before it can accept any command.

//----------
// INIT DDRC
//----------
4 Configure DCFG = 0x.
5 Configure DTIMING1~6 = 0x.
6 Configure DMMAP0 = 0x.
7 Configure DMMAP1 = 0x0000FF00.
//----------
// INIT DDR PHY
//----------
8 Configure DDR PHY and finish PHY training process.(relate to DDRPHY spec)
//----------
// DDRC performance control configure (optional)
//----------
9 Configure RCMDCTRL0 = 0x08080808 (optional)
10 Configure RCMDCTRL1 = 0x00000808 (optional)
11 Configure WDATTHD0 = 0x08080808 (optional)
12 Configure WDATTHD1 = 0x00000808 (optional)
//----------
// INIT DDR memory device
//----------
13 Set CKE Pin HIGH : Configure DCTRL = 0x00000002.
//----------
// Enable Refresh Counter
//----------
14 Enable Refresh Counter : Configure DREFCNT = 0x.
15 AUTO-REFRESH : Configure DCTRL = 0x.
//----------
// END INITIALIZING SEQUENCE

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

```
//-----------------------------------------------------------------------------------------------------------
```

16.4 Change Clock Frequency

To reduce power consumption, the system clock frequency may be changed frequently according to the application. There are 3 ways to change the clock frequency.

16.4.1 Manually SELF-REFRESH Mode

DDR can stay in SELF-REFRESH & DEEP-POWER-DOWN mode for a long period of time. System clock frequency can be changed during this time. Even more, the clocks to DDRC module can also be stopped to save power-consumption.

Reference Sequence:

1. Manually issue SELF-REFRESH command to DDR.
2. Change relative register in CPM.
3. Change system clock frequency.
4. Drive DDR exit SELF-REFRESH mode.

16.4.2 CPM driven SELF-REFRESH Mode

CPM will auto drive DDRC to self-refresh mode, when use ddr2, ddr3.

To change clock frequency, please refer CPM spec.

16.4.3 DLL bypass mode

The clock frequency on which DLL in ddr phy works much faster than 200MHz, As the result, when ddr clock is failed to meet this condition (for example, work in 12MHz) , the DLL must be bypassed in order to supply clock to ddr controller. There are two ways to bypass the DLL:

Way 1:

```
Step1, Configure DLL into bypass mode:
    DDRP_ACDLLCR.DLLDIS <= 1'b1;
    DDRP_DXnDLLCR.DLLDIS <= 1'b1;
Step2, Configure PIR to bypass DLL
    DDRP_PIR.DLLBYP <= 1'b1;
Step3, Disable LP handshake with DLL
    DDRP_DSGCR.LPDLLPD <= 1'b0;
```

Way 2:

```
Step1, Configure DLL into reset mode to guarantee a driving clock.
        (0xB00000D0) CPM_DRCG[1] <= 1'b1;
Step2, Configure DLL into bypass mode.
        DDR_ACDLLCR.DLLDIS <= 1'b1;
```


Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Then, (0xB00000D0) CPM_DRCG[1] <= 0;

16.5 Data Endian

Fix to little Endian.

16.6 Retention Flow

16.6.1 Enter Retention mode

Description:

S0: Prepare Retention Program

Before perform retention enter flow, the program for this flow should be mapped to on chip RAM (such as TCSM in PDMA), during retention mode, any content in DRAM is not accessible. The PC of CPU should jump out of DDR range to on chip RAM

S1: Disable All Bus

Disable all device which addressing DDR, only DDRC APB/ DDRP APB port can be active.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

S2: Put DRAM in self-refresh mode

there are two ways to drive DRAM into self-refresh

1. manually configuration (please refer DCTRL.SR)
2. Auto-self-refresh (AUTOSR_EN) (*recommended*)

polling the DDRC_STATUS.ddr_sr_mode to make sure the DRAM is in self-refresh mode

S3: Read ZQ value and reserve it future use.

(refer: M200_ddrphy_spec: ZQnSR0.ZCTRL)

S4: Enable signal isolation

PHYRET_CFG.cfg_iso_en <= 1

PHYRET_CFG.clk_iso_en <= 1

S5: Assert DDR PHY Retention

PHYRET_CFG.ret_en <= 1

PHYRET_CFG.ret_en_n <= 0

S6: Remove Power

please refer Power off sequence

S7: Further Operation for power saving

- Configure CPM, gating DDR clock
- Others

Now, the DDR is safely in retention mode, any transfer to DDR from CH0~CH6 is forbidden. Please refer 'Exit Retention mode' for wake DDR

Please note that, the retention mode is only used in system sleep mode, which still powering on GPIO. Because GPIO is must for exiting retention operation.

16.6.2 Exit Retention mode

Soft ware control flow

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Description:

S1: Ready for Power-On

S2: Restore DDR Power

Please refer Sequence of DDR Power up.

S3: Guarantee clock

- Turn off CPM: ddr gating if it is gated in '*Enter Retention Mode: S7*'.
- Disable PHY clock isolation
  PHYRET_CFG.clk_iso_en <= 0
- Disable Auto-clk-stp mode
  CLKSTP.clkstp_en <= 0

S4: Soft reset DDR PHY

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1) PHYRST_CFT.*_reset <= 1
2) Delay for some cycles
3) PHYRST_CFT.*_reset <= 0

S5: Reconfigure PUBL registers

Please refer '*M200_ddrphy_spec*'

S6: Set ZQ bypass enable

DDR_PHY_PIR.ZCLBYP <= 1

S7: Polling register

DDR_PHY_PGSR.IDONE = 1

S8: Override ZQ

DDR_PHY_ZQnCR0.ZDEN <= 1 + DDR_PHY_ZQnCR0.ZDATA <= previous value (in '*Enter Retention Mode: S3*').

S9: De-assert ret_en

PHYRET_CFG.ret_en <= 0 + PHYRET_CFG.ret_en_n <= 1

S10: Override ZQ disable

DDR_PHY_ZQnCR0.ZDEN <= 0

S11: Trigger ZQ Calibration

DDR_PHY_PIR.ZQCAL <= 1 + DDR_PHY_PIR.INIT <= 1

S12: Polling register

DDR_PHY_PGSR.IDONE = 1

S13: Grant DFI bus to DDRC

DDR_PHY_PIR.CTLDINIT <= 1 + DDR_PHY_PIR.INIT <= 1

S14: Polling register

DDR_PHY_PGSR.IDONE = 1

S15: Disable signal isolation

PHYRET_CFG.cfg_iso_en <= 0

S16: Exit self-refresh

- If system works in Auto-self-refresh mode, disable this function
  AUTOSR_EN <= 0
- if works in manually self-refresh mode. Configure register to exit
  DCTL.SR <= 0

S17: Polling register to ensure ddrc exit self-refresh mode

DDRC_STATUS.ddr_sr_mode = 0

S18: Perform Data training

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Please note that it will override some data in DDR, so make sure the training address is not in use.

DDR_PHY_PIR.DQTRN <= 1 + DDR_PHY_PIR.DRAMINIT <= 1

S19: Polling register

DDR_PHY_PGSR.DTDONE = 1 + DDR_PHY_PGSR.DTERR = 0

S20: Reconfigure DDRC feature:

eg:

- AUTODR_EN <= 1
- CLKSTP_CFG.clkstp_en <= 1
- ..... others .....

S21: Finish

DDR is thoroughly get out of Retention mode. Change PC to DDR for normal operation.

16.7 DDR Connection Diagrams

The following diagrams give examples on the connection to external DDR2 devices.
Note not all the possible connections are listed.

16.7.1 Connection to one 512Mb x16 DDR2 device

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.7.2 Connection to two 512Mb x16 DDR2 devices

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

17 NAND Flash Interface

17.1 Overview

The Nand Flash Interface (NFI) divides the off-chip memory space and outputs control signals complying with specifications of various types of static memory and bus interfaces. It enables the connection of static memory such as conventional NAND flash memory, Toggle NAND flash memory, ONFI DDR flash memory, SRAM/NOR Flash memory etc. to this processor.

- Static memory interface
  - Support 3 external chip selection CS3~1#. Each bank can be configured separately
  - The size and base address of static memory banks are programmable
  - Direct interface to 8-bit/16-bit bus width external memory interface devices or external static memory to each bank. Read/Write strobe setup time and hold time periods can be programmed and inserted in an access cycle to enable connection to low-speed memory
  - Wait insertion by WAIT pin
  - Automatic wait cycle insertion to prevent data bus collisions in case of consecutive memory accesses to different banks, or a read access followed by a write access to the same bank

- NAND flash interface
  - Support on CS3~CS1, sharing with static memory bank3~bank1
  - Support both of conventional NAND flash memory and Toggle NAND flash memory/ONFI NAND flash memory
  - Support most types of NAND flashes, 8-bit/16-bit data access, 512B/2K/4K/8KB page size. For 512B page size, 3 and 4 address cycles are supported. For 2K/4K/8KB page size, 4 and 5 address cycles are supported
  - Support read/erase/program NAND flash memory
  - Support boot from NAND flash
  - 8-bit NAND support word/h-word/byte bus operation by CPU access, 16-bit NAND support word/h-word bus operation by CPU access
  - Support 8-bit common nand(not need CS mode) work together with 8-bit SRAM, 16-bit common nand(not need CS mode) work together with 16-bit SRAM/nor, Not Support toggle nand work together with SRAM

17.2 Pin Description

Following table list the NFI pins.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 17-1 NFI Pin Description

Pin Name	I/O	Signal	Description
Data Bus	I/O	SD15 – SD0	Data I/O.
Address bus	O	SA5,SA4,SA3,SA2,SA1(ALE),SA0(CLE)	Address output.
Static chip select 3 ~ 1	O	CS3~1#	Chip select signal that indicates the static bank being accessed.
Read enable	O	RD# /	For Static memory read enable signal.
Write enable	O	WE# /	Static memory write enable signal.
Wait	I	Wait# /	External wait state request signal for memory-like devices.
NAND flash read enable	O	FRE#	NAND flash read enable signal.
NAND flash write enable	O	FWE#	NAND flash write enable signal.
NAND flash ready/busy	I	FRB#	Indicates NAND flash is ready or busy.
Toggle NAND/ONFI DDR flash data strobe	I/O	DQS	Toggle/ONFI NAND data output is aligned with DQS falling/rising edge, and data inputs at DQS falling/rising.

17.3 Physical Address Space Map

Both virtual spaces and physical spaces are 32-bit wide in this architecture. Virtual addresses are translated by MMU into physical address which is further divided into several partitions for static memory, SDRAM, and internal I/O.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 17-1 Physical Address Space Map

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 17-2 Physical Address Space Map

Start Address	End Address	Connectable Memory	Capacity
0x00000000	0x0FFFFFFF	SDRAM Memory	256MB
0x10000000	0x13FFFFFF	Internal I/O Devices	64MB
0x14000000	0x14FFFFFF	Reserved	16MB
0x15000000	0x15FFFFFF	Reserved	16MB
0x16000000	0x1600003F	Reserved	64B
0x16000040	0x16FFFFFF	Reserved	
0x17000000	0x1700003F	Reserved	64B
0x17000040	0x17FFFFFF	Reserved	
0x18000000	0x1800003F	Reserved	64B
0x18000040	0x18FFFFFF	Reserved	
0x19000000	0x1900003F	Static Memory, CS3#	64B
0x19000040	0x19FFFFFF	Reserved	
0x1A000000	0x1A00003F	Static Memory, CS2#	64B
0x1A000040	0x1AFFFFFF	Reserved	
0x1B000000	0x1B00003F	Static Memory, CS1#	64B
0x1B000040	0x1BFFFFFF	Reserved	
0x1C000000	0x1FBFFFFF	Reserved	
0x1FC00000	0x1FC03FFF	On-chip Boot ROM (32kB)	32KB
0x1FC04000	0x1FFFFFFF	Reserved	
0x20000000	0xDFFFFFFF	SDRAM Memory	3072MB
0xE0000000	0xFFFFFFFF	Reserved	512MB

The base address and size of each memory banks are configurable. Software can re-configure these memory banks according to the actual connected memories. Following table lists the default configuration after reset.

Table 17-3 Default Configuration of NFI Chip Select Signals

Chip-Select Signal	Connected Memory	Capacity	Memory Width *1	Start Address	End Address
CS1#	Static memory bank 1	64 B	8, 16	0x1B000000	0x1B00003F
CS2#	Static memory bank 2	64 B	8, 16	0x1A000000	0x1A00003F
CS3#	Static memory bank 3	64 B	8, 16	0x19000000	0x1900003F

NOTES:

1 Data width of static memory banks can be configured to 8, 16 bits by software.

2 The 32KB address space from H'1FC00000 to H'1FC08000 in bank 0 is mapped to on-chip boot ROM. The other memory spaces in bank 0 are not used.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.4 NAND Register Description

NAND flash can be connected to static memory bank 3~ band 1. A mechanism for booting from NAND flash is also supported.

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

17.4.1 Register Description

Table 17-4 Registers Memory Map-Address Base

Name	Base	Description
NFI	0x13410000	Address base of NFI

Table 17-5 NAND Flash Interface command and address Registers

Name	Description	Reset Value	Address	Access Width
NFCM0	NAND FLASH Command0	0x00000000	0x1b400000	32
NFCM1	NAND FLASH Command1	0x00000000	0x1a400000	32
NFCM2	NAND FLASH Command2	0x00000000	0x19400000	32
NFAD0	NAND FLASH Address0	0x00000000	0x1b800000	32
NFAD1	NAND FLASH Address1	0x00000000	0x1a800000	32
NFAD2	NAND FLASH Address2	0x00000000	0x19800000	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 17-6 NAND Flash Interface Registers

Name	Description	Reset Value	Address offset	Access Width
NFCE	NAND FLASH Chip Select	0x00000000	0x08	32
NFBC	NFI FLASH Busy Control Register	0x0fc00000	0x0c	32
NFCR	NFI Enable Register	0x003f0841	0x10	32
PNCR	NAND PN Control Register	0x00000000	0x14	32
PNDR	NAND PN Data Register	0x00000000	0x18	32
BCNT	NAND Bit Counter	0x00000000	0x1c	32
NFDL	NAND FLASH DQS Delay Line	0x00000000	0x20	32
NFI_T0	NAND FLASH Timing0	0x00030003	0x28	32
NFI_T1	NAND FLASH Timing1	0x00030003	0x2c	32
NFI_T2	NAND FLASH Timing2	0x00030003	0x30	32
NFI_T3	NAND FLASH Timing3	0x00030003	0x34	32
NFI_T4	NAND FLASH Timing4	0x00000000	0x38	32
NFI_TG0	Toggle NAND FLASH Timing0	0x00000000	0x3c	32
NFI_TG1	Toggle NAND FLASH Timing1	0x00000000	0x40	32
NFI_OF0	ONFI NAND FLASH Timing0	0x00000000	0x44	32
NFI_OF1	ONFI NAND FLASH Timing1	0x00000000	0x48	32
NFBS0	NAND FLASH Busy Detector0	0x08040201	0x4c	32
NFBS1	NAND FLASH Busy Detector1	0x00020010	0x50	32
NFRB	NAND FLASH BUSY FIR	0x0000001f	0x54	32

17.4.1.1 NAND FLASH Command (NFCM 0~2)

NFCM0~2 is a 32-bit write only register that is used to send a command to NAND Flash memory.

Figure 17-2 Basic Write Command to NAND FLASH

	NFCM0																															**0x1B400000**
	NFCM1																															**0x1A400000**
	NFCM2																															**0x19400000**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLY_F											DLY_B											Reserved	BUSY	CMD							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:21	DLY_F	**Extra Delay Time before CLE goes to high.**	W
20:10	DLY_B	**Extra Delay Time after CLE goes to low.** NOTE: ONLY Valid for Asynchronous Nand & Toggle Nand;	W
9	Reserved	Writing has no effect, read as zero.	W
8	BUSY	**Wait Busy Start:** If a RB_ come with this command, software can set this bit high. When RB_ goes to high, controller will set NFCR.BUSY(x) to 1.	W
7:0	CMD	**Nand Flash Command**	W

17.4.1.2 NAND FLASH Address (NFAD 0~2)

NFAD0~2 is a 32-bit write only register that is used to send a address to NAND Flash memory.

Figure 17-3 Basic Write Address to NAND FLASH

	NFAD0																															**0x1B800000**
	NFAD1																															**0x1A800000**
	NFAD2																															**0x19800000**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLY_F											DLY_B											Reserved	BUSY	ADDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:21	DLY_F	**Extra Delay Time before CLE goes to high.**	W
20:10	DLY_B	**Extra Delay Time after CLE goes to low.** NOTE: ONLY Valid for Asynchronous Nand & Toggle Nand;	W
9	Reserved	Writing has no effect, read as zero.	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8	BUSY	**Wait Busy Start:** If a RB_ come with this command, software can set this bit high. When RB_ goes to high, controller will set NFCR.BUSY(x) to 1.	W
7:0	ADDR	**Nand Flash Address**	W

17.4.1.3 NAND FLASH Chip Select (NFCS)

NFCS Controls the assertion of NAND Flash Chip Select.

	NFCS																															BASE + 0x08
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										Reserved	Reserved	Reserved	CS3	CS2	CS1
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5	Reserved	Inner usage, keep to 0	RW
4	Reserved	Inner usage, keep to 0	RW
3	Reserved	Inner usage, keep to 0	RW
2	CS3	**Chip Select 3**	RW
1	CS2	**Chip Select 2**	RW
0	CS1	**Chip Select 1:** **NOTE:** If need change NFCS, you **MUST** set NFCS to 6'b000000 first; **For example**, CS1(000001)->(000000)->CS3(000100)	RW

NOTE:

(1) For Toggle DDR Nand: after Write/Read Data(Do NOT Include CMD/ADDR), CS MUST be Cleared before any Access;(Clear CS: NFCS.CS0=1, then set NFCS.CS0 to 0)

(2) For ONFI DDR Nand: Before a Data Read Access, CS MUST be Set; (Set CS: NFCS.CS0=1, then set NFCS.CS0=1 again)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

After a Data Write/Read Access, CS MUST be Set; (Set CS: NFCS.CS0=1, then set NFCS.CS0=1 again)

17.4.1.4 NFI Enable Register (NFCR)

NFCR is a 32-bit read/write register that is used to configure NFI controller. It is initialized by any reset.

	NFCR																															BASE + 0x10
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RESERVED										CSn_SEL						CM_CS	RESERVED			EMPTY	Reserved	Reserved	RESERVED		IDLE	16X	SEL			INIT	EN
RST	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1

Bits	Name	Description	RW
31:22	Reserved	Writing has no effect, read as zero.	R
21:16	CS(n)_SEL (n = 0~5)	Chip(n) Select : 0: NAND Flash 1: non-NAND Flash	RW
15	CM_CS	Only used for common nand : 0: common nand do not need NFCS 1:common nand need NFCS	RW
14:12	Reserved	Writing has no effect, read as zero.	R
11	EMPTY	**NFI Empty**: All Command/Address/Write Data/Read Data process done.	R
10	Reserved	Inner usage, keep to 0	RW
9	Reserved	Inner usage, keep to 0	RW
8:7	Reserved	Writing has no effect, read as zero.	R
6	IDLE	NFI is IDLE When Write -> Read, before Read Access MUST make sure IDLE =1	R
5	NAND_16X	**16bit NAND:** 0: 8 bit NAND 1: 16 bit NAND(ONLY for Common Nand Interface)	RW
4:2	SEL	NAND Flash Interface Select 000: Common Nand 001: Common Nand EDO Mode 010: Toggle DDR Nand	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		011: ONFI DDR Nand	
1	INIT	Initial NFI(i.e. soft reset, need software to delay at list 3 ahb-clock)	W1
0	EN	NFI Enable	RW

17.4.1.5 NFI FLASH Busy Control Register (NFBC)

NFCR is a 32-bit read/write register that is used to configure NFI busy. It is initialized by any reset.

	NFBC																															BASE + 0x0c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RESERVED				BUSY_M6	BUSY_M5	BUSY_M4	BUSY_M3	BUSY_M2	BUSY_M1	BUSY6	BUSY5	BUSY4	BUSY3	BUSY2	BUSY1	RESERVED							BUSY_MOD		RESERVED						
RST	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:22	BUSY_M(x)	**BUSY Int Mask**	WR
21:16	BUSY(x)	**Busy Detect Done:** If a RB_ pose-edge detected by NFI, the coordinate bit will goes to high by hardware, and Write 1 to clear.	RWC
15:9	Reserved	Writing has no effect, read as zero.	R
8:7	BUSY_MOD	**BUSY Mode Select:** 00: negedge->posedge 01: posedge only 10: negedge only	WR
6:0	Reserved	Writing has no effect, read as zero.	R

17.4.1.6 NAND PN Control Register (PNCR)

PNCR is a 32-bit read/write register that is used to control NAND flash data randomization. It is initialized by any reset.

	PNCR																															BASE + 0x14
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNWDA																Reserved								TEST_EN	Reserved		BIT_RST	BIT_SEL	BIT_EN	PNRST	PNEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	PNWDA	**Write data after PN:** PNWDA is enable when TEAT_EN is enable. This bit is read-only	R
15:8	Reserved	Writing has no effect, read as zero.	R
7	TEST_EN	**TEST Enable:** ONLY used by software 0 Test is disable(initial) 1 Test is enable	RW
6:5	Reserved	Writing has no effect, read as zero.	R
4	BIT_RST	**NAND Read BIT Counter Reset:** Reset Bit counter. When this bit is written to 1, the bit counter is reset to 0. This bit is write-only.	W
3	BIT_SEL	**NAND Read BIT Counter Select:** Bit counter's counting select. **BIT_SEL Description** 0 Calculate number of "0" in NAND read data 1 Calculate number of "1" in NAND read Data	RW
2	BIT_EN	**NAND Read BIT Counter Enable:** Enable/disable bit counter counting. **BIT_EN Description** 0 Bit counting is disabled 1 Bit counting is enabled	RW
1	PNRST	**NAND Flash PN Reset:** Reset seed of randomizer. When this bit is written to 1, the seed of randomizer is reset. This bit is write-only.	W
0	PNEN	**NAND Flash PN Enable:** Specifies if NAND flash read/write data randomization is enabled. This bit is initialized to 0. **PNEN Description** 0 Data randomization is disabled 1 Data randomization is enabled	RW

17.4.1.7 NAND PN Data Register (PNDR)

PNDR is a 23-bit read/write register that is used for seed of randomizer during NAND read/write.

	PNDR																															BASE + 0x18
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved									PNDR																						
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1

17.4.1.8 NAND Bit Counter (BCNT)

BCNT is a 32-bit read register that is used to counting the number of "1" or "0" (based on BIT_SEL) in Nand read data and keep counting during Nand read till BIT Counter Reset. It is initialized by any reset.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	BCNT																																BASE + 0x1c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BCNT																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

17.4.1.9 NAND Flash DQS Delay Line (NFDL)

NFDL contains an on-chip DLL to control the DQS Delay for read data.

	NFDL																																BASE + 0x20
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RESERVED		ERR	DONE	RCNT				DET	AUTO	RESERVED	ADQS					RESERVED			IDQS					RESERVED			RDQS				
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29	ERR	**DQS Delay Detect ERROR:** When hardware detect one ahb_clk cycle delay failed, ERR is set high; ERR is cleared zero when a new detection starts; ERR is valid only when DONE=1.	R
28	DONE	**Delay Detect Done:** When hardware detect complete, ERR is valid. DONE is cleared zero when a new detection starts.	R
27:24	RCNT	**Re-Detect Wait Cycle.**	RW
23	DET	**Start Delay Detecting:** Write 1 to START bit starts a new delay detect progress.	W
22	AUTO	**Hardware Auto-detect & Set Delay Line.** 0: Hardware does NOT auto-set delay line. 1: Hardware auto-set delay line after detect success.	RW
21	Reserved	Writing has no effect, read as zero.	R
20:16	ADQS	**Number of Delay Elements Detect by Hardware**	R
15:13	Reserved	Writing has no effect, read as zero.	R
12:8	IDQS	**Initial Number of Delay Elements Used on Auto Detect**	RW
7:5	Reserved	Writing has no effect, read as zero.	R
4:0	RDQS	**Number of Delay Elements Used on the Read DQS Delay-line**:	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Set the number of delay elements used on the read DQS delay-line. When RDQS increase one, the delay value of read DQS increase approximately 0.3 ns. The range of RDQS: 0 to 31.	

Example of DQS Delay Line Auto-Detect-Operation

17.4.1.10 NAND Flash Timing0 (NFI_T0)

	NFI_T0																BASE + 0x28															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tSET_WE																tWAIT_WE															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:16	tSET_WE	**Number of clock cycles+1 (HCLK) to set up the WE** Min~Max:0x0~0xfffe; Example: 0 : 1 clock 1 : 2 clocks	RW
15:0	tWAIT_WE	**Defines the minimum number of HCLK clock cycles to assert WE, Can NOT be set as 0;** Min~Max:0x1~0xfffe;	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Example:

tWH = tSET_WE+tHOLD_WE+2;
tWP = tWAIT_WE;
tREH = tSET_RE+tHOLD_RE+2;
tRC - tREH = tWAIT_RE;

Figure 17-4 Basic Write/Read NAND FLASH

17.4.1.11 NAND Flash Timing1 (NFI_T1)

	NFI_T1																BASE + 0x2C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tHOLD_WE																tSET_RE															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:16	tHOLD_WE	**Defines the number of HCLK clock cycles(+1) to hold the WE;** Min~ Max : 0x0~0xfffe;	RW
15:0	tSET_RE	**Number of clock cycles(+1) to set up the RE;** Min~Max:0x0~0xfffe; **Example:** 0 : 1 clock 1 : 2 clocks	RW

17.4.1.12 NAND Flash Timing2 (NFI_T2)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	NFI_T2																															BASE + 0x30
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tWAIT_RE																tHOLD_RE															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:16	tWAIT_RE	**Defines the minimum number of HCLK clock cycles to assert RE, Can NOT set as 0;** Min~Max:0x1~0xfffe;	RW
15:0	tHOLD_RE	**Defines the number of HCLK clock cycles(+1) to hold the RE;** Min~ Max : 0x0~0xfffe;	RW

17.4.1.13 NAND Flash Timing3 (NFI_T3)

	NFI_T3																															BASE + 0x34
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tSET_CS																tWAIT_CS															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:16	tSET_CS	**Chip Select Setup Time:** Can NOT Set As 0	RW
15:0	tWAIT_CS	**Chip Select Wait Time:** Can NOT Set As 0	RW

17.4.1.14 NAND Flash Timing4 (NFI_T4)

	NFI_T4																															BASE + 0x38
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tBUSY																tEDO															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	tBUSY	**Ready to RE High Time:** e.g. tRR;	RW
15:0	tEDO	**Nand Flash EDO Mode Delay For Read Data** Can NOT Set As 0	RW

Figure 17-5 RB_ posedge Detected by NFI, with tBUSY

Figure 17-6 Read Data By Sel=0/1(EDO mode)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.4.1.15 Toggle NAND Flash Timing0 (NFI_TG0)

	NFI_TG0																															BASE + 0x3c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLY_F_RE_TG								DLY_F_DQS_TG								tSET_DQS_TG								tHOLD_DQS_TG							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	DLY_F_RE_TG	**First Toggle Nand Data Read Delay;**	RW
23:16	DLY_F_DQS_TG	**First Toggle Nand Data Write Delay;**	RW
15:8	SET_DQS_TG	**Number of clock cycles+1 (HCLK) to set up the DQS** Valid value: 0x0 to 0xfffe;	RW
7:0	HOLD_DQS_TG	**Defines the number of HCLK clock cycles(+1) to hold the DQS;** Valid value: 0x0 to 0xfffe;	RW

NOTE:

DQS : default is 1, if the Write Process is Done CS MAST be Clear immediately, and then DQS will return to 1;

RE: default is 1, if the Read Process(DQS is Hi-z) is Done CS MAST be Clear immediately, and after DLY_B_DQS_TG DQS will return to 1;

Figure 17-7 Basic Toggle Nand Write, with CS disable

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 17-8 Basic Toggle Nand Read, with CS disable

17.4.1.16 Toggle NAND Flash Timing1 (NFI_TG1)

	NFI_TG1																															BASE + 0x40
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																tRE_DQS2IDLE								tDQSRE_TG							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:8	RE_DQS2IDLE	**RE & DQS_o goes to High when CS clear;** **NOTE:** Make sure that tRE_DQS2IDLE < tWAIT_CS+1	RW
7:0	DQSRE_TG	**Number of clock cycles+1 (HCLK) to set up the DQS_I_mask** Valid value: 0x1 to 0xfe	RW

17.4.1.17 ONFI NAND Flash Timing0 (NFI_OF0)

	NFI_OF0																															BASE + 0x44
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	tWR2CLE_HI								tDQSS(write delay)								tSET_CLK								tHOLD_CLK							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	WR2CLE_HI	**WR to CLE Goes To High:** Can NOT be 0	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		MUST > 20 ns(e.g. tDQSD)	
23:16	DQSS	tDQSS = (tSET_CLK+tHOLD_CLK+2)*2 0: bypass 1:1 nfi_clk delay for DQS_o & DQ_o; ... 15:15 nfi_clk delay for DQS_o & DQ_o;	RW
15:8	SET_CLK	**Number of clock cycles+1 (HCLK) to set up the CLK** Min~Max:0x0~0xfffe; Example: 0 : 1 clock 1 : 2 clocks	RW
7:0	HOLD_CLK	**Defines the number of HCLK clock cycles(+1) to hold the CLK;** Min~ Max : 0x0~0xfffe;	RW

Figure 17-9 ONFI NAND Clock Timing

Figure 17-10NFI CMD/ADDR Timing

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 17-11 ONFI NAND Write Data Timing

Figure 17-12 ONFI NAND Read Data Timing

17.4.1.18 ONFI NAND Flash Timing1 (NFI_OF1)

	NFI_OF1																															BASE + 0x48
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																tDQS2IDLE								tR2IDLE							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:0	DQS2IDLE	**When DQS2IDLE count done, DQS and DQ goes to Input state(Hi-z);** **Start at CS begin;** **Can NOT be 0;**	RW
7:0	R2IDLE	**When R2IDLE count done, W/R_n goes to Hi ;** **Start at CS begin;** **Can NOT be 0;**	RW

17.4.1.19 NAND FLASH Busy Detector0 (NFBS0)

	NFBS0																															BASE + 0x4c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1	Reserved		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1	Reserved		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1	Reserved		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1
RST	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:24	CB4	FRB Detector4 Select	RW
23:22	Reserved	Writing has no effect, read as zero.	R
21:16	CB3	FRB Detector3 Select	RW
15:14	Reserved	Writing has no effect, read as zero.	R
13:8	CB2	FRB Detector2Select	RW
7:6	Reserved	Writing has no effect, read as zero.	R
5:0	CB1	FRB Detector1 Select	RW

Figure 17-13 Select Both of FLASH1.RB_ and FLASH2.RB_ to BUSY Detector1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.4.1.20 NAND FLASH Busy Detector1 (NFBS1)

	NFBS1																															BASE + 0x50
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1	Reserved		FRB6	FRB5	FRB4	FRB3	FRB2	FRB1
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0

Bits	Name	Description	RW
31:14	Reserved	Writing has no effect, read as zero.	R
13:8	CB6	FRB Detector6 Select	RW
7:6	Reserved	Writing has no effect, read as zero.	R
5:0	CB5	FRB Detector5 Select	RW

17.4.1.21 NAND FLASH Busy Fir (NFRB)

NFRB is used to FRB filter.

	NFBS1																															BASE + 0x54
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											FIR_NUM				FIR_EN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

Bits	Name	Description	RW
31:5	Reserved	Writing has no effect, read as zero.	R
4:1	FIR_NUM	Filter Cycle number	RW
0	FIR_EN	FIR Enable	RW

17.4.2 NAND Flash Boot Loader

To support boot from NAND flash, 32KB on-chip Boot ROM is implemented. Following figure illustrates the structure of NAND Flash Boot Loader.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 17-14 Structure of NAND Flash Boot Loader

When system is configured to boot from NAND flash, after reset, the program in Boot ROM is executed and the program will copy the first 8K bytes of NAND flash to internal memory for further initialization.

Generally, the boot code will copy more NAND flash content to DRAM. Then the main program will be executed on DRAM.
When system is configured to boot from NAND flash, software may know the nand flash page size through BOOT_SEL[2:0] pin.

17.4.3 NAND Flash Operation

The partition of static bank n (n=1~3) is changed as following figure. Writes to any of address space will be translated to NAND flash address cycle. Writes to any of command space will be translated to NAND flash command cycle.

CAUTION:

1. Don't read to address and command space, and these two partitions should be uncacheable. Reads and writes to any of data space will be translated to NAND flash data read/write cycle. DMA access to data space is supported to increase the speed of data read/write. The DMA access cannot exceed the page boundary (512 bytes or 2K bytes) of NAND.
2. The least significant bit of the column address shall always be zero for a Toggle interface, i.e. an **EVEN** number of bytes is always transferred.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 17-15 Static Bank 1 Partition When NAND Flash is Used (an example)

address 0x100000d0 bit[0]=1

Common NAND CS operation has two mode :

1) Configure NAND FLASH Chip Select Register(NFCS) , need

Set CM_CS=1 of NFI Enable Register(NFCR)

Set CS(n) of NAND FLASH Chip Select(NFCS)

2) Not need Configure NFCS Register, only need select bank n address.

ONFI/TOGGLE NAND need

Set CM_CS=1 of NFI Enable Register(NFCR)

Set CS(n) of NAND FLASH Chip Select(NFCS)

The control signals for direction connection of NAND flash are CS(n)#, FRE#, FWE#, FRB#(GPIO), A1 and A0. Following figure shows the connection between processor and NAND Flash.

Figure 17-16 Example of 8-bit NAND Flash Connection

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.5 Static Memory Interface

17.5.1 Register Description

Table 17-7 Registers Memory Map-Address Base

Name	Base	Description
NFI	0x13410000	Address base of NFI

Table 17-8 NAND Flash Interface command and address Registers

Name	Description	Reset Value	Address offset	Access Width
SACR1	Static memory bank 1 address configuration register	0x00001BFF	0x0104	32
SACR2	Static memory bank 2 address configuration register	0x00001AFF	0x0108	32
SACR3	Static memory bank 3 address configuration register	0x000019FF	0x010c	32
SM0_T0	Static memory bank1 control register 0	0x00030303	0x011c	32
SM1_T0	Static memory bank2 control register 0	0x00030303	0x0120	32
SM2_T0	Static memory bank3 control register 0	0x00030303	0x0124	32
SM0_T1	Static memory bank1 control register 1	0x00030303	0x0134	
SM1_T1	Static memory bank2 control register 1	0x00030303	0x0138	
SM2_T1	Static memory bank3 control register 1	0x00030303	0x013c	

17.5.1.1 Static Bank Address Configuration Register (SACR1~3)

SACR1~3 defines the physical address for static memory bank 1 to 3, respectively. Each register contains a base address and a mask. When the following equation is met:

(*physical address* [31:24] & **MASK** $_{(n)}$) == **BASE**$_{(n)}$

The bank n is active. The *physical address* is address output on internal system bus. Static bank regions must be programmed so that each bank occupies a unique area of the physical address space. Programming overlapping bank regions will result in unpredictable error. These registers are initialized by a reset.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SACR1 SACR2 SACR3																															BASE + 0x104 BASE + 0x108 BASE + 0x10c
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															nNAND_16X	BASE								MASK							
RST1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1
RST2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1	1	1	1	1
RST3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31:17	Reserved	Writing has no effect, read as zero.	R
16	nNAND_ 16X	**Non-NAND Interface Width:** 0: 8 bit 1: 16 bit	RW
15:8	BASE	**Address Base:** Defines the base address of Static Bank n (n = 1 to 3). The initial values are: SACR1.BASE 0x1B SACR2.BASE 0x1A SACR3.BASE 0x19	RW
7:0	MASK	**Address Mask**: Defines the mask of Static Bank n (n = 1 to 3). The initial values are: SACR1.MASK 0xFF SACR2.MASK 0xFF SACR3.MASK 0xFF	RW

17.5.1.2 Static Memory timing0 (SM_T0)

	SM0_T0 SM1_T0 SM2_T0																															BASE + 0x11c BASE + 0x120 BASE + 0x124
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								tHOLD_WE_S								tWAIT_WE_S								tSET_WE_S							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

23:16	tHOLD_WE_S	**Defines the number of HCLK clock cycles(+1) to hold the WE;** Min~ Max : 0x0~0xfffe;	RW
15:8	tWAIT_WE_S	**Defines the minimum number of HCLK clock cycles to assert WE, Can NOT be set as 0;** Min~Max:0x1~0xfffe;	RW
7:0	tSET_WE_S	**Number of clock cycles+1 (HCLK) to set up the WE** Min~Max:0x0~0xfffe; Example: 0 : 1 clock 1 : 2 clocks	RW

Example:

tWH = tSET_WE+tHOLD_WE+2;
tWP = tWAIT_WE;
tREH = tSET_RE+tHOLD_RE+2;
tRC - tREH = tWAIT_RE;

Figure 17-17 Basic Write/Read NAND FLASH

17.5.1.3 Static Memory timing1 (SM_T1)

	SM0_T1 **SM1_T1** **SM2_T1**																															**BASE + 0x134** **BASE + 0x138** **BASE + 0x13c**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								tHOLD_RE_S								tWAIT_RE_S								tSET_RE_S							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	tHOLD_RE_S	**Defines the number of HCLK clock cycles(+1) to hold the WE;** Min~ Max : 0x0~0xfffe;	RW
15:8	tWAIT_RE_S	**Defines the minimum number of HCLK clock cycles to assert WE, Can NOT be set as 0;** Min~Max:0x1~0xfffe;	RW
7:0	tSET_RE_S	**Number of clock cycles+1 (HCLK) to set up the WE** Min~Max:0x0~0xfffe; Example: 0 : 1 clock 1 : 2 clocks	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18 BCH Controller

18.1 Overview

The BCH Controller implements data ECC encoding and decoding.

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 18-1 Registers Memory Map-Address Base

Name	Base	Description
BCH	0x134D0000	Address base of BCH

Table 18-2 BCH Registers

Name	Description	RW	Reset Value	Address offset	Access Width
BHCR	BCH Control register	RW	0x00000000	0x0000	32
BHCSR	BCH Control Set register	W	0x????????	0x0004	32
BHCCR	BCH Control Clear register	W	0x????????	0x0008	32
BHCNT	BCH ENC/DEC Count register	RW	0x00000000	0x000C	32
BHDR	BCH data register	W	0x????????	0x0010	32/16/8
BHPAR0	BCH Parity 0 register	R	0x00000000	0x0014	32
BHPAR1	BCH Parity 1 register	R	0x00000000	0x0018	32
BHPAR2	BCH Parity 2 register	R	0x00000000	0x001C	32
BHPAR3	BCH Parity 3 register	R	0x00000000	0x0020	32
BHPAR4	BCH Parity 4 register	R	0x00000000	0x0024	32
BHPAR5	BCH Parity 5 register	R	0x00000000	0x0028	32
BHPAR6	BCH Parity 6 register	R	0x00000000	0x002C	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

BHPAR7	BCH Parity 7 register	R	0x00000000	0x0030	32
BHPAR8	BCH Parity 8 register	R	0x00000000	0x0034	32
BHPAR9	BCH Parity 9 register	R	0x00000000	0x0038	32
BHPAR10	BCH Parity 10 register	R	0x00000000	0x003C	32
BHPAR11	BCH Parity 11 register	R	0x00000000	0x0040	32
BHPAR12	BCH Parity 12 register	R	0x00000000	0x0044	32
BHPAR13	BCH Parity 13 register	R	0x00000000	0x0048	32
BHPAR14	BCH Parity 14 register	R	0x00000000	0x004C	32
BHPAR15	BCH Parity 15 register	R	0x00000000	0x0050	32
BHPAR16	BCH Parity 16 register	R	0x00000000	0x0054	32
BHPAR17	BCH Parity 17 register	R	0x00000000	0x0058	32
BHPAR18	BCH Parity 18 register	R	0x00000000	0x005C	32
BHPAR19	BCH Parity 19 register	R	0x00000000	0x0060	32
BHPAR20	BCH Parity 20 register	R	0x00000000	0x0064	32
BHPAR21	BCH Parity 21 register	R	0x00000000	0x0068	32
BHPAR22	BCH Parity 22 register	R	0x00000000	0x006C	32
BHPAR23	BCH Parity 23 register	R	0x00000000	0x0070	32
BHPAR24	BCH Parity 24 register	R	0x00000000	0x0074	32
BHPAR25	BCH Parity 25 register	R	0x00000000	0x0078	32
BHPAR26	BCH Parity 26 register	R	0x00000000	0x007C	32
BHPAR27	BCH Parity 27 register	R	0x00000000	0x0080	32
BHERR0	BCH Error Report 0 register	R	0x00000000	0x0084	32
BHERR1	BCH Error Report 1 register	R	0x00000000	0x0088	32
BHERR2	BCH Error Report 2 register	R	0x00000000	0x008C	32
BHERR3	BCH Error Report 3 register	R	0x00000000	0x0090	32
BHERR4	BCH Error Report 4 register	R	0x00000000	0x0094	32
BHERR5	BCH Error Report 5 register	R	0x00000000	0x0098	32
BHERR6	BCH Error Report 6 register	R	0x00000000	0x009C	32
BHERR7	BCH Error Report 7 register	R	0x00000000	0x00A0	32
BHERR8	BCH Error Report 8 register	R	0x00000000	0x00A4	32
BHERR9	BCH Error Report 9 register	R	0x00000000	0x00A8	32
BHERR10	BCH Error Report 10 register	R	0x00000000	0x00AC	32
BHERR11	BCH Error Report 11 register	R	0x00000000	0x00B0	32
BHERR12	BCH Error Report 12 register	R	0x00000000	0x00B4	32
BHERR13	BCH Error Report 13 register	R	0x00000000	0x00B8	32
BHERR14	BCH Error Report 14 register	R	0x00000000	0x00BC	32
BHERR15	BCH Error Report 15 register	R	0x00000000	0x00C0	32
BHERR16	BCH Error Report 16 register	R	0x00000000	0x00C4	32
BHERR17	BCH Error Report 17 register	R	0x00000000	0x00C8	32
BHERR18	BCH Error Report 18 register	R	0x00000000	0x00CC	32
BHERR19	BCH Error Report 19 register	R	0x00000000	0x00D0	32
BHERR20	BCH Error Report 20 register	R	0x00000000	0x00D4	32
BHERR21	BCH Error Report 21 register	R	0x00000000	0x00D8	32
BHERR22	BCH Error Report 22 register	R	0x00000000	0x00DC	32
BHERR23	BCH Error Report 23 register	R	0x00000000	0x00E0	32
BHERR24	BCH Error Report 24 register	R	0x00000000	0x00E4	32
BHERR25	BCH Error Report 25 register	R	0x00000000	0x00E8	32
BHERR26	BCH Error Report 26 register	R	0x00000000	0x00EC	32
BHERR27	BCH Error Report 27 register	R	0x00000000	0x00F0	32
BHERR28	BCH Error Report 28 register	R	0x00000000	0x00F4	32
BHERR29	BCH Error Report 29 register	R	0x00000000	0x00F8	32
BHERR30	BCH Error Report 30 register	R	0x00000000	0x00FC	32
BHERR31	BCH Error Report 31 register	R	0x00000000	0x0100	32
BHERR32	BCH Error Report 32 register	R	0x00000000	0x0104	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

BHERR33	BCH Error Report 33 register	R	0x00000000	0x0108	32
BHERR34	BCH Error Report 34 register	R	0x00000000	0x010C	32
BHERR35	BCH Error Report 35 register	R	0x00000000	0x0110	32
BHERR36	BCH Error Report 36 register	R	0x00000000	0x0114	32
BHERR37	BCH Error Report 37 register	R	0x00000000	0x0118	32
BHERR38	BCH Error Report 38 register	R	0x00000000	0x011C	32
BHERR39	BCH Error Report 39 register	R	0x00000000	0x0120	32
BHERR40	BCH Error Report 40 register	R	0x00000000	0x0124	32
BHERR41	BCH Error Report 41 register	R	0x00000000	0x0128	32
BHERR42	BCH Error Report 42 register	R	0x00000000	0x012C	32
BHERR43	BCH Error Report 43 register	R	0x00000000	0x0130	32
BHERR44	BCH Error Report 44 register	R	0x00000000	0x0134	32
BHERR45	BCH Error Report 45 register	R	0x00000000	0x0138	32
BHERR46	BCH Error Report 46 register	R	0x00000000	0x013C	32
BHERR47	BCH Error Report 47 register	R	0x00000000	0x0140	32
BHERR48	BCH Error Report 48 register	R	0x00000000	0x0144	32
BHERR49	BCH Error Report 49 register	R	0x00000000	0x0148	32
BHERR50	BCH Error Report 50 register	R	0x00000000	0x014C	32
BHERR51	BCH Error Report 51 register	R	0x00000000	0x0150	32
BHERR52	BCH Error Report 52 register	R	0x00000000	0x0154	32
BHERR53	BCH Error Report 53 register	R	0x00000000	0x0158	32
BHERR54	BCH Error Report 54 register	R	0x00000000	0x015C	32
BHERR55	BCH Error Report 55 register	R	0x00000000	0x0160	32
BHERR56	BCH Error Report 56 register	R	0x00000000	0x0164	32
BHERR57	BCH Error Report 57 register	R	0x00000000	0x0168	32
BHERR58	BCH Error Report 58 register	R	0x00000000	0x016C	32
BHERR59	BCH Error Report 59 register	R	0x00000000	0x0170	32
BHERR60	BCH Error Report 60 register	R	0x00000000	0x0174	32
BHERR61	BCH Error Report 61 register	R	0x00000000	0x0178	32
BHERR62	BCH Error Report 62 register	R	0x00000000	0x017C	32
BHERR63	BCH Error Report 63 register	R	0x00000000	0x0180	32
BHINT	BCH Interrupt Status register	RW	0x00000000	0x0184	32
BHINTES	BCH Interrupt Set register	W	0x????????	0x0188	32
BHINTEC	BCH Interrupt Clear register	W	0x????????	0x018C	32
BHINTE	BCH Interrupt Enable register	RW	0x00000000	0x0190	32
BHTO	BCH User Tag Output register	R	0x????????	0x0194	32

18.2 Register Description

18.2.1 BCH Control Register (BHCR)

BHCR is a 32-bit read/write register that is used to configure BCH controller. It is initialized by any reset.

	BHCR																															BASE + 0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAG																MZEB			BPS	Reserved	BSEL							Reserved	ENCE	INIT	BCHE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	TAG	**User-provided TAG:** Indicates a user-provided tag associated with the data block. The tag can be used to track data through the ECC module and the contents of the tag are presented on the BHTO.TAGO at the completion of the correction operation.	RW
15:13	MZEB	**MAX ZERO BIT IN ERASED BLOCK:** Indicates that the maximum number of zero bits in an erased data block. In decoding mode, if the number of zero bits less than MZEB, the BCH Controller will set BHINT. ALL_f to 1.	RW
12	BPS	**DECODER BYPASS:** Indicates that no corrections were applied to the data block since the BPS was asserted at the beginning of the correction operation. **BPS** **Description** 0 Decoder is active (Initial value) 1 Decoder is bypassed	RW
11	Reserved	Writing has no effect, read as zero.	R
10:4	BSEL	**BCH Encoding/Decoding Bit Select**: It is used to select the correction algorithm. Only multiples of 4 are supported (i.e. ECC 4, 8, 12 ... 56, 60, 64). **BSEL** **Description** 4 4-bit correction 8 8-bit correction 12 12-bit correction … 60 60-bit correction 64 64-bit correction	RW
3	Reserved	Inner usage, keep to 1	R
2	ENCE	**BCH Encoding/Decoding Select**: It is used to define whether in encoding or in decoding phase when BCH is used. **ENCE** **Description** 0 Decoding (Initial value) 1 Encoding	RW
1	INIT	**BCH Controller Initial**：Initializes BCH Controller for a new operation. This bit is cleared automatically by hardware and always read as 0.	W
0	BCHE	**BCH Enable:** BCH correction is enable/disable. **BCHE** **Description** 0 BCH is disabled (initial value) 1 BCH is enabled	RW

18.2.2 BCH Control Set Register (BHCSR)

BHCSR is a 32-bit write-only register that is used to set BCH controller to 1.

When write 1 to BHCSR, the corresponding bit in BHCR register is set to 1. Write 0 to BHCSR is

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

ignored.

	BHCSR																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAGS																MZEBS			BPSS	Reserved	BSELS							Reserved	ENCES	INITS	BCHES
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:16	TAGS	**User-provided TAG Set:** It is used to set BHCR.TAG to 1.	W
15:13	MZEBS	**MAX ZERO BIT IN ERASED BLOCK Set:** it is used to set BHCR.MZEB to 1.	W
12	BPSS	**DECODER BYPASS Set:** it is used to set BHCR.BPS to 1	W
11	Reserved	Writing has no effect, read as zero.	R
10:4	BSELS	**BCH Encoding/Decoding Bit Select Set**: It is used to set BHCR.BSEL to 1.	W
3	Reserved	Writing has no effect, read as zero.	R
2	ENCES	**BCH Encoding/Decoding Select Set**: It is used to set BHCR.ENCE to 1.	W
1	INITS	**BCH Controller Initial Set：** It is used to set BHCR.INIT to 1.	W
0	BCHES	**BCH Enable Set:** It is used to set BHCR.BCHE to 1.	W

18.2.3 BCH Control Clear Register (BHCCR)

BHCCR is a 32-bit write-only register that is used to clear BCH controller to 0.

When write 1 to BHCCR, the corresponding bit in BHCR register is cleared to 0. Write 0 to BHCCR is ignored.

	BHCCR																															BASE + 0x0008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAGC																MZEBC			BPSC	Reserved	BSELC							Reserved	ENCEC	INITC	BCHEC
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:16	TAGC	**User-provided TAG Clear:** It is used to clear BHCR.TAG to 0.	W
15:13	MZEBC	**MAX ZERO BIT IN ERASED BLOCK Clear:** It is used to clear BHCR.MZEB to 0.	W
12	BPSC	**DECODER BYPASS Clear:** It is used to clear BHCR.BPS to 0	W
11	Reserved	Writing has no effect, read as zero.	R
10:4	BSELC	**BCH Encoding/Decoding Bit Select Clear**: It is used to clear	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		BHCR.BSEL to 0.	
3	Reserved	Writing has no effect, read as zero.	R
2	ENCEC	**BCH Encoding/Decoding Select Clear**: It is used to clear BHCR.ENCE to 0.	W
1	INITC	**BCH Controller Initial Clear**： It is used to clear BHCR.INIT to 0.	W
0	BCHEC	**BCH Enable Clear:** It is used to clear BHCR.BCHE to 0.	W

18.2.4 BCH ENC/DEC Count Register (BHCNT)

BHCNT is a 32-bit read/write register that is used to indicate the total number of 8-bit data during encoding or decoding. It is initialized by any reset.

	BHCNT																															**BASE + 0x000C**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved									PARITYSIZE							Reserved					BLOCK SIZE										
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:23	Reserved	Writing has no effect, read as zero.	R
22:16	PARITYSIZE	**Parity size:** The number of parity BYTE generated is *14*ecc_level/8*. **Example:** If BHCR.BSEL = 64, then parity size = 14***** BHCR.BSEL/8 = 112Byte	RW
15:11	Reserved	Writing has no effect, read as zero.	R
10:0	BLOCKSIZE	**Block Size**: Indicates the number of bytes in the data payload for an operation. Supports a wide range of block size (2-1900 bytes), but the size must be a multiple of two bytes. The block size ONLY describes the data size.	RW

18.2.5 BCH Data Register (BHDR)

BHDR is a write-only register that is used to transfer ECC data to BCH.

Data on BHDR is in little-endian format.

	BHDR																															**BASE + 0x0010**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BHDR																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

18.2.6 BCH Parity Register (BHPARn, n=0, 1, 2, 3, 4, 5…26, 27)

BHPARn (n=0, 1, 2, 3, 4… 26, 27) are all 32-bit read/write register that contains the encoding parity

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

data during BCH correction.

Data on BHPARn is in little-endian format with the least-significant-byte (LSB) on bits [15:0] and the most-significant-byte (MSB) on bits [31:16].

Example：
When 64-bit BCH is selected, BHPAR0~BHPAR27 consist of the 896 (14x64）bits of parity data and bit 0 of BHPAR0 is the 896th bit of parity data and bit 0 of BHPAR27 is the 1st bit of parity data.

When 60-bit BCH is selected, BHPAR0~BHPAR26 consist of the 840 bits of parity data and bit 0 of BHPAR0 is the 840th bit of parity data and bit 7 of BHPAR26 is the 1st bit of parity data.

When 8-bit BCH is selected, BHPAR0~BHPAR3 consist of the 112 bits of parity data and bit 0 of BHPAR0 is the 112th bit of parity data and bit 15 of BHPAR3 is the 1st bit of parity data.

	BHPAR0																															**BASE + 0x0014**
	BHPAR1																															**BASE + 0x0018**
	BHPAR2																															**BASE + 0x001C**
	BHPAR3																															**BASE + 0x0020**
	BHPAR4																															**BASE + 0x0024**
	BHPAR5																															**BASE + 0x0028**
	BHPAR6																															**BASE + 0x002C**
	…																															
	BHPAR25																															**BASE + 0x0078**
	BHPAR26																															**BASE + 0x007C**
	BHPAR27																															**BASE + 0x0080**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BHPAR0~BHPAR27																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

18.2.7 BCH Error Report Register (BHERRn, n=0,1,2,3,4,5,6,7,…,61,62,63)

BHERRn is 32-bit read/write register that contains the index for each error after BCH decoding.
The correction index indicates the location of the correction with in the block. The index is a halfword index since the associated correction mask contains a 16-bits mask.
The correction mask provides a 16bit mask that should be applied to the data at the associated index to correct the data.
Any bit errors found in the parity region will NOT be reported as a correction (INDEX), but these bits will be reported in the number of errors status field (BHINT.TERRC).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	BHERR0	BASE + 0x0084
	BHERR1	BASE + 0x0088
	BHERR2	BASE + 0x008C
	BHERR3	BASE + 0x0090
	BHERR4	BASE + 0x0094
	BHERR5	BASE + 0x0098
	BHERR6	BASE + 0x009C
	BHERR7	BASE + 0x00A0
	…	…
	BHERR61	BASE + 0x0178
	BHERR62	BASE + 0x017C
	BHERR63	BASE + 0x0180

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MASKn(n=0,1,2,…62,63)																Reserved					INDEXn(n=0,1,2,…62,63)										
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	MASKn	**Error Byte MASK:** The correction mask provides a bitmask that should be applied to the data at the associated index to correct the data. Example, if n = 1, INDEX1 = 2, MASK1=0x5080, it means the second half word is an error-halfword (three-error-bits, because 0x5080 have three bits 1 in binary) and needs to be XOR MASK1 (0x5080).	R
15:11	Reserved	Writing has no effect, read as zero.	R
10:0	INDEXn	**Error Byte Index:** It is used to indicate the location of the error Byte. Example, INDEXn=0, it means the 1st halfword is an error halfword. INDEXn=1, it means the 2nd halfword is an error halfword. INDEXn=2, it means the 3rd halfword is an error halfword.	R

18.2.8 BCH Interrupt Status Register (BHINT)

BHINT is a 32-bit read-only register that contains the interrupt flag and error count information during BCH correction. It is initialized by any reset. Software writes **1** to clear the corresponding bit except ERRC\BPSO\TERRC.

NOTE.

{DECF,UNCOR,ERR}=3'b100, means no error need be correct;

{DECF,UNCOR,ERR}=3'b11x, means uncorrectable error;

{DECF,UNCOR,ERR}=3'b101, means error can be correct;

	BHINT	BASE + 0x0184

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

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved	ERRC							BPSO	TERRC							Reserved										SDMF	ALL_f	DECF	ENCF	UNCOR	ERR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	Reserved	Writing has no effect, read as zero.	R
30:24	ERRC	**Error Count:** It indicates the number of error **halfword** (16bits) in the data block. Any error found in the parity region will NOT be reported in this counter. NOTE: ERRC is only valid when UNCOR = 0 and ERR = 1. **ERRC** **Description** 0 No errors (Initial value) 1 One error-half-word in the data block 2 Two error-half-words in the data block 3 Three error-half-words in the data block … 64 Sixty-four error-half-words in the data block	RW
23	BPSO	**Bypass output:** Indicates that no corrections were applied to the data block since the BPSO was asserted at the beginning of the correction operation.	RW
22:16	TERRC	**Total Error Count:** It indicates the total number of error **bits** in the data block and parity region. NOTE: TERRC is only valid when UNCOR = 0 and ERR = 1. **TERRC** **Description** 0 No errors (Initial value) 1 One error in the data block 2 Two errors in the data block 3 Three errors 4 Four errors … 64 Sixty-four errors	RW
15:6	Reserved	Writing has no effect, read as zero.	R
5	SDMF	**Decoding SYNDROME Finish:** It indicates that hardware finish Syndrome generating and only in decoding mode this bit will be set. **SDMF** **Description** 0 Syndrome not Finish (Initial value) 1 Syndrome Finish	RW
4	ALL_f	**ALL_f:** It indicates that all data received during decoding are 0xff. When receiving all 0xff data, BCH doesn't correct the data and no error occurs. **ALL_f** **Description** 0 Not all data (data + parity bytes) are 0xff (Initial value)	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1 All data (data + parity bytes) are 0xff	
3	DECF	**Decoding Finish:** It indicates that hardware finish BCH decoding. **DECF Description** 0 Decoding not Finish (Initial value) 1 Decoding Finish	RW
2	ENCF	**Encoding Finish:** It indicates that hardware finish BCH encoding. **ENCF Description** 0 Encoding not Finish (Initial value) 1 Encoding Finish	RW
1	UNCOR	**Un-correction Error:** Indicates that the block could not be corrected because the number of errors present is greater than the ECC capability. **UNCOR Description** 0 No uncorrectable error (Initial value) 1 Uncorrectable error occur	RW
0	ERR	**Error:** It indicates that hardware detects error bits in data in the data block and parity block during BCH decoding. **ERR Description** 0 No error (Initial value) 1 Error occur	RW

18.2.9 BCH Interrupt Enable Set Register (BHINTES)

BHINTES is a 32-bit write-only register that is used to set BHINTE register. Writing 1 to BHINTES will set the corresponding bit in BHINTE to 1. Writing 0 to BHINTES is ignored.

	BHINTES																															BASE + 0x0188
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										SDMFES	ALL_FES	DECFES	ENCFES	UNCORES	ERRES
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5	SDMFES	**SYNDROME Finish Interrupt Enable Set:** It is used to set BHINTE.SDMFE to 1.	W
4	ALL_FES	**ALL_F Interrupt Enable Set:** It is used to set BHINTE.ALL_FE to 1.	W
3	DECFES	**Decoding Finish Interrupt Enable Set:** It is used to set BHINTE.DECFE to 1.	W
2	ENCFES	**Encoding Finish Interrupt Enable Set:** It is used to set BHINTE.ENCFE to 1.	W
1	UNCORES	**Un-correction Error Interrupt Enable Set:** It is used to set	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		BHINTE.ENCFE to 1.	
0	ERRES	**Error Interrupt Enable Set:** It is used to set BHINTE.ERRE to 1.	W

18.2.10 BCH Interrupt Enable Clear Register (BHINTEC)

BHINTEC is a 32-bit write-only register that is used to clear BHINTE register. Writing 1 to BHINTEC will clear the corresponding bit in BHINTE to 0. Writing 0 to BHINTEC is ignored.

	BHINTEC																															BASE + 0x018C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										SDMFEC	ALL_FEC	DECFEC	ENCFEC	UNCORC	ERREC
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:5	Reserved	Writing has no effect, read as zero.	R
5	SDMFEC	**SYNDROME Finish interrupt Enable Clear:** It is used to clear BHINTE.SDMFE to 0.	W
4	ALL_FEC	**ALL_F Interrupt Enable Clear:** It is used to clear BHINTE.ALL_FE to 0.	W
3	DECFEC	**Decoding Finish Interrupt Enable Clear:** It is used to clear BHINTE.DECFE to 0.	W
2	ENCFEC	**Encoding Finish Interrupt Enable Clear:** It is used to clear BHINTE.ENCFE to 0.	W
1	UNCOREC	**Un-correction Error Interrupt Enable Clear:** It is used to clear BHINTE.ENCFE to 0.	W
0	ERREC	**Error Interrupt Enable Clear:** It is used to set BHINTE.ERRE to 0.	W

18.2.11 BCH Interrupt Enable Register (BHINTE)

BHINTE is a 32-bit read/write register that is used to enable/disable interrupts during BCH correction. It is initialized by any reset.

	BHINTE																															BASE + 0x0190
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										SDMFE	ALL_FE	DECFE	ENCFE	UNCORE	ERRE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5	SDMFE	**SYNDROME Finish Interrupt Enable:** It is used to enable or disable syndrome finish interrupt. **SDMFE Description** 0 Disable Syndrome Finish interrupt(Initial value) 1 Enable Syndrome Finish interrupt	RW
4	ALL_FE	**ALL_F Interrupt Enable:** It is used to enable or disable all_f data interrupt. **ALL_FE Description** 0 Disable ALL_F data interrupt (Initial value) 1 Enable ALL_F data interrupt	RW
3	DECFE	**Decoding Finish Interrupt Enable:** It is used to enable or disable decoding finish interrupt. **DECFE Description** 0 Disable Decoding Finish Interrupt (Initial value) 1 Enable Decoding Finish Interrupt	RW
2	ENCFE	**Encoding Finish Interrupt Enable:** It is used to enable or disable encoding finish interrupt. **ENCFE Description** 0 Disable Encoding Finish Interrupt (Initial value) 1 Enable Encoding Finish Interrupt	RW
1	UNCORE	**Un-correction Error Interrupt Enable:** It is used to enable or disable un-correction error interrupt. **UNCORE Description** 0 Disable Uncorrectable Error interrupt (Initial value) 1 Enable Uncorrectable Error Interrupt	RW
0	ERRE	**Error Interrupt Enable:** It is used to enable or disable error interrupt. **ERRE Description** 0 Disable Error interrupt (Initial value) 1 Enable Error interrupt	RW

18.2.12 BCH User TAG OUTPUT Register (BHTO)

BHTO is a 32-bit read only register that is used to indicate the tag for the current corrected data block.

	BHTO																															BASE + 0x0194
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																TAGO															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero. NOTE. Software must **NOT** write 1 to this field.	R
15:0	TAGO	**User-provided Tag OUTPUT:** Indicates the tag for the current corrected data block.	R

18.3 BCH Operation

BCH controller supports a wide range of data payloads (2-1900 bytes), but the payload must be a multiple of two bytes. Only even ECC multiples are supported (i.e. ECC 2, 4, 6, 8 ... 58, 60, 62, 64).
During encoding, hardware will generate 896-bit (14x64) parity data in 64-bit correction, 840-bit parity data in 60-bit correction, 560-bit parity data in 40-bit correction, 336-bit parity data in 24-bit correction, 224-bit parity data in 16-bit correction or 112-bit parity data in 8-bit correction, etc.. Parity data can be read out by CPU or DMA.
During decoding, if there are error bits in data block, after decoding BHERRn registers will hold the error location that can be read by CPU or DMA.

18.3.1 Encoding Sequence

BCH encoding can be operated by CPU or DMA.

18.3.1.1 CPU

1. Set BHCR.BCHE to 1 to enable BCH controller.
2. Select correction mode by setting BHCR.BSEL (64-bit, 60-bit… 8-bit, 4-bit or 2-bit).
3. Set BHCR.ENCE to 1 to enable encoding.
4. Set BHCNT to data block size and parity size in bytes.
5. Set BHCR.INIT to 1 to start a new operation.
6. Writes all data blocks to BHDR.
7. Check BHINTS.ENCF bit or by enabling encoding finish interrupt.
8. When encoding finishes, read out the parity data in BHPAR n.
9. After parity data is read out, clear BHINT.ENCF.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18.3.1.2 DMA

Figure 18-1 Block Diagram for BCH encoding

1 Set BHCR.BCHE to 1 to enable BCH controller.
2 Select correction mode by setting BHCR.BSEL (64-bit, 60-bit… 8-bit, 4-bit or 2-bit).
3 Set BHCR.ENCE to 1 to enable encoding.
4 Set BHCNT to data block size and parity size in bytes.
5 Set BHCR.INIT to 1 to start a new operation.
6 Start DMA transfer after configuring DMA channel.
7 DMA read data block from system memory and write to BCH controller automatically.
8 DMA will wait BCH *encoding* request when finishes writing data block.
9 BCH controller will issue encoding request to DMA when encoding ends.
10 DMA start to read out parity data.
11 After parity data is read out, DMA automatically clear BHINT.ENCF.

NOTE: When DMA is enabled, software should guarantee not to enable encoding finish interrupt.

18.3.2 Decoding Sequence

BCH decoding can be operated by CPU or DMA.

18.3.2.1 CPU

1 Set BHCR.BCHE to 1 to enable BCH controller.
2 Select correction mode by setting BHCR.BSEL (64-bit, 60-bit… 8-bit, 4-bit or 2-bit).
3 Clear BHCR.ENCE to 0 to enable decoding.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 Set BHCNT to data block size and parity size in bytes.
5 Set BHCR.INIT to 1 to start a new operation.
6 Write all data block to BHDR.
7 If another block needs to be decoded, please check BHINT.SDMF or by enabling syndrome finish interrupt. If BHINT.SDMF is asserted and another block needs to be decoded, set this bit to 1 to clear interrupt and go to step 5 to start a new decoding operation.
8 Check BHINTS.DECF bit or by enabling decoding finish interrupt.
9 When decoding finishes, read out the status in BHINT and error report in BHERRn.
10 After status and error report data is read out, set BHINT.DECF to 1.

18.3.2.2 DMA

Figure 18-2 Block Diagram for BCH decoding

1 Set BHCR.BCHE to 1 to enable BCH controller.
2 Select correction mode by setting BHCR.BSEL (64-bit, 60-bit… 8-bit, 4-bit or 2-bit).
3 Clear BHCR.ENCE to 0 to enable decoding.
4 Set BHCNT to data block and parity size in bytes.
5 Set BHCR.INIT to 1 to start a new operation.
6 Start DMA transfer after configuring DMA channel.
7 DMA read data block from system memory and write to BCH controller automatically.
8 DMA will wait BCH *syndrome* or *decoding* request when finishes writing data block. (If *syndrome* request is asserted and another block needs to be decoded, set BHINT. SDMF to 1 to clear request, then go to step 6 to start a new decoding operation.)
9 BCH controller will issue decoding request to DMA when decoding ends.
10 DMA start to read out bch interrupt status (BHINT) and error report data (BHERRn) and

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

write to memory.

11 After status and error report data is read out, set BHINT.DECF to 1 to clear *decoding* request.

Figure 18-3 BCH decoding data flow

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 5

SYSTEM FUNCTIONS

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19 Clock Reset and Power Controller

19.1 Overview

The Clock & Power management block consists of three parts: Clock control, PLL control, and Power control, Reset control.

The Clock control logic can generate the required clock signals including CCLK for CPU, HHCLK for L2CACHE, H0CLK for the AHB0, H2CLK for the AHB2, PCLK for the APB bus peripherals , VPU_CLK for VPU, DDR_CLK for DDR.

The Chip has two Phase Locked Loops (PLL):

1. APLL mainly is used to CPU and L2CACHE, also may use the AHB0 and AHB2, or other peripherals
2. MPLL mainly is used to DDR, also uses AHB0 and AHB2 APB, or other peripherals

It is recommend DDR clock to use MPLL .

For the power control logic, there are various power management schemes to keep optimal power consumption for a given task. The power management block can activate four modes: NORMAL mode, DOZE mode, IDLE mode, SLEEP mode.

Support power supply shut down for 4 power domain separately. Software may separately shut down P0, P1, L2MP, VPU, GPU, H2D, ISP module. When in Sleep mode, software may shut down P0,P1.. Thus, the chip may best reduce leakage current.

For reset control logic, the hardware reset and hibernate reset is extended to more 80ms. It controls or distributes all of the system reset signals.

Feature:

- On-chip 2MHz~27MHZ oscillator circuit
- On-chip 32.768KHZ oscillator circuit
- One two-chip phase-locked loops (PLL) with programmable multiplier
- CCLK, HHCLK, H2CLK, PCLK, H0CLK, DDR_CLK, VPU_CLK frequency can be changed separately for software by setting registers
- MSC clock supports 50M clock
- Functional-unit clock gating
- Shut down power supply for P0, P1, VPU, GPU, IPU, ISP.,BCH,HASH,LCDC,USB and UHC.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.1 CGU Block Diagram

Following figure illustrates a block diagram of CGU.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Clock mux **in grey color** represents **glitch-free** clock mux, which is free of glitches if clock selection is changed. Clock mux in white color represents non-glitch-free clock mux, which can suffer from glitches when changing clock sources. **Care must be taken in using each of clock muxes**. For glitch-free mux, it should be guaranteed that both of clock sources are running when clock selection is changed from one to the other. If that's not the case, clock changing is not finished fully and resulting clock output can have unknown states. For non-glitch-free clock mux, it is possible to have a glitch when clock selections are changed. To prevent the glitch signals, it is recommended to disable output of non-glitch-free muxes before trying to change clock sources. After clock changing is completed, users can re-enable output of the non-glitch-free clock mux so that there will be no glitches resulting from clock changes. Masking output of non-glitch-free muxes are handled by clock source control registers.

the detail clk structure of VPU/LCD_PIX clk and so on is as follows:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

clk_gate = clkgr & sleep & vpu_pd, and you can see how the the VPUCDR.VPU_STOP and CLKGR can stop vpu's clk. when sleep, MPLL and sclk are stopped, so clkgr or vpu_stop is not so important. Note: chip top's output PCLK of TFT panel is from output of lcd_div, not from gate cell of lcd.

19.1.2 CGU Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 19-1 Registers Memory Map-Address Base

Name	Base	Description
CPM	0x10000000	Address base of CPM

Table 19-2 CGU Registers Configuration

Name	description	RW	Reset Value	Address offset	Access Size
CPCCR	Clock Control Register	RW	0x95000000	0x0000	32
CPCSR	Clock status register	RW	0x00000000	0x00D4	32
DDCDR	DDR clock divider Register	RW	0x40000000	0x002C	32
VPUCDR	VPU clock divider Register	RW	0x00000000	0x0030	32
I2SCDR	I2S device clock divider Register	RW	0x00000000	0x0060	32
USBCDR	OTG PHY clock divider Register	RW	0x00000000	0x0050	32
LPCDR	LCD pix clock divider Register	RW	0x00000000	0x0064	32
MSC0CDR	MSC0 clock divider Register	RW	0x40000000	0x0068	32
MSC1CDR	MSC1 clock divider Register	RW	0x00000000	0x00A4	32
MSC2CDR	MSC2 clock divider Register	RW	0x00000000	0x00A8	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

UHCCDR	UHC 48M clock divider Register	RW	0x00000000	0x006C	32
SSICDR	SSI clock divider Register	RW	0x00000000	0x0074	32
CIMCDR	CIM MCLK clock divider Register	RW	0x00000000	0x007C	32
PCMCDR	PCM device clock divider Register	RW	0x00000000	0x0084	32
GPUCDR	GPU clock divider Register	RW	0x40000000	0x0088	32
ISPCDR	HDMI clock divider Register	RW	0x00000000	0x0080	32
BCHCDR	BCH clock divider Register	RW	0x40000000	0x00AC	32
CPM_INTR	CPM interrupt Register	RW	0x00000000	0x00B0	32
CPM_INTRE	CPM interrupt Enable Register	RW	0x00000000	0x00B4	32
DRCG	DDR clk gate register	RW	0x00000001	0x00D0	32
CPSPR	CPM Scratch Pad Register	RW	0x????????	0x0034	32
CPSPPR	CPM Scratch Protected Register	RW	0x0000a5a5	0x0038	32
USBPCR	USB Parameter control register	RW	0x429919b8	0x003C	32
USBRDT	USB Reset Detect Timer Register	RW	0x02000096	0x0040	32
USBVBFIL	USB jitter filter Register	RW	0x00ff0080	0x0044	32
USBPCR1	USB Parameter control register 1	RW	0x00000004	0x0048	32
CPPCR	PLL Control Register	RW	0x80000020	0x000C	32
CPAPCR	APLL Control Register	RW	0x00000000	0x0010	32
CPMPCR	MPLL Control Register	RW	0x00000000	0x0014	32
CPAPACR	APLL Frac Register	RW	0x00000000	0x0018	32
CPMPACR	MPLL Frac Register	RW	0x00000000	0x001C	32

19.1.2.1 Clock Control Register

The Clock Control Register (CPCCR) is a 32-bit read/write register, which controls CCLK, H0CLK, H2CLK and PCLK division ratios. It is initialized to 0x95000000 by any reset. Only word access can be used on CPCCR.

	CPCCR																															BASE+0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SEL_SRC		SEL_CPLL		SEL_H0PLL		SEL_H2PLL		GATE_SCLKA	CE_CPU	CE_AHB0	CE_AHB2	PDIV				H2DIV				H0DIV				L2CDIV				CDIV			
RST	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	SEL_SRC	00: stop MUX clock output 01: APLL 10: EXCLK The mux clock output is SCLK_A	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		The clock mux is free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to both are running.	
29:28	SEL_CPLL	00: stop MUX clock output 01: SCLK_A 10: MPLL The mux clock output is to CPU and L2CAHE The clock mux is free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to both are running.	RW
27:26	SEL_H0PLL	00: stop MUX clock output 01: SCLK_A 10: MPLL The mux clock output is to AHB0 The clock mux is free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to both are running.	RW
25:24	SEL_H2PLL	00 : stop MUX clock output 01: SCLK_A 10: MPLL The mux clock output is to AHB2 The clock mux is free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to both are running.	RW
23	GATE_SCLKA	0: SCLK_A not gating 1: SCLK_A is gating. Gating module: LCD GPU VPU ISP MSC BCH AIC USB UHC SSI PCM CIM_MCLK etc	RW
22	CE_CPU	Change enable for CPU and L2CAHE. If CE_CPU is 1 , write on CDIV, L2CDIV will start a frequency changing sequence immediately. If CE_CPU is 0, writes on CDIV and L2CDIV have no affect.	RW
21	CE_AHB0	Change enable for AHB0. If CE_AHB0 is 1 , write on H0DIV will start a frequency changing sequence immediately. If CE_AHB0 is 0, writes on H0DIV have no affect.	RW
20	CE_AHB2	Change enable for AHB2. If CE_AHB2 is 1 , write on H2DIV, PDIV will start a frequency changing sequence immediately. If CE_AHB2 is 0, writes on H2DIV, PDIV have no affect.	RW
19:16	PDIV	Divider for Peripheral Frequency. Specified the PCLK division ratio.	RW

Bit 19~16: PDIV				Description
0	0	0	0	X1
0	0	0	1	X1/2
0	0	1	0	X1/3

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			0	0	1	1	X1/4	
			0	1	0	0	X1/5	
			0	1	0	1	X1/6	
			0	1	1	0	X1/7	
			0	1	1	1	X1/8	
			1	0	0	0	X1/9	
			1	0	0	1	X1/10	
			1	0	1	0	X1/11	
			1	0	1	1	X1/12	
			1	1	0	0	X1/13	
			1	1	0	1	X1/14	
			1	**1**	**1**	**0**	**Clr stop CLK**	
			1	**1**	**1**	**1**	**Set Stop CLK**	
		When software sets PDIV == 1111 and CE is 1, PCLK will stop. PDIV value will not be changed actually, it is only set register. When software sets PDIV == 1110 and CE is 1, PCLK will continue to run. PDIV value will not be changed actually, it is only clear register.						
15:12	H2DIV	Divider for AHB2 Frequency. Specified the AHB2 CLK division ratio.						RW
			Bit 15~12: H2DIV				Description	
			0	0	0	0	X1	
			0	0	0	1	X1/2	
			0	0	1	0	X1/3	
			0	0	1	1	X1/4	
			0	1	0	0	X1/5	
			0	1	0	1	X1/6	
			0	1	1	0	X1/7	
			0	1	1	1	X1/8	
			1	0	0	0	X1/9	
			1	0	0	1	X1/10	
			1	0	1	0	X1/11	
			1	0	1	1	X1/12	
			1	1	0	0	X1/13	
			1	1	0	1	X1/14	
			1	**1**	**1**	**0**	**Clr stop CLK**	
			1	**1**	**1**	**1**	**Set Stop CLK**	
		When software sets H2DIV == 1111 and CE is 1, H2CLK will stop. H2DIV value will not be changed actually, it is only set						

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>register.
When software sets H2DIV == 1110 and CE is 1, H2CLK will continue to run. H2DIV value will not be changed actually, it is only clear register.</td><td></td></tr>
<tr><td>11:8</td><td>H0DIV</td><td>Divider for AHB0 Frequency. Specified the AHB0 CLK division ratio.
<table>
<tr><th colspan="4">Bit 11~8: H0DIV</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>X1</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>X1/2</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>X1/3</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>X1/4</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>X1/5</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>X1/6</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>X1/7</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>X1/8</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>X1/9</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>X1/10</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>X1/11</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>X1/12</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>X1/13</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>X1/14</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>Clr stop CLK</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>Set Stop CLK</td></tr>
</table>
When software sets H0DIV == 1111 and CE is 1, H0CLK will stop. H0DIV value will not be changed actually, it is only set register.
When software sets H0DIV == 1110 and CE is 1, H0CLK will continue to run. H0DIV value will not be changed actually, it is only clear register.</td><td>RW</td></tr>
<tr><td>7:4</td><td>L2CDIV</td><td>Divider for L2CACHE Frequency. Specified the L2C CLK division ratio.
<table>
<tr><th colspan="4">Bit 7~4: L2CDIV</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>X1</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>X1/2</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>X1/3</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>X1/4</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>X1/5</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>X1/6</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>X1/7</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>X1/8</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>X1/9</td></tr>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	0	0	1	X1/10
1	0	1	0	X1/11
1	0	1	1	X1/12
1	1	0	0	X1/13
1	1	0	1	X1/14
1	**1**	**1**	**0**	**Clr stop CLK**
1	**1**	**1**	**1**	**Set Stop CLK**

When software sets L2CDIV == 1111 and CE is 1, L2CCLK will stop. L2CDIV value will not be changed actually, it is only set register.
When software sets L2CDIV == 1110 and CE is 1, L2CCLK will continue to run. L2CDIV value will not be changed actually, it is only clear register.

3:0 CDIV

Divider for CPU Frequency. Specified the CPU CLK division ratio.

Bit 3~0: CDIV				Description
0	0	0	0	X1
0	0	0	1	X1/2
0	0	1	0	X1/3
0	0	1	1	X1/4
0	1	0	0	X1/5
0	1	0	1	X1/6
0	1	1	0	X1/7
0	1	1	1	X1/8
1	0	0	0	X1/9
1	0	0	1	X1/10
1	0	1	0	X1/11
1	0	1	1	X1/12
1	1	0	0	X1/13
1	1	0	1	X1/14
1	**1**	**1**	**0**	**Clr stop CLK**
1	**1**	**1**	**1**	**Set Stop CLK**

When software sets CDIV == 1111 and CE is 1, CCLK will stop. CDIV value will not be changed actually, it is only set register.
When software sets CDIV == 1110 and CE is 1, CCLK will continue to run. CDIV value will not be changed actually, it is only clear register.

Import Note1 : for PDIV and AHB2DIV, AHB2's clk frequency must be 1 or 2 times of PDIV's

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

clk frequency. For L2CDIV and CDIV, CPU's clk frequency must be 1,2,3, or 4 times of L2CACHE's clk frequency.

Import Note2 : After reset, H0DIV and CE_AHB0 will be zero,and ahb0's clk freq and apb0's clk freq will be 1:1, but if you have ever set CE_AHB0 and change the value of H0DIV, ahb0's clk freq and apb0's clk freq will be 2:1 from then on.

19.1.2.2 Clock Status Register

Clock status register (CPCSR) is a 32-bit read/write register that specifies the status of CPCCR .This register is initialized to 0x00000000 only by any reset. Only word access can be used on CPCSR.

	CPCSR																															BASE+0x00D4		
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																													H2DIV_BUSY	H0DIV_BUSY	CDIV_BUSY
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:3	Reserved	Writing has no effect, read as zero.	R
2	H2DIV_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until H2DIV_BUSY == 0, then may begin another frequency change.	R
1	H0DIV_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until H0DIV_BUSY == 0, then may begin another frequency change.	R
0	CDIV_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until CDIV_BUSY == 0, then may begin another frequency change.	R

19.1.2.3 DDR Memory clock divider Register

DDR memory clock divider Register (DDRCDR) is a 32-bit read/write register that specifies the divider of DDR memory clock . This register is initialized to 0x40000000 only by any reset. Only word

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

access can be used on DDRCDR.

	DDRCDR																															BASE+0x002C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCS		CE_DDR	DDR_BUSY	DDR_STOP	GATE_EN	CHANGE_EN	FLAG	Reserved																				DDRCDR			
RST	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	DCS	DDR Clock Source Selection. Selects the DDR clock source between APLL output and MPLL output. 00: stop MUX clock output 01: DDR clock source is SCLK_A 10: DDR clock source is MPLL The clock mux is free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to both are running.	RW
29	CE_DDR	Change enable for DDR. If CE_DDR is 1 , write on DDRCDR will start a frequency changing sequence immediately. If CE_DDR is 0, writes on DDRCDR have no affect	RW
28	DDR_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until DDR_BUSY == 0, then may begin another frequency change.	RW
27	DDR_STOP	When DDR_STOP is 1 and CE_DDR is 1, the DDR clock will stop. When DDR_STOP is 0 and CE_DDR is 1, the DDR clock will continue. Software should wait until DDR_BUSY == 0, then may begin another frequency change. DDR_STOP is prior to DDRCDR.	RW
26	GATE_EN	1'b1: if ddr gives a clk stop req, cpm will gate ddr's clk.	
25	CHANGE_EN	When this bit set high ,cpm will change ddr's clk when when CE_DDR is high and FLAG is low	RW
24	FLAG	Will be set high when ddr gives relative flag Can only be write zero by software	RW 0
23:4	Reserved	Writing has no effect, read as zero.	R
3:0	DDRCDR	Divider for DDR Frequency. Specified the DDR memory clock division ratio, which varies from 1 to 16 (division ratio = DDRCDR + 1).	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.2.4 VPU clock divider Register

VPU clock divider Register (VPUCDR) is a 32-bit read/write register that specifies the divider of VPU clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on VPUCDR.

	VPUCDR																															BASE+0x 0030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCS	Reserved	CE_VPU	VPU_BUSY	VPU_STOP	Reserved																							VPUCDR			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	VCS	VPU Clock Source Selection. Selects the VPU clock source 0: VPU clock source is SCLK_A 1: VPU clock source is MPLL It is no glitch free mux. Software should be stop VPU clock, when change this bit.	RW
30	Reserved		
29	CE_VPU	Change enable for VPU. If CE_VPU is 1 , write on VPUCDR will start a frequency changing sequence immediately. If CE_VPU is 0, writes on VPUCDR have no affect	RW
28	VPU_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until VPU_BUSY == 0, may begin another frequency change.	RW
27	VPU_STOP	When VPU_STOP is 1 and CE_VPU is 1, the VPU clock will stop. When VPU_STOP is 0 and CE_VPU is 1, the VPU clock will continue. Software should wait until VPU_BUSY == 0, may begin another frequency change. VPU_STOP is prior to VPUCDR. If system don't use this module, may set this bit to reduce power .	RW
26:4	Reserved	Writing has no effect, read as zero.	R
3:0	VPUCDR	Divider for VPU Frequency. Specified the VPU clock division ratio, which varies from 1 to 16 (division ratio = VPUCDR + 1).	RW

19.1.2.5 USB clock divider Register

USB clock divider Register (USBCDR) is a 32-bit read/write register that specifies the divider of USBPHY clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on USBCDR.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	USBCDR																															BASE+0x 0050
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UCS	UPCS	CE_USB	USB_BUSY	USB_STOP	USB_EN	MIPI_CS	Reserved																	USBCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	UCS	USB Clock Source Selection. Selects the USB clock source between PLL output and pin EXCLK. 0: USB clock source is EXCLK 1: USB clock source is PLL output divided by USBCDR If UCS == 0, softwar should set USB_STOP bit to reduce power. It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
30	UPCS	0: USB clock source is SCLK_A output 1: USB clock source is MPLL output	
29	CE_USB	Change enable for USB. If CE_USB is 1 , write on USBCDR ill start a frequency changing sequence immediately. If CE_USB is 0, writes on USBCDR have no affect	RW
28	USB_BUSY	The bit is read only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should wait until USB_BUSY == 0, may begin another frequency change.	RW
27	USB_STOP	When USB_STOP is 1 and CE_USB is 1, the USB clock will stop. When USB_STOP is 0 and CE_USB is 1, the USB clock will continue. Software should wait until USB_BUSY == 0, may begin another frequency change. USB_STOP is prior to USBCDR. and USB_STOP will just stop clk from UPCS, and if UCS is high, usb phy's CLKCORE will be stopped. If system don't use this module, may set this bit to reduce power .	RW
26	USB_EN	0: USB PHY clock is free runing 1: USB PHY clock is shut off	RW
25	MPI_CS	1'b0:MIPI use exclk; 1'b1:MIPI use usb's clk	
25:8	Reserved	Writing has no effect, read as zero.	R
7:0	USBCDR	Divider for USB Frequency. Specified the USB device clock division ratio division ratio = USBCDR + 1	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.2.6 I2S device clock divider Register//aic

I2S device clock divider Register (I2SCDR) is a 32-bit read/write register that specifies the divider of I2S device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on I2SCDR.

	I2SCDR																															BASE+0x 0060
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	I2PCS		CE_I2S	I2S_BUSY	I2S_STOP	Reserved																			I2SCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	I2PCS	0: SCLK_A output 1: MPLL output 2: EXCLK	RW
29	CE_I2S	Change enable for I2S. If CE_I2S is 1 , write on I2SCDR ill start a frequency changing sequence immediately. If CE_I2S is 0, writes on I2SCDR have no affect	RW
28	I2S_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until I2S_BUSY == 0, may begin another frequency change.	R
27	I2S_STOP	When I2S_STOP is 1 and CE_I2S is 1, the I2S clock will stop. When I2S_STOP is 0 and CE_I2S is 1, the I2S clock will continue. Software should wait until I2S_BUSY == 0, may begin another frequency change. I2S_STOP is prior to I2SCDR. If system don't use this module, may set this bit to reduce power .	RW
26:8	Reserved	Writing has no effect, read as zero.	
7:0	I2SCDR	Divider for I2S Frequency. Specified the I2S device clock division ratio, which varies from 1 to 256 (division ratio = I2SCDR + 1).	RW

19.1.2.7 LCD pix clock divider Register

LCD pix clock divider Register (LPCDR) is a 32-bit read/write register that specifies the divider of LCD pixel clock (LPCLK). This register is initialized to 0x04000000 by any reset. Only word access can be used on LPCDR.

	LPCDR																															BASE+0x 0064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LPCS	Reserved		CE_LCD	LCD_BUSY	LCD_STOP	Reserved																			LPCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	LPCS	0: SCLK_A output 1: MPLL output It is no glitch free mux. Software should be stop PIX clock, when change this bit.	RW
30:29	Reserved	Writing has no effect, read as zero.	RW
28	CE_LCD	Change enable for LCD. If CE_LCD is 1 , write on LPCDR will start a frequency changing sequence immediately. If CE_LCD is 0, writes on LPCDR have no affect	RW
27	LCD_BUSY	The bit is read only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until LCD_BUSY == 0, may begin another frequency change.	RW
26	LCD_STOP	When LCD_STOP is 1 and CE_LCD is 1, the LCD clock will stop. When LCD_STOP is 0 and CE_LCD is 1, the LCD clock will continue. Software should wait until LCD_BUSY == 0, may begin another frequency change. LCD_STOP is prior to LPCDR. If system don't use this module, may set this bit to reduce power .	RW
25:8	Reserved	Writing has no effect, read as zero.	RW
7:0	LPCDR	Divider for Pixel Frequency. Specified the LCD pixel clock (LPCLK) division ratio, which varies from 1 to 256 (division ratio = LPCDR + 1).	RW

19.1.2.8 MSC0 device clock divider Register

MSC0 device clock divider Register (MSC0CDR) is a 32-bit read/write register that specifies the divider of MSC0 device clock . This register is initialized to 0x40000000 only by any reset. Only word access can be used on MSC0CDR.

	MSC0CDR																																**BASE+0x 0068**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPCS	Reserved	CE_MSC0	MSC0_BUSY	MSC0_STOP	Reserved											S_CLK0_SEL	Reserved							MSC0CDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	MPCS	0: select SCLK_A clock output 1: select MPLL clock output The clock mux is not free of glitches if clock selection is changed The clock mux output is to MSC0, MSC1, MSC2 . It is source clock of MSC0, MSC1, MSC2.	RW
30	Reserved		
29	CE_MSC0	Change enable for MSC0. If CE_MSC0 is 1 , write on MSC0CDR ill start a frequency changing sequence immediately. If CE_MSC0 is 0, writes on MSC0CDR have no affect	RW
28	MSC0_BUSY	The bit is read only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until MSC0_BUSY == 0, may begin another frequency change.	RW
27	MSC0_STOP	When MSC0_STOP is 1 and CE_MSC0 is 1, the MSC0 clock will stop. When MSC0_STOP is 0 and CE_MSC0 is 1, the MSC0 clock will continue. Software should wait until MSC0_BUSY == 0, may begin another frequency change.MSC0_STOP is prior to MSC0CDR. If system don't use this module, may set this bit to reduce power .	RW
26:16	Reserved	Writing has no effect, read as zero.	R
15	S_CLK0_SEL	MSC sample clock selection: 0: Sample clock is 90-degree phase shifted by device clock 1: Sample clock is 180-degree phase shifted by device clock	RW
14:8	Reserved		
7:0	MSC0CDR	Divider for MSC0 Frequency. Specified the MSC0 device clock division ratio division ratio = (MSC0CDR + 1)*2.	RW

19.1.2.9 MSC1 device clock divider Register

MSC1 device clock divider Register (MSC1CDR) is a 32-bit read/write register that specifies the divider of MSC1 device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on MSC1CDR.

	MSC1CDR																															BASE+0x 00A4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		CE_MSC1	MSC1_BUSY	MSC1_STOP	Reserved											S_CLK1_SEL	Reserved							MSC1CDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:30	Reserved		R
29	CE_MSC1	Change enable for MSC1. If CE_MSC1 is 1 , write on MSC1CDR ill start a frequency changing sequence immediately. If CE_MSC1 is 0, writes on MSC1CDR have no affect	RW
28	MSC1_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until MSC1_BUSY == 0, may go next	RW
27	MSC1_STOP	When MSC1_STOP is 1 and CE_MSC1 is 1, the MSC1 clock will stop. When MSC1_STOP is 0 and CE_MSC1 is 1, the MSC1 clock will continue. Software should wait until MSC1_BUSY == 0, may begin another frequency change.MSC1_STOP is prior to MSC1CDR. If system don't use this module, may set this bit to reduce power .	RW
26:16	Reserved	Writing has no effect, read as zero.	R
15	S_CLK1_SEL	MSC1 sample clock selection: 0: Sample clock is 90-degree phase shifted by device clock 1: Sample clock is 180-degree phase shifted by device clock	RW
14:8	Reserved		
7:0	MSC1CDR	Divider for MSC1 Frequency. Specified the MSC1 device clock division ratio division ratio = (MSC1CDR + 1)*2.	RW

19.1.2.10 MSC2 device clock divider Register

MSC2 device clock divider Register (MSC2CDR) is a 32-bit read/write register that specifies the divider of MSC2 device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on MSC2CDR.

	MSC2CDR																															0x100000A8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		CE_MSC2	MSC2_BUSY	MSC2_STOP	Reserved											S_CLK2_SEL	Reserved							MSC2CDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	Reserved		R
29	CE_MSC2	Change enable for MSC2. If CE_MSC2 is 1 , write on MSC2CDR ill start a frequency changing sequence immediately. If CE_MSC2 is 0, writes on MSC2CDR have no affect	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28	MSC2_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until MSC2_BUSY == 0, may begin another frequency change.	RW
27	MSC2_STOP	When MSC2_STOP is 1 and CE_MSC2 is 1, the MSC2 clock will stop. When MSC2_STOP is 0 and CE_MSC2 is 1, the MSC2 clock will continue. Software should wait until MSC2_BUSY == 0, may begin another frequency change.MSC2_STOP is prior to MSC2CDR. If system don't use this module, may set this bit to reduce power .	RW
26:16	Reserved	Writing has no effect, read as zero.	R
15	S_CLK2_SEL	MSC2 sample clock selection: 0: Sample clock is 90-degree phase shifted by device clock 1: Sample clock is 180-degree phase shifted by device clock	RW
14:8	Reserved		
7:0	MSC2CDR	Divider for MSC2 Frequency. Specified the MSC2 device clock division ratio division ratio = (MSC2CDR + 1)*2.	RW

19.1.2.11 UHC device clock divider Register

UHC device clock divider Register (UHCCDR) is a 32-bit read/write register that specifies the divider of UHC 48M device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on UHCCDR.

	UHCCDR																															BASE+0x 006C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UHCS		CE_UHC	UHC_BUSY	UHC_STOP	Reserved																			UHCCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	UHCS	00: UHC clock source is SCLK_A output 01: UHC clock source is MPLL output 10: UHC clock source is USB_PHY It is no glitch free mux. Software should be stop the module clock, when change this bit **When using USB_PHY, the UHCCDR should be = 0;**	RW
29	CE_UHC	Change enable for UHC. If CE_UHC is 1 , write on UHCCDR ill start a frequency changing sequence immediately. If CE_UHC is 0, writes on	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		UHCCDR have no affect	
28	UHC_BUSY	The bit is read only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should wait until UHC_BUSY == 0, may begin another frequency change.	RW
27	UHC_STOP	When UHC_STOP is 1 and CE_UHC is 1, the UHC clock will stop. When UHC_STOP is 0 and CE_UHC is 1, the UHC clock will continue. Software should wait until UHC_BUSY == 0, may begin another frequency change. UHC_STOP is prior to UHCCDR. If system don't use this module, may set this bit to reduce power .	RW
26:8	Reserved	Writing has no effect, read as zero.	R
7:0	UHCCDR	Divider for UHC Frequency. Specified the UHC device clock division ratio division ratio = UHCCDR + 1	RW

19.1.2.12 SSI device clock divider Register

SSI device clock divider Register (SSICDR) is a 32-bit read/write register that specifies the divider of SSI device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on SSICDR.

	SSICDR																															BASE+0x 0074
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPCS		CE_SSI	SSI_BUSY	SSI_STOP	Reserved																			SSICDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	SPCS	00: select SCLK_A output 01: select MPLL output 10: EXCLK It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
29	CE_SSI	Change enable for SSI. If CE_SSI is 1 , write on SSICDR will start a frequency changing sequence immediately. If CE_SSI is 0, writes on SSICDR have no affect	RW
28	SSI_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until SSI_BUSY == 0, may begin another	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		frequency change.	
27	SSI_STOP	When SSI_STOP is 1 and CE_SSI is 1, the SSI clock will stop. When SSI_STOP is 0 and CE_SSI is 1, the SSI clock will continue. Software should wait until SSI_BUSY == 0, may begin another frequency change. SSI_STOP is prior to SSICDR. If system don't use this module, may set this bit to reduce power .	RW
26:8	Reserved	Writing has no effect, read as zero.	R
7:0	SSICDR	Divider for SSI Frequency. Specified the SSI device clock division ratio, which varies from 1 to 256 (division ratio = SSICDR + 1).	RW

19.1.2.13 CIM MCLK clock divider Register

CIM mclk clock divider Register (CIMCDR) is a 32-bit read/write register that specifies the divider of CIM mclk clock (CIM_MCLK). This register is initialized to 0x00000000 only by any reset. Only word access can be used on CIMCDR.

	CIMCDR																															BASE+0x 007C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CIMPCS	CE_CIM	CIM_BUSY	CIM_STOP	Reserved																				CIMCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

Bits	Name	Description	RW
31	CIMPCS	0: select SCLK_A clock output 1: select MPLL clock output It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
30	CE_CIM	Change enable for CIM. If CE_CIM is 1 , write on CIMCDR ill start a frequency changing sequence immediately. If CE_CIM is 0, writes on CIMCDR have no affect	RW
29	CIM_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until CIM_BUSY == 0, may begin another frequency change.	RW
28	CIM_STOP	When CIM_STOP is 1 and CE_CIM is 1, the CIM clock will stop. When CIM_STOP is 0 and CE_CIM is 1, the CIM clock will continue.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Software should wait until CIM_BUSY == 0, may begin another frequency change. CIM_STOP is prior to CIMCDR. If system don't use this module, may set this bit to reduce power .	
27:8	Reserved	Writing has no effect, read as zero.	R
7:0	CIMCDR	Divider for CIM MCLK Frequency. Specified the CIM MCLK clock (CIM_MCLK) division ratio, which varies from 1 to 256 (division ratio = CIMCDR + 1). the output clk is io_setup.cim_mclk_o	RW

19.1.2.14 PCM device clock divider Register

PCM device clock divider Register (PCMCDR) is a 32-bit read/write register that specifies the divider of PCM device clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on PCMCDR.

	PCMCDR																															BASE+0x 0084
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCMCPCS		Reserved	CE_PCM	PCM_BUSY	PCM_STOP	Reserved																		PCMCDR							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	PCMPCS	00: select SCLK_A output 01: select MPLL output 10: select EXCLK output It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
29	Reserved		RW
28	CE_PCM	Change enable for PCM. If CE_PCM is 1 , write on PCMCDR ill start a frequency changing sequence immediately. If CE_PCM is 0, writes on PCMCDR have no affect	RW
27	PCM_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until PCM_BUSY == 0, may go next	RW
26	PCM_STOP	When PCM_STOP is 1 and CE_PCM is 1, the PCM clock will stop. When PCM_STOP is 0 and CE_PCM is 1, the PCM clock will continue. Software should wait until PCM_BUSY == 0, may go next. PCM_STOP is prior to PCMCDR. If system don't use this module, may set this bit to reduce power .	RW
25:8	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7:0	PCMCDR	Divider for PCM Frequency. Specified the PCM device clock division ratio, which varies from 1 to 256 (division ratio = PCMCDR + 1).	RW

19.1.2.15 GPU clock divider Register

GPU clock divider Register (GPUCDR) is a 32-bit read/write register that specifies the divider of GPU clock . This register is initialized to 0x40000000 only by any reset. Only word access can be used on GPUCDR.

	GPUCDR																															BASE+0x 0088
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPCS	Reserved	CE_GPU	GPU_BUSY	GPU_STOP	Reserved																							GPUCDR			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	GPCS	0: select SCLK_A output 1: select MPLL output The clock mux is not free of glitches if clock selection is changed	RW
30	Reserved		
29	CE_GPU	Change enable for GPU. If CE_GPU is 1 , write on GPUCDR ill start a frequency changing sequence immediately. If CE_GPU is 0, writes on GPUCDR have no affect	RW
28	GPU_BUSY	The bit is read only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until GPU_BUSY == 0, may begin another frequency change.	RW
27	GPU_STOP	When GPU_STOP is 1 and CE_GPU is 1, the GPU clock will stop. When GPU_STOP is 0 and CE_GPU is 1, the GPU clock will continue. Software should wait until GPU_BUSY == 0, may begin another frequency change. GPU_STOP is prior to GPUCDR. If system don't use this module, may set this bit to reduce power .	RW
26:4	Reserved	Writing has no effect, read as zero.	R
3:0	GPUCDR	Divider for GPU Frequency. Specified the GPU clock division ratio, which varies from 1 to 16 (division ratio = GPUCDR + 1).	RW

19.1.2.16 ISP clock divider Register

ISP clock divider Register (ISPCDR) is a 32-bit read/write register that specifies the divider of ISP

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

clock . This register is initialized to 0x00000000 only by any reset. Only word access can be used on ISPCDR.

	ISPCDR																															BASE+0x 0080
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ISPCS	Reserved	CE_ISP	ISP_BUSY	ISP_STOP	Reserved																							ISPCDR			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	ISPCS	0: select SCLK_A output 1: select MPLL output It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
29	CE_ISP	Change enable for ISP. If CE_ISP is 1 , write on ISPCDR ill start a frequency changing sequence immediately. If CE_ISP is 0, writes on ISPCDR have no affect	RW
28	ISP_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until ISP_BUSY == 0, may begin another frequency change.	RW
27	ISP_STOP	When ISP_STOP is 1 and CE_ISP is 1, the ISP clock will stop. When ISP_STOP is 0 and CE_ISP is 1, the ISP clock will continue. Software should wait until ISP_BUSY == 0, may begin another frequency change.ISP_STOP is prior to ISPCDR. If system don't use this module, may set this bit to reduce power .	RW
25:4	Reserved	Writing has no effect, read as zero.	R
3:0	ISPCDR	Divider for ISP Frequency. Specified the ISP clock division ratio, which varies from 1 to 16 (division ratio = ISPCDR + 1).	RW

19.1.2.17 BCH clock divider Register

BCH clock divider Register (BCHCDR) is a 32-bit read/write register that specifies the divider of BCH clock . This register is initialized to 0x40000000 only by any reset. Only word access can be used on BCHCDR.

	BCHCDR																															BASE+0x 00AC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	BPCS	Reserved	CE_BCH	BCH_BUSY	BCH_STOP	Reserved																							BCHCDR			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:30	BPCS	0: select SCLK_A output 1: select MPLL output The clock mux isn't free of glitches if clock selection is changed When switching clock source, it should be ensured that the clock switched from and the clock switched to are running.	RW
29	CE_BCH	Change enable for BCH. If CE_BCH is 1 , write on BCHCDR ill start a frequency changing sequence immediately. If CE_BCH is 0, writes on BCHCDR have no affect	RW
28	BCH_BUSY	The bit is ready only bit. It indicates whether the frequency change has finished. When the bit is 0, it indicates frequency change has finished., otherwise it indicates the frequency change is on going Software should be wait until BCH_BUSY == 0, may begin another frequency change.	RW
27	BCH_STOP	When BCH_STOP is 1 and CE_BCH is 1, the BCH clock will stop. When BCH_STOP is 0 and CE_BCH is 1, the BCH clock will continue. Software should wait until BCH_BUSY == 0, may begin another frequency change.BCH_STOP is prior to BCHCDR. If system don't use this module, may set this bit to reduce power .	RW
26:4	Reserved	Writing has no effect, read as zero.	R
3:0	BCHCDR	Divider for BCH Frequency. Specified the BCH clock division ratio, which varies from 1 to 16 (division ratio = BCHCDR + 1).	RW

19.1.2.18 CPM interrupt Register

	CPM_INTR																														BASE+0x 00B0	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														VBUS_INTR	ADEV_INTR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:3	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	VBUS_INTR	B device insert interrupt.	R
0	ADEV_INTR	A device insert interrupt.	R

19.1.2.19 CPM interrupt enable Register

	CPM_INTRE																															BASE+0x 00B4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														VBUS_INTRE	ADEV_INTRE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:3	Reserved	Writing has no effect, read as zero.	R
1	VBUS_INTRE	B device insert interrupt enable.	RW
0	ADEV_INTRE	A device insert interrupt enable.	RW

19.1.2.20 DDR Clk Gate Register

	DRCG																															BASE+0x 00D0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																							DRCG								
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Note: Please refer to DDR and NFI's spec for detail use of this register.

19.1.2.21 CPM Scratch Pad Protected Register

The Scratch Pad Protected Register is reset to 0x0000a5a5. When CPSPPR value equals to 0x00005a5a, software can write the CPSPR. or else can't.

	CPSPPR																															BASE+0x 0038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Reserved																CPSPPR															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:0	CPSPPR	The value is only = 0x00005a5a, software can write the CPSPR.	RW

19.1.2.22 CPM Scratch Pad Register

The Scratch Pad Register is a 32-bit read/write register that allows software to preserve some critical data . It is not initialized by power on and WDT reset.

	CPPSR																															BASE+0x 0034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPPSR																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

19.1.2.23 USB Parameter Control Register

The USBPCR is a 32-bit read/write register that allows software to control USB PHY some functions. It is initialized to 0x409919b8.

	USBPCR																															BASE+0x 003C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USB_MODE	AVLD_REG	IDPULLUP_MASK		INCR_MASK	TXRISETUNE	COMONONN	VBUSVLDEXT	VBUSVLDEXTSEL	POR	SIDDQ	OTGDISABLE	COMPDISTUNE				OTGTUNE			SQRXTUNE		TXFSLSTUNE				TXPREMTUNE	TXHSXVTUNE		TXVREFTUNE			
RST	0	1	0	0	0	0	0	0	1	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	1	0	1	1	1	0	0	0

Bits	Name	Description	RW
31	USB_MODE	0: work as USB device 1: work as OTG	RW
30	AVLD_REG	This bit is used to set “avalid”(VBUS above A-device session threshold) signal.	RW
29:28	IDPULLUP_MASK	These 2 bits control “idpullup” signal in otg mode.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		2'b1x: "idpullup" always active 2'b01: "idpullup" always active when usb suspend 2'b00: use "idpullup" from otg controller	
27	INCR_MASK	This bit controls whether the ahb interface enhancement for "incr transfer" takes effect. Set this bit to 0 will active the enhancement.	RW
26	TXRISETUNE	This bit adjusts the rise/fall times of the high-speed waveform 1: -8% 0: default	RW
25	COMMONONN	This bit is the USB PHY common block power down control signal. 0: The common blocks remain powered in suspend mode 1: The common blocks are powered down in suspend mode	RW
24	VBUSVLDEXT	External VBUS Valid Indicator This signal is valid in Device mode and only when the VBUSVLDEXTSEL signal is set to 1'b1. VBUSVLDEXT indicates whether the VBUS signal on the USB cable is valid. In addition, VBUSVLDEXT enables the pull-up resistor on the D+ line. VBUSVLDEXT does not change the OTGSESSVLD0 output. 1: The VBUS signal is valid, and the pull-up resistor on D+ is enabled. 0: The VBUS signal is not valid, and the pull-up resistor on D+ is disabled. In Host mode, this input is not used.	RW
23	VBUSVLDEXTSEL	External VBUS Valid Select This signal selects either the VBUSVLDEXT input or the internal Session Valid comparator to detect whether the VBUS signal on the USB cable is valid and to assert the DP pull-up resistor. To avoid potential glitches in DP , VBUSVLDEXTSEL must be static prior to a power-on reset and remain static during normal operation. 1: The VBUSVLDEXT input is used to assert the DP pull-up resistor. 0: The internal Session Valid comparator is used to assert the DP pull-up resistor.	RW
22	POR	This signal resets all test registers and state machines in the USB PHY The POR signal must be asserted for a minimum of 10 µs. 1: reset 0: not reset *Please make sure SUSPENDM = 1'b1, REFCLKDIV、	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		REFCLKSEL and other USB PHY input signals value are correct before assert POR. *The PHYCLOCK will available after POR deasserted 45us..	
21	SIDDQ	This bit is the USB PHY analog blocks power down signal. Please make this bit to 0.	RW
20	OTG_DISABLE	This bit is the power control for otg block in USB PHY. It can powers down the VBUS Valid comparator, but not the Session Valid comparator,, nor the ID detection circuitry. To save power, if the application does not use the OTG function, this input can be set high. 1: The OTG block is powered down. 0: The OTG block is powered up.	RW
19:17	COMPDISTUNE	These bits control disconnect threshold adjustment. 3'b111 +4.5% 3'b110 +3% 3'b101 +1.5% 3'b100 Default 3'b011 -1.5% 3'b010 -3% 3'b001 -4.5% 3'b000 -6%	RW
16:14	OTGTUNE	These bits control VBUS valid threshold adjustment. 3'b111 +4.5% 3'b110 +3% 3'b101 +1.5% 3'b100 Default 3'b011 -1.5% 3'b010 -3% 3'b001 -4.5% 3'b000 -6%	RW
13:11	SQRXTUNE	These bits control squelch threshold adjustment. 3'b111 -20% 3'b110 -15% 3'b101 -10% 3'b100 -5% 3'b011 default 3'b010 +5% 3'b001 +10% 3'b000 +15%	RW
10:7	TXFSLSTUNE	These bits control FS/LS source impedance adjustment. 4'b1111 -5% 4'b0111 -2.5%	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		4’b0011	Default	
		4’b0001	+2.5%	
		4’b0000	+5%	
6	TXPREEMPHTUNE	This bit controls HS transmitter Pre-emphasis enable. 1: enable 0: disable		RW
5:4	TXHSXVTUNE	These bits adjust the voltage at which dp and dm signals cross while transmitting in HS mode.		RW
		2’b11	Default	
		2’b10	+15mv	
		2’b01	-15mv	
		2’b00	reserved	
3:0	TXVREFTUNE	These bits control HS DC voltage level adjustment.		RW
		4’b1111	+12.5%	
		4’b1110	+11.25%	
		4’b1101	+10%	
		4’b1100	+8.75%	
		4’b1011	+7.5%	
		4’b1010	+6.255	
		4’b1001	+5%	
		4’b1000	+3.75%	
		4’b0111	+2.5%	
		4’b0110	+1.25%	
		4’b0101	Default	
		4’b0100	-1.25%	
		4’b0011	-2.5%	
		4’b0010	-3.75%	
		4’b0001	-5%	
		4’b0000	-6.25%	

19.1.2.24 USB Reset Detect Timer Register

It is initialized to 0x2000096

	USBRDT																															BASE+0x 0040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved					HB_MASK	VBFIL_LD_EN	IDDIG_EN	IDDIG_REG	USBRDT																						
RST	?	?	?	?	?	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:27	Reserved	Writing has no effect, read as zero.	R
26	HB_MASK	Halfword/Byte transfer support mask. 0: enable 1: mask	RW
25	VBFIL_LD_EN	VBUS filter data load enable.	RW
24	IDDIG_EN	This bit indicates using IDDIG_REG to control “iddig” signal.	RW
23	IDDIG_REG	This bit controls “iddig” when IDDIG_REG_EN = 1’b1.	RW
22:0	USBRDT	These bits control USB reset detect time.	RW

19.1.2.25 USB VBUS Jitter Filter Register

It is initialized to 0xff0080

	USBVBFIL																BASE+0x 0044															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IDDIGFIL																USBVBFIL															
RST	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	IDDIGFIL	These bits controls iddig jitter filter time.	RW
15:0	USBVBFIL	These bits controls VBUS jitter filter time.	RW

19.1.2.26 USB Parameter Control Register1

The USBPCR1 is a 32-bit read/write register that allows software to control USB PHY some functions. It is initialized to 0x8dc23370..

	USBPCR1																BASE+0x 0048															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVLD_REG	Reserved			REFCLKSEL		REFCLKDIV		DMPD1	DPPD1	PORT0_RST	PORT1_RST	WORD_IF0	WORD_IF1	COMPDISTUNE1		SQRXTUNE1				TXFSLSTUNE1				TXPREEMPHTUNE1	TXHSXVTUNE1		TXVREFTUNE1				TXRISETUNE1
RST	1	0	0	0	1	1	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	1	1	0	1	1	1	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
30	BVLD_REG	This bit is used to set “bvalid”(VBUS above B-device session threshold) signal.	
30:28	Reserved	Writing has no effect, read as zero.	
27:26	REFCLKSEL	Reference Clock Select for USB PHY. 11, 10: The PLL uses CLKCORE as reference. 01: The XO block uses an external, 2.5-V clock supplied on the XO pin. 00: The XO block uses the clock from a crystal. This bus is a strapping option that must be set prior to a power-on reset and remain static during normal operation. Please fix it to 2'b11 or 2'b10.	
25:24	REFCLKDIV	This bus selects the reference clock frequency 11: 19.2MHz 10: 48MHz 01: 24MHz 00: 12MHz	
23	DMPD1	1: The pull-down resistance on D- of port1 is enabled 0: The pull-down resistance on D- of port1 is disabled.	
22	DPPD1	1: The pull-down resistance on D+ of port1 is enabled 0: The pull-down resistance on D+ of port1 is disabled.	
21	PORT0_RST	Port0 reset: 1: The transmit and receive finite state machines are reset, and the linestate logic combinatorially reflects the state of the single-ended receivers 0: The transmit and receive finite machines are operational, and the linestate inputs becomes sequential after 11 PHYCLOCK cycles.	
20	PORT1_RST	Port1 reset: 1: The transmit and receive finite state machines are reset, and the linestate logic combinatorially reflects the state of the single-ended receivers 0: The transmit and receive finite machines are operational, and the linestate inputs becomes sequential after 11 PHYCLOCK cycles.	
19	WORD_IF0	This bit selects utmi data bus width of port0 1: 16bit/30M 0: 8bit/60M	
18	WORD_IF1	This bit selects utmi data bus width of port1. 1: 16bit/30M 0: 8bit/60M	
17:15	COMPDISTUNE1	These bits control disconnect threshold adjustment. 3’b111 +4.5% 3’b110 +3% 3’b101 +1.5%	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		3’b100	Default	
		3’b011	-1.5%	
		3’b010	-3%	
		3’b001	-4.5%	
		3’b000	-6%	
14:12	SQRXTUNE1	These bits control squelch threshold adjustment.		
		3’b111	-20%	
		3’b110	-15%	
		3’b101	-10%	
		3’b100	-5%	
		3’b011	default	
		3’b010	+5%	
		3’b001	+10%	
		3’b000	+15%	
11:8	TXFSLSTUNE1	These bits control FS/LS source impedance adjustment.		**RW**
		4’b1111	-5%	
		4’b0111	-2.5%	
		4’b0011	Default	
		4’b0001	+2.5%	
		4’b0000	+5%	
7	TXPREEMPHTUNE1	This bit controls HS transmitter Pre-emphasis enable. 1: enable 0: disable		**RW**
6:5	TXHSXVTUNE1	These bits adjust the voltage at which dp and dm signals cross while transmitting in HS mode.		RW
		2’b11	Default	
		2’b10	+15mv	
		2’b01	-15mv	
		2’b00	reserved	
4:1	TXVREFTUNE1	These bits control HS DC voltage level adjustment.		RW
		4’b1111	+12.5%	
		4’b1110	+11.25%	
		4’b1101	+10%	
		4’b1100	+8.75%	
		4’b1011	+7.5%	
		4’b1010	+6.255	
		4’b1001	+5%	
		4’b1000	+3.75%	
		4’b0111	+2.5%	
		4’b0110	+1.25%	
		4’b0101	Default	
		4’b0100	-1.25%	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		4'b0011	-2.5%	
		4'b0010	-3.75%	
		4'b0001	-5%	
		4'b0000	-6.25%	
0	TXRISETUNE1	This bit adjusts the rise/fall times of the high-speed waveform 1: -8% 0: default		RW

Note:

*[1]: port1 is uhc, port 0 is otg

19.1.2.27 PLL Control Register

The PLL Control Register (CPPCR) is a 32-bit read/write register, which controls PLL some controls bits and stabilize time. It is initialized to 0x80000020 only by any reset. Only word access can be used on CPPCR. It is used to control all PLL.

	CPPCR																															BASE+0x 000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A_DACPD	APLL_BP	M_DACPD	MPLL_BP	Reserved																				PLLST							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bits	Name	Description	RW
31	A_DACPD	**APLL DAC Power Down** **0 - active mode** **1- test mode** **Don't change the value**	RW
30	APLL_BP	APLL ByPass Mode: 0 - not bypass 1- bypass mode	RW
29	M_DACPD	MPLL DAC Power Down 0 - active mode 1- test mode Don't change the value	RW
27:8	Reserved		R
7:0	PLLST	PLL Stabilize Time. Specifies the PLL stabilize time by unit of RTCCLK (approximate 32kHz) cycles. It is used when change PLL multiplier or change PLL from off to on. It is initialized to H'20. It is used to control all PLL.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.2.28 APLL Control Register

The APLL Control Register (CPAPCR) is a 32-bit read/write register, which controls PLL multiplier, on/off state and stabilize time. It is initialized to 0x00000000 only by any reset. Only word access can be used on CPAPCR.

	CPAPCR																															BASE+0x 0010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APLLM												APLLN						APLLOD1			APLLOD0			A_DSMPD	A_PHASEPD	A_POSTDIVPD	A_VCOPD	APLL_ON	ALOCK	Reserved	APLLEN
RST	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	1	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	APLLM	APLL feedback 12-bit multiplication factor NF = APLLM[11:0]	RW
19:14	APLLN	APLL input reference 6-bit divider. NR = APLLN[5:0]	RW
13:11	APLLOD1	APLL post VCO divider1 OD1 = APLLOD1[2:0]	RW
10:8	APLLOD0	APLL post VCO divider0 OD0 = APLLOD0[2:0]	RW
7	A_DSMPD	Power Down Delta-Sigma Modulator 0 - DSM is active 1- DSM is power down	RW
6	A_PHASEPD	Power down 4 phase clock generator 0 - active 1- power down	RW
5	A_POSTDIVPD	PLL post divide power down 0 - active 1 - power down	R
4	A_VCOPD	**VCO output Power Down** **0 - active** **1- power down**	RW
3	APLL_ON	PLL ON Flag. 0: PLL is off or not stable 1: PLL is on and stable	R
2	ALOCK		R
1	Reserved		RW
0	APLLEN	APLL Enable. When APLLEN is set to 1, APLL starts to lock phase. After APLL stabilizes, APLL_ON bit is set.	RW

Note: APLLOD0 is connected to POSTDIV2, APLLOD1 is connected to POSTDIV1. and POSTDIV1 must be equal or greater than POSTDIV2. so APLLOD1 must be equal or greater than APLLOD0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.2.29 MPLL Control Register

The MPLL Control Register (CPMPCR) is a 32-bit read/write register, which controls PLL multiplier, on/off state and stabilize time. It is initialized to 0x00000000 only by any reset. Only word access can be used on CPMPCR.

	CPMPCR																															BASE+0x 0014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPLLM												MPLLN						MPLLOD1			MPLLOD0			M_DSMPD	M_PHASEPD	M_POSTDIVPD	M_VCOPD	MPLL_ON	MLOCK	Reserved	MPLLEN
RST	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	1	0	0	0	0	0	0

Bits	Name	Description	RW
31:20	MPLLM	MPLL feedback 12-bit multiplication factor NF = MPLLM[11:0]	RW
19:14	MPLLN	MPLL input reference 6-bit divider. NR = MPLLN[5:0]	RW
13:11	MPLLOD1	MPLL post VCO divider1 OD1 = MPLLOD1[2:0]	RW
10:8	MPLLOD0	MPLL post VCO divider0 OD0 = MPLLOD0[2:0]	RW
7	M_DSMPD	Power Down Delta-Sigma Modulator 0 - DSM is active 1- DSM is power down	RW
6	M_PHASEPD	Power down 4 phase clock generator 0 - active 1- power down	RW
5	M_POSTDIVPD	PLL post divide power down 0 - active 1 - power down	R
4	M_VCOPD	**VCO output Power Down** **0 - active** **1- power down**	RW
3	MPLL_ON	PLL ON Flag. 0: PLL is off or not stable 1: PLL is on and stable	R
2	ALOCK		R
1	Reserved		RW
0	MPLLEN	MPLL Enable. When MPLLEN is set to 1, MPLL starts to lock phase. After MPLL stabilizes, MPLL_ON bit is set.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Note: MPLLOD0 is connected to POSTDIV2, MPLLOD1 is connected to POSTDIV1. and POSTDIV1 must be equal or greater than POSTDIV2. so MPLLOD1 must be equal or greater than MPLLOD0

19.1.2.30 APLL Frac Register

The APLL Frac Register (CPAPACR) is a 32-bit read/write register, which controls PLL Fractional value. It is initialized to 0x00000000 only by any reset. Only word access can be used on CPAPACR.

	CPAPACR																															BASE+0x 0018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								FRAC																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved		R
23:0	FRAC	Fractional part of the APLL feedback divide value controlled by the Delta-Sigma modulator. Total feedback divide value is (FBDIV + FRAC/2^{24}).	RW

19.1.2.31 MPLL Frac Register

The MPLL Frac Register (CPMPACR) is a 32-bit read/write register, which controls PLL Fractional value. It is initialized to 0x00000000 only by any reset. Only word access can be used on CPMPACR.

	CPMPACR																															BASE+0x 001C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								FRAC																							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved		R
23:0	FRAC	Fractional part of the MPLL feedback divide value controlled by the Delta-Sigma modulator. Total feedback divide value is (FBDIV + FRAC/2^{24}).	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.1.3 PLL Operation

The PLL developed as a macro cell for clock generator. It can generate a stable high-speed clock from a slower clock signal. The output frequency is adjustable and can be up to 2400MHz. The PLL integrates a phase frequency detector (PFD), a low pass filter (LPF), a voltage controlled oscillator (VCO) and other associated support circuitry. All fundamental building blocks as well as fully programmable dividers are integrated on the core. It is useful for clock multiplication of stable crystal oscillator sources and for de-skew clock signals.

The PLL block diagram is shown in following figure:

Figure 19-1 Block Diagram of PLL

19.1.3.1 PLL Configuration

PLL Divider Value Setting

There are 3 divider values (N, M and NO) to set the PLL output clock frequency CLKOUT:

The PLL output frequency, FOUT, is determined by the ratio set between the value set in the input divider and the feedback divider. PLL output frequency FOUT is calculated from the following equations:

FBDIV = PLLM ;

REFDIV = PLLN ;

POSTDIV2= PLLOD0;

POSTDIV1 = PLLOD1;

FVCO = FREF * FBDIV / REFDIV

FOUTPOSTDIV = FVCO / POSTDIV2/ POSTDIV1

FREF range between 1M -800M

FVCO should between600M – 2400M; 600M is the lowest power mode, while 2400M is the lowest gitter mode

FOUTPOSTDIV between 12M - 2400M

Note0：POSTDIV1 should always be programmed to be greater than or equal to POSTDIV2

Not:1: PLLM, PLLN, PLLOD0 and PLLOD1 shoule not be set 0.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Note2：when you want to change the frequcy of any PLL, you should disable the PLL first and after rewite the FBDIV, **REFDIV, POSTDIV2 and POSTDIV1, then you can enable the PLL agagin.**

19.1.4 Main Clock Division Change Sequence

Main clock (CCLK, L2CLK, H2CLK, PCLK) frequencies can be changed separately or simultaneously by changing division ratio. Following conditions must be obeyed:

1 the frequency of CCLK must be 1,2,3 or 4 times of the frequency of L2CLK

2 the frequency of H2CLK must be 1 or 2 times of the frequency of PCLK

Don't violate this limitation, otherwise unpredictable error may occur.

Important Note: When cpu enters sleep mode, the apllen must be set to 1.

19.2 Power Manager

In the Low-Power mode, part or whole processor is halted. This will reduce power consumption. The Power Management Controller contains low-power mode control and reset sequence control.

19.2.1 Low-Power Modes and Function

The processor supports six low-power modes and function:

- NORMAL mode

  In Normal mode, all peripherals and the basic blocks including power management block, the CPU core, the bus controller, the memory controller, the interrupt controller, DMA, and the external master may operate completely. But, the clock to each peripheral, except the basic blocks, can be stopped selectively by software to reduce the power consumption.

- DOZE mode

  DOZE mode is entered by setting DOZE bit of LCR to 1. In DOZE mode, clock is burst to CPU core and the clock duty is set by DUTY field of LCR. DOZE mode is canceled by reset, interrupt or clearing DOZE bit to 0. Continuous clock is supplied immediately after DOZE mode is canceled. The other Clocks except CCLK run continuously in DOZE mode.

- IDLE mode

  In IDLE mode, the clock to the CPU core is stopped except the bus controller, the memory controller, the interrupt controller, and the power management block. To exit the IDLE mode, the any interrupts should be activated.

- SLEEP mode

  In SLEEP mode, all clocks except RTC clock are disabled. PLL is disabled also. SLEEP mode is canceled by reset or interrupt. When SLEEP mode is canceled, PLL is restarted ,

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

the PLL needs clock stabilization time (PLL lock time). This PLL stabilization time is automatically inserted by the internal logic with lock time count register. and all clocks start operating after PLL stability time.

- CLOCK GATE function
  CLOCK GATE function is used to gate specified on-chip module when it is not used. Set specified CLKG0~40 bits in CLKGR will enter specified CLK gate function. CLOCK gate function is canceled by reset or clearing specified CLKGR0~40 to 0.

- Power down Mode
  In order to reduce power leakage, software may shut down power supply for AHB1 and GPS module. When system enters into SLEEP mode, the software may shut down power for J1 according to OPCR.PD bit.

19.2.2 Register Description

All PMC register 32bit access address is physical address.

Table 19-3 Power/Reset Management Controller Registers Configuration

Name	description	RW	Initial Value	Address	Access Size
LCR	Low Power Control Register	RW	0x00000000	0x0004	32
PGR	Power Gate Register	RW	0x00000000	0x00E4	32
SPCR0	SRAM Power Control Register0	RW	0x00000000	0x 00B8	32
PSWC0ST	Power Switch Chain0 Start Time	RW	0x00000000	0x 0090	32
PSWC1ST	Power Switch Chain1 Start Time	RW	0x00000000	0x 0094	32
PSWC2ST	Power Switch Chain2 Start Time	RW	0x00000000	0x 0098	32
PSWC3ST	Power Switch Chain3 Start Time	RW	0x00000000	0x 009c	32
CLKGR0	Clock Gate Register0	RW	0x1FFFFF80	0x 0020	32
CLKGR1	Clock Gate Register1	RW	0x000021FF	0x 0028	32
SRBC	Soft Reset and Bus Control Register	RW	0x00000000	0x 00C4	32
SLBC	Sleep Boot Control Register	RW	0x00000000	0x 00C8	32
SLPC	Sleep PC Register	RW	0x????????	0x 00CC	32
ERNG	Enable rng regsiter	RW	0x00000000	0x 00D8	32
RNG	rng regsiter	RW	0x????????	0x 00DC	32
OPCR	Oscillator and Power Control Register	RW	0x10801500	0x 0024	32

19.2.2.1 Low Power Control Register

The Low Power Control Register (LCR) is a 32-bit read/write register that controls low-power mode status. It is initialized to 0x00000000 by any reset.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LCR																															BASE+0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD_P0	PD_P1	PD_VPU	PD_GPU	PD_ISP	PD_H2D	P0S	P1S	VPUS	GPUS	ISPS	H2DS	PST												PD_DMIC	DMIC_S	Reserved		IDLE_DS	SLEEP_DS	LPM	
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	PD_P0	Power Down big core CPU 0: not shut down power supply to big core CPU 1: shut down power supply to big core CPU	RW
30	PD_P1	Power Down small core CPU. 0: not shut down power supply to small core CPU 1: shut down power supply to small core CPU	RW
29	PD_VPU	Power Down Module VPU. 0: not shut down power supply to VPU 1: shut down power supply to VPU	RW
28	PD_GPU	Power Down Module GPU. 0: not shut down power supply to GPU 1: shut down power supply to GPU	RW
27	PD_ISP	Power Down Module ISP. 0: not shut down power supply to ISP 1: shut down power supply to ISP	RW
26	PD_H2D	Power Down Module IPU 0: not shut down power supply to IPU 1: shut down power supply to IPU Warning: this bit is used to control IPU's isolation	RW
25	P0S	Big core CPU power down status. 0: Big core CPU module not shut down 1: Big core CPU module has entered shut down mode	R
24	P1S	Small core CPU power down status. 0: Small core CPU module not shut down 1: Small core CPU module has entered shut down mode	
23	VPUS	VPU power down status. 0: VPU module not shut down 1: VPU module has entered shut down mode	R
22	GPUS	GPU power down status. 0: GPU module not shut down 1: GPU module has entered shut down mode	R
21	ISPS	ISP power down status. 0: ISP module not shut down	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: ISP module has entered shut down mode	
20	H2D	H2D power down status. 0: H2D module not shut down 1: H2D module has entered shut down mode Warning: this bit is used to indicate IPU's shutdown state.	R
19:8	PST	Power stability Time. Specifies the Power stabilize time by unit of RTCCLK (approximate 32kHz) cycles.	RW
7	PD_DMIC	Power Dwon DMIC	
6	DMIC_S	DMIC pwer down status	
5:4	Reseved		R
3	IDLE_DS	When enter idle mode, set IDLE_DS to 1, the cache and p1's RAM enters rentention mode, software should set OPRCS.mask_int to 1	RW
2	SLEEP_DS	When enter sleep mode, set SLEEP_DS 1, only OPCR.pd bit = 00, Small core cpu and L2cache RAM enters retention mode. This is only don't shut down P0o or P1 0: cache dont' retention mode 1: cache enters retention mode	RW
1:0	LPM	Low Power Mode. Specifies which low-power mode will be entered when SLEEP instruction is executed. Bit 1~0: **00**: IDLE mode will be entered when SLEEP instruction is executed **01**: SLEEP mode will be entered when SLEEP instruction is executed	RW

Note1: software should use P0 to powerdown P1, or P1 to powerdown P0. not use the same core to powerdown itself.

Note2: Before you want to power down cpu buy set PD_P0, PD_P1 or PD in OPCR, you need to reduce cpu's frequency as the same as L2C's freq; or else unpredictable error will accurs when CPU is waken up again.

19.2.2.2 Power Gate Register

	PGR																															BASE+0x 00E4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						BCHS	HASHS	Reserve			LCDCS	USBS	UHCS	PFMONS	CSIS	DSIS						BCH	HASH	Reserved		EPDC	LCDC	USB	UHC	PFMON	CSI	DSI
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EPDC : do not change this bit.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.2.3 Guard Safe Reset Register

	GSRST																															BASE+0x 00E0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																GSRST															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

If this register is written to be 16'hfb75, the hole chip will be reset, then run uboot again.But this function can not be used.

19.2.2.4 SRAM Power Control Register0

The SRAM Power Control Register0 (SPCR0) is a 32-bit read/write register that controls SRAM power status. It is initialized to 0x00000000 by any reset.

	SPCR0																															BASE+0x 00B8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	pdma_ds	Reserved			NFI	AES	SSI1	PCM	SSI2	I2S	TCU	UART4	UART3	UART2	UART1	UART0	PDMA	UHC	MSC2	MSC1	OTG0	LCD	EPD	DSI	DES	SADC	SSI0	MSC0	BCH	AHB2_BRI	NEMC	DDR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	**PDMA_DS**	**Deep sleep of pdma**	
30:28	Reserved		R
27	NFI	Memory Power control for NFI 0: on 1: shutdown	
26	AES	Memory Power control for AES 0: on 1: shutdown	
25	SSI1	Memory Power control for SSI1 0: on 1: shutdown	RW
24	PCM	Memory Power control for PCM 0: on 1: shutdown	RW
23	SSI2	Memory Power control for SSI2 0: on 1: shutdown	RW
22	I2S	Memory Power control for I2S 0: on	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: shutdown	
21	TCU	Memory Power control for TCU 0: on 1: shutdown	RW
20	UART4	Memory Power control for UART4 0: on 1: shutdown	RW
19	UART3	Memory Power control for UART3 0: on 1: shutdown	RW
18	UART2	Memory Power control for UART2 0: on 1: shutdown	RW
17	UART1	Memory Power control for UART1 0: on 1: shutdown	RW
16	UART0	Memory Power control for UART0 0: on 1: shutdown	RW
15	PDMA	Memory Power control for PDMA 0: on 1: shutdown	RW
14	UHC	Memory Power control for UHC 0: on 1: shutdown	RW
13	MSC2	Memory Power control for MSC2 0: on 1: shutdown	RW
12	MSC1	Memory Power control for MSC1 0: on 1: shutdown	RW
11	OTG	Memory Power control for OTG 0: on 1: shutdown	RW
10	LCD	Memory Power control for LCD 0: on 1: shutdown	RW
9	EPD	Memory Power control for EPD 0: on 1: shutdown	RW
8	DSI	Memory Power control for DSI 0: on 1: shutdown	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7	DES	Memory Power control for DES 0: on 1: shutdown	RW
6	SADC	Memory Power control for SADC 0: on 1: shutdown	RW
5	SSI0	Memory Power control for SSI0 0: on 1: shutdown	RW
4	MSC0	Memory Power control for MSC0 0: on 1: shutdown	RW
3	BCH	Memory Power control for BCH 0: on 1: shutdown	RW
2	AHB2_BRI	Memory Power control for AHB2_BRI 0: on 1: shutdown	RW
1	NEMC	Memory Power control for NEMC 0: on 1: shutdown	RW
0	DDR	Memory Power control for DDR 0: on 1: shutdown	RW

Important Note :

When shutting down the power of more than one ram, the corresponding controlling bits can be set to 1 at the same time.

But, when supplying power to more than one ram, these bits should be set to 0 sequentially.

It will take at least 1us before ram can work from power off to power on

Note : When change sram power value (in spite of on or shutdown), it must enable the corresponding module clock, set the corresponding CLKGR[x] value equal 0. Afer delay some times (about 1-2us), you can operate the module normaly.

19.2.2.5 Power Switch Chain0 Start Time Register

	PSWC0ST																															BASE+0x 0090
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				PSWC0ST											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.2.6 Power Switch Chain1 Start Time Register

	PSWC1ST																															BASE+0x 0094
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				PSWC1ST											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

19.2.2.7 Power Switch Chain2 Start Time Register

	PSWC2ST																															BASE+0x 0098
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				PSWC2ST											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

19.2.2.8 Power Switch Chain3 Start Time Register

	PSWC3ST																															BASE+0x 009C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				PSWC3ST											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

NOTE: The Start Time by the unit of PCLK cycles.

19.2.2.9 Clock Gate Register0

The Clock Gate Register (CLKGR0) is a 32-bit read/write register that controls the CLOCK GATE function of peripherals. It is reset to 0x1FFFFF80.

	CLKGR																															BASE+0x 0020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLKGR0																															
RST	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	CLKGR0	Clock gate Bits. Controls the clock supplies to some peripherals. If set, clock supplies to associated devices are stopped, and registers of the device cannot be accessed also.	RW

Bit	Module	Description
31	DDR	Gate ddr's clk
30	TCU	Gate TCU's pclk
29	RTC	Gate pclk of RTC on apb
28	DES	After reset period, the clock is stopped.
27	PCM	After reset period, the clock is stopped.
26	DSI	After reset period, the clock is stopped.
25	CSI	After reset period, the clock is stopped.
24	LCD	After reset period, the clock is stopped.
23	ISP	After reset period, the clock is stopped.
22	UHC	After reset period, the clock is stopped.
21	PDMA	After reset period, the clock is stopped.
20	SSI2	After reset period, the clock is stopped.
19	SSI1	After reset period, the clock is stopped.
18	UART4	After reset period, the clock is stopped.
17	UART3	After reset period, the clock is stopped.
16	UART2	After reset period, the clock is stopped.
15	UART1	After reset period, the clock is stopped.
14	UART0	After reset period, the clock is stopped.
13	SADC	After reset period, the clock is stopped.
12	MSC2	After reset period, the clock is stopped.
11	AIC	After reset period, the clock is stopped.
10	SMB3	After reset period, the clock is stopped.
9	SMB2	After reset period, the clock is stopped.
8	SMB1	After reset period, the clock is stopped.
7	SMB0	After reset period, the clock is stopped.
6	SSI0	
5	MSC1	
4	MSC0	
3	OTG	
2	BCH	
1	NEMC	
0	NFI	

19.2.2.10 Clock Gate Regiater1

The Clock Gate Register (CLKGR1) is a 32-bit read/write register that controls the CLOCK GATE

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

function of peripherals. It is reset to 0x000021FF.

	CLKGR1																															BASE+0x 0028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																CLKGR1															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:0	CLKGR1	Clock gate Bits. Controls the clock supplies to some peripherals. If set, clock supplies to associated devices are stopped, and registers of the device cannot be accessed also.	

Bit	Module	Description
15	CPU	This does not gate any clk, 1'b1: cpm will not give stp req to CPU 1'b0:cpm will give stp req to CPU
14	APB0	This will gate p0clk of some modules on ahb0 whose slv port is apb. such ad ddr,csi and dsi
13	DLINE	Gate for delay_line's hclk
12	TCU_EXCLK	Gate for EXCLK
11	SYS_OST	Gate for sys_ost's hclk
10	AHB0	
9	P0	
8	P1	After reset period, the clock is stopped.
7	DMIC	After reset period, the clock is stopped.
6	HASH	After reset period, the clock is stopped.
5	AES	After reset period, the clock is stopped.
4	EPD	After reset period, the clock is stopped.
3	AHB_MON	After reset period, the clock is stopped.
2	IPU	After reset period, the clock is stopped
1	GPU	After reset period, the clock is stopped.
0	VPU	After reset period, the clock is stopped.

Note: you can use XXCDR.XX_STOP, CLKGR.XX or SRBC.XX_STP to stop XX module's clk, and when CLKGR.XX or SRBC.XX_STP is set high , CPU can turn into sleep mode successfully, but if only set XXCDR.XX_STOP high, CPU may not turn into sleep mode, becase XX module can not give out stop_ack when clk is stopped; CPM will not wait for XX module's stop_ack if you stop it's clk by set CLKGR.XX.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.2.11 Soft Reset and Bus Control Register (SRBC)

The Soft Reset and Bus Control Register is a 32-bit read/write register that control some module reset and bus transmission. It is initialized to 0x0000000 0by reset.

	SRBC																															BASE+0x 00C4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPU_SR	VPU_STP	VPU_ACK	GPU_SR	GPU_STP	GPU_ACK	LCD_SR	LCD_STP	LCD_ACK	ISP_SR	ISP_STP	ISP_ACK	H2D_SR	H2D_STP	H2D_ACK	Reserved	CPU_STP	CPU_ACK	UHC_SR	OTG1_SR	OTG0_SR	AHB0_STP	AHB0_ACK	AHB2_STP	AHB2_ACK	DDR_STP	DDR0_ACK	Reserved				
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	VPU_SR	0: VPU does not enter soft reset mode 1: VPU enters soft reset mode	RW
30	VPU_STP	Request for VPU to Stop bus transfer.	RW
29	VPU_ACK	VPU Stop Ack.	R
28	GPU_SR	0: GPU does not enter soft reset mode 1: GPU enters soft reset mode	RW
27	GPU_STP	Request for GPU to Stop bus transfer.	RW
26	GPU_ACK	GPU Stop Ack.	R
25	LCD_SR	0: LCD does not enter soft reset mode 1: LCD enters soft reset mode	RW
24	LCD_STP	Request for LCD to Stop bus transfer.	RW
23	LCD_ACK	LCD Stop Ack.	R
22	ISP_SR	0: ISP does not enter soft reset mode 1: ISP enters soft reset mode	RW
21	ISP_STP	Request for ISP to Stop bus transfer.	RW
20	ISP_ACK	ISP Stop Ack.	R
19	H2D_SR	0: H2D does not enter soft reset mode 1: H2D enters soft reset mode	RW
18	H2D_STP	Request for H2D to Stop bus transfer.	RW
17	H2D_ACK	H2D Stop Ack.	R
16	Reseved		
15	CPU_STP	Request for CPU to Stop bus transfer.	RW
14	CPU_ACK	CPU Stop Ack.	R
13	UHC_SR	UHC soft reset	RW
12	OTG1_SR	OTG soft reset	RW
11	OTG0_SR	OTG0 soft reset	RW
10	AHB0_STP	Request for AHB0 to Stop bus transfer.	RW
9	AHB0_ACK	AHB0 Stop Ack.	R
8	AHB2_STP	Request for AHB2 to Stop bus transfer.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7	AHB2_ACK	AHB2 Stop Ack.	R
6	DDR_STP	Request for DDR to Stop bus transfer.	RW
5	DDR_ACK	DDR Stop Ack.	R
4:0	Reserved		R

19.2.2.12 Sleep Boot Control Register (SLBC)

The Sleep Boot Control Register is a 32-bit read/write register that control sleep boot mode.It is initialized to 0x0000000 0by reset.

	SLBC																															BASE+0x 00C8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															SLBC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:1	Reserved		R
0	SLBC	0: Sleep Mode Wake up Boot Process is same with any reset. 1: Sleep Mode Wake up Boot Process is jumped to SLPC, don't care Boot_Sel. The bit is only valid When shut down P0 in Sleep Mode	RW

19.2.2.13 Sleep PC Register (SLPC)

The Sleep PC Register is a 32-bit read/write register that control sleep mode jump address.

	SLPC																															BASE+0x 00CC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SLPC																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	SLPC	When SLBC is 1, sleep boot is jumped to SLPC, not true boot as any reset. The bit is only valid When shut down P0 in Sleep Mode	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.2.14 Enable Rng Register (ERNG)

The Enable rng Register is a 32-bit read/write register that control whether enable rng regsiter.It is initialized to 0x00000000 by reset.

	ERNG																															BASE+0x 00D8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ready	Reserved																														ERNG
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	ready	Data ready When software read RNG register, hardware auto clear ready to 0.	RC W
30:1	Reserved		R
0	ERNG	0: not enable 1: enable	RW

19.2.2.15 Rng Register (RNG)

The rng Register is a 32-bit read only registers.

	RNG																															BASE+0x00DC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNG																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	RNG		R

19.2.2.16 Oscillator and Power Control Register (OPCR)

The Oscillator and Power Control Register is a 32-bit read/write register that specifies some special controls to oscillator and analog block. It is initialized to 0x00801500 by reset.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	OPCR																															BASE+0x 0024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IDLE_DIS	Mask_int_req	MASK)VOU	GPU_ACTIVE	L2C_PD	PD		REQ_MODE	RTCLK_EN	Reserved			O1ST												SPENDN0	SPENDN1	Reserved	O1SE	Reserved	ERCS	BUS_MODE	Reserved
RST	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	IDLE_DIS	0: when CPU enters idle mode, CPU clock is stopped 1: When CPU enters idle mode, CPU clock is not stopped	RW
30	MASK_INT	0: interrupt not mask 1: interrupt masked When enters sleep/idle mode, cpu should set MASK_INt 1, hardware will clear.	RW
29	MASK_VPU	Don't change this bit , it is only used to debug	RW
28	GPU_ACTIVE	GPU ACTIVE	R
27	L2C_PD	0 : L2MP doesn't power down 1: L2MP power down when CPU enters sleep mode	RW
26:25	**PD**	**00: CPU doesn't power down** **01: big core power down** **10: small core power down** **11: doesn't power down**	RW
24	**REQ_MODE**	**0: Before go into sleep mode, CPM send stop req signal to DDR** **1: Before go into sleep mode, CPM doesn't send stop req signal to DDR** **When DDR enters retention mode, software should be set the bit to 1**	RW
23	RTCLK_EN	0： RTCLK isn't enable 1: RTCLK enable This bit control the rtclk to ssi0,ssi1,ssi2,sadc and usb	RW
22：20			
19:8	O1ST	EXCLK Oscillator Stabilize Time. This filed specifies the EXCLKoscillator stabilize time by unit of 16 RTCCLK periods (oscillator stable time O1ST × 16 / 32768) cycles. It is initialized to H'15.	RW
7	SPENDN0	force port0(otg) to enter suspend mode. 0: port0(otg) has forced to entered SUSPEND mode 1: port0(otg) hasn't forced to entered SUSPEND mode	RW
6	SPENDN	force port1(uhc) to enter suspend mode.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	1	0: port1(uhc) has forced to entered SUSPEND mode 1: port1(uhc) hasn't forced to entered SUSPEND mode	
5	Reserved		RW
4	O1SE	EXCLK Oscillator Sleep Mode Enable. This filed controls the state of the EXCLK oscillator in Sleep mode. 0: EXCLK oscillator is disabled in Sleep mode 1: EXCLK oscillator is enabled in Sleep mode	RW
3	Reserved		RW
2	ERCS	EXCLK/512 clock and RTCLK clock selection. 0: select EXCLK/512 division ration clock 1: select RTCLK clock the clock only output to CPM INTC SSI TCU etc.	RW
1	BUS_MODE	The default value is 0 When the value is 1, the BUS access is acclerated.	RW
0	Reserved		

19.2.3 Doze Mode

Firstly, software should set the DUTY bits of LCR. Then set DOZE bit of LCR to 1 to enter doze mode. When slot controller of PMC indicates that the CPU clock's time-slot has expired, CPU is halted but its register contents are retained. During doze mode, program can modify clock duty-cycle according to core resource requirement. Clock control is in increments of approximately 3% (1/31).

Doze is exited by software, interrupt, reset or SLEEP instruction.

19.2.4 IDLE Mode

In normal mode, when LPM bits in LCR are 0 and SLEEP instruction is executed, the processor enters idle mode. CPU is halted but its register contents are retained All critical application must be finished and peripherals must be configured to generate interrupts when they need CPU attention.

The procedure of entering sleep mode is shown blow:

1. Set LPM bits in LCR to 0.
2. Executes SLEEP instruction.
3. When current operation of CPU core has finished and CPU core is idle, CCLK supply to CPU core is stopped.

IDLE mode is exited by an interrupt (IRQ or on-chip devices) or a reset.

19.2.5 SLEEP Mode

In normal mode, when LPM bits in LCR is 1 and SLEEP instruction is executed, the processor enter SLEEP mode. CPU and on-chip devices are halted, except some wakeup-logic. PLL is shut off.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Clock output from CKO pin is also stopped. SDRAM content is preserved by driving into self-refresh state. CPU registers and on-chip devices registers contents are retained.

Before enter SLEEP mode, software should ensure that all peripherals are not running. The procedure of entering SLEEP mode is shown blow:

1 Set LPM bit in LCR to 1.
2 Execute a SLEEP instruction.
3 When current access on system bus complete, the arbiter will not grant any following request. EMC will drive SDRAM from auto-refresh mode to self-refresh mode.
4 When system bus is idle state and SDRAM is self-refresh mode, internal clock supplies are stopped.
5 SLEEP mode can be exited by an interrupt (IRQ or on-chip devices), WDT reset or a poweron reset via the RESETP pin.
6. If Sleep mode can be wakeup by DMIC interrupt, you need to set OPCR O1SE value = 1, don't disbale EXCLK and dont CLKGR DMIC module, proper configure DMIC registers, then at some conditons, the dmic interrup cant be generated, the Sleep mode be wakedup.

19.2.6 Power Down Mode

When PD_VPU bit in LCR is 1 , the processor enters shut down VPU module power sequence.

When PD_VPUS bit in LCR is 1, it indicates that the VPU module has been shut off. The leakage current of VPU is reduced almost to 0.

When enter sleep mode, when PD bits in OPCR is 1, the J1 power supply would be shut off. The leakage current of J1 is reduced almost to 0.

The procedure of entering Power Down mode is shown blow:

1 set proper values for PSWC0ST, PSWC1ST, PSWC2ST, PSWC3ST.
2 set PD_VPU bit in LCR to 1.
3 wait until PD_VPUS = 1.
4 When need for supply power for VPU, set PD_VPU in LCR to 0.
5 wait until PD_VPUS = 0.
6 the hardware auto generate RESET signal to the module, the software must again config the module as the same to POWER ON Reset.

The same as GPU , GPS, P1, software should power on these modules one by one.

19.3 Reset Control Module

19.3.1 Register Description

All RCM register 32bit access address is physical address.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Name	description	RW	Initial Value	Address	Access Size
RSR	Reset Status Register	RW	0x????????	0x0008	32

19.3.1.1 Reset Status Register (RSR)

The Reset Status Register (RSR) is a 32-bit read/write register which records last cause of reset. Each RSR bit is set by a different source of reset. Please refer to Reset Sequence Control for reset sources description.

	RSR																															BASE+0x 0008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																												HR	P0R	WR	PR
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?

Bits	Name	Description	RW
31:4	Reserved	Writing has no effect, read as zero.	R
3	HR	Hibernate Reset. Software can only write 0,Write 1 will be ignored	RW
2	P0R	P0 power up Reset. It indicates that P0 has been shut down, now it has been power up. When P0 reset is detected, P0R is set and remains set until software clears it or another reset occurs. This bit can only be written with 0. Write with 1 will be ignored. 0: P0 reset has not occurred since the last time the software clears this bit 1: P0 reset has occurred since the last time the software clears this bit	RW
1	WR	WDT Reset. When a WDT reset is detected, WR is set and remains set until software clears it or another reset occurs. This bit can only be written with 0. Write with 1 will be ignored. 0: WDT reset has not occurred since the last time the software clears this bit 1: WDT reset has occurred since the last time the software clears this bit	RW
0	PR	Power On Reset. When a poweron reset via PRESET pin is detected, PR is set and remains set until software clears it or another reset occurs. This bit can only be written with 0. Write with 1 is ignored. 0: Power on reset has not occurred since the last time the software clears this bit 1: Power on reset has occurred since the last time the software clears this bit	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.3.2 Power On Reset

Power on reset is generated when PRESET pin is driven to low. Internal reset is asserted immediately. All pins return to their reset states. The Power on reset is extended to 40MS.

PRESET pin must be held low until power stabilizes and the EXCLK oscillator stabilize. CPU and peripherals are clocked by EXCLK oscillator output directly. PLL is reset to off state. All internal modules are initialized to their predefined reset states.

19.3.3 WDT Reset

WDT reset is generated when WDT overflow. Internal reset is asserted within two RTCCLK cycles. All pins return to their reset states.

Then WDT reset source is cleared because of internal reset. The internal reset is asserted for about 10 milliseconds. CPU and peripherals are clocked by EXCLK oscillator output directly. PLL is reset to off state.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

20 Timer/Counter Unit

20.1 Overview

The TCU (Timer/Counter with PWM output) contains 8 channels of 16-bit programmable timers (timers 0 to 5). They can be used as Timer or PWM.

TCU has the following features:

- There are two modes of TCU for the eight channels
  - TCU1: Channel 0, 3,4, 5, 6,and 7
  - TCU2: Channel 1,2
- Six independent channels, each consisting of
  - Counter
  - Data register (FULL and HALF)
  - Control register
- Independent clock for each counter, selectable by software
  - PCLK, EXTAL and RTCCLK can be used as the clock for counter
  - The division ratio of the clock can be set to 1, 4, 16, 64, 256 and 1024 by software
- FULL interrupt and HALF interrupt can be generated for each channel using the compare data registers
  - Timer 0-7 can be used as PWM (Set the initial signal level)
  - Timer 0,3-7 can be used as a counter to count external signal (like trackball)
  - Timer 5 has separated interrupt
  - Timer 0-4 and timer 6-7 has one interrupt in common
  - OST uses interrupt 0, Timer 5 uses interrupt 1, and Timer 0-4/ 6-7 uses interrupt 2
- The difference between TCU1 and TCU2
  - TCU1: It cannot work in sleep mode, but operated easily
  - TCU2: It can work in sleep mode, but operated more complicated than TCU1

20.2 Pin Description

Table 20-1 PWM Pins Description

Name	I/O	Description
PWM [7:0]	Output	PWM channel output signals.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.3 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 20-2 Registers Memory Map-Address Base

Name	Base	Description
TCU	0x10002000	Address base of TCU

Table 20-3 TCU Registers Configuration

Name	Description	RW	Reset Value	Address offset	Access Size
TSTR	Timer Status Register	R	0x00000000	0x0F0	32
TSTSR	Timer Status Set Register	W	0x????????	0x0F4	32
TSTCR	Timer Status Clear Register	W	0x????????	0x0F8	32
TSR	Timer STOP Register	R	0x00000000	0x01C	32
TSSR	Timer STOP Set Register	W	0x00000000	0x02C	32
TSCR	Timer STOP Clear Register	W	0x0000	0x03C	32
TER	Timer Counter Enable Register	R	0x0000	0x010	16
TESR	Timer Counter Enable Set Register	W	0x????	0x014	16
TECR	Timer Counter Enable Clear Register	W	0x????	0x018	16
TFR	Timer Flag Register	R	0x003F003F	0x020	32
TFSR	Timer Flag Set Register	W	0x????????	0x024	32
TFCR	Timer Flag Clear Register	W	0x????????	0x028	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

TMR	Timer Mask Register	R	0x00000000	0x030	32
TMSR	Timer Mask Set Register	W	0x????????	0x034	32
TMCR	Timer Mask Clear Register	W	0x????????	0x038	32
TDFR0	Timer Data FULL Register 0	RW	0x????	0x040	16
TDHR0	Timer Data HALF Register 0	RW	0x????	0x044	16
TCNT0	Timer Counter 0	RW	0x????	0x048	16
TCSR0	Timer Control Register 0	RW	0x0000	0x04C	16
TDFR1	Timer Data FULL Register 1	RW	0x????	0x050	16
TDHR1	Timer Data HALF Register 1	RW	0x????	0x054	16
TCNT1	Timer Counter 1	RW	0x????	0x058	16
TCSR1	Timer Control Register 1	RW	0x0000	0x05C	16
TDFR2	Timer Data FULL Register 2	RW	0x????	0x060	16
TDHR2	Timer Data HALF Register 2	RW	0x????	0x064	16
TCNT2	Timer Counter 2	RW	0x????	0x068	16
TCSR2	Timer Control Register 2	RW	0x0000	0x06C	16
TDFR3	Timer Data FULL Register 3	RW	0x????	0x070	16
TDHR3	Timer Data HALF Register 3	RW	0x????	0x074	16
TCNT3	Timer Counter 3	RW	0x????	0x078	16
TCSR3	Timer Control Register 3	RW	0x0000	0x07C	16
TDFR4	Timer Data FULL Register 4	RW	0x????	0x080	16
TDHR4	Timer Data HALF Register 4	RW	0x????	0x084	16
TCNT4	Timer Counter 4	RW	0x????	0x088	16
TCSR4	Timer Control Register 4	RW	0x0000	0x08C	16
TDFR5	Timer Data FULL Register 5	RW	0x????	0x090	16
TDHR5	Timer Data HALF Register 5	RW	0x????	0x094	16
TCNT5	Timer Counter 5	RW	0x????	0x098	16
TCSR5	Timer Control Register 5	RW	0x0000	0x09C	16
TDFR6	Timer Data FULL Register 6	RW	0x????	0x0A0	16
TDHR6	Timer Data HALF Register 6	RW	0x????	0x0A4	16
TCNT6	Timer Counter 6	RW	0x????	0x0A8	16
TCSR6	Timer Control Register 6	RW	0x0000	0x0AC	16
TDFR7	Timer Data FULL Register 7	RW	0x????	0x0B0	16
TDHR7	Timer Data HALF Register 7	RW	0x????	0x0B4	16
TCNT7	Timer Counter 7	RW	0x????	0x0B8	16
TCSR7	Timer Control Register 7	RW	0x0000	0x0BC	16
TCUMOD0	Timer control mode Register 0	RW	0x????	0x100	16
TCUMOD3	Timer control mode Register 3	RW	0x????	0x110	16
TCUMOD4	Timer control mode Register 4	RW	0x????	0x120	16
TCUMOD5	Timer control mode Register 5	RW	0x????	0x130	16
TFWD0	Timer fifo write data Register 0	RW	0x????????	0x104	32
TFWD3	Timer fifo write data Register 3	RW	0x????????	0x114	32
TFWD4	Timer fifo write data Register 4	RW	0x????????	0x124	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

TFWD5	Timer fifo write data Register 5	RW	0x????????	0x134	32
TFIFOSR0	Timer fifo state Register	R	0x??	0x108	6
TFIFOSR3	Timer fifo state Register	R	0x??	0x118	6
TFIFOSR4	Timer fifo state Register	R	0x??	0x128	6
TFIFOSR5	Timer fifo state Register	R	0x??	0x138	6

20.3.1 Timer Control Register (TCSR)

The TCSR is a 16-bit read/write register. It contains the control bits for each channel. It is initialized to 0x00 by any reset.

	TCSR0, TCSR1, TCSR2, TCSR3, TCSR4, TCSR5 TCSR6, TCSR7																BASE+0x04C, 0x05C, 0x06C, 0x07C, 0x08C, 0x09C, 0x0AC, 0x0BC															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved				BYPASS	CLRZ	SD	INITL	PWM_EN	PWM_IN_EN	PRESCALE			EXT_EN	RTC_EN	PCK_EN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:11	Reserved	Writing has no effect, read as zero.	R
11	BYPASS	PWM bypass mode. 1: If PCK_EN = 1, this channel output PIXCLK; If RTC_EN = 1, this channel output RTCCLK; If EXT_EN = 1, this channel output EXTAL; Only one of those XXX_EN is permit available during one time. 0: This BYPASS function disable. *Only when you want to let this PWM channel output some special clock (PCLK, RTCLK and EXTAL clock), you can set this register to 1. Otherwise keep it to 0. When you want to use BYPASS function, not forget offer clock supplies of relate channel (relate to register TSR, TSSR, TSCR).(the current version can not bypass pclk,and when bypass a clk,the relative timer will be stopped to reduce power consumption.)	RW
10	CLRZ	Clear counter to 0. It is only used in TCU2 mode. Writing 1 to this bit will clear the counter to 0. When the counter is finished setting to 0, it will be cleared by hardware. Writing 0 to this bit will be ignored.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td>9</td><td>SD</td><td>Shut Down (SD) the PWM output. It is only used in TCU1 mode.
0: Graceful shutdown
1: Abrupt shutdown
Graceful shutdown: The output level for PWM output will keep the level only after the comparison match of FULL.(when write register to stop the timer,the timer will continue run utill full match)
Abrupt shutdown: The output level for PWM output will keep the level as soon as software write to stop the counter.(the conter will stop as soon as softerware write to stop the counter).</td><td>RW</td></tr>
<tr><td>8</td><td>INITL</td><td>Selects an initial output level for PWM output.
0: Low
1: High</td><td>RW</td></tr>
<tr><td>7</td><td>PWM_EN</td><td>PWM output pin control bit.
0: PWM pin output disable, and the PWM pin will be set to the initial level according to INITL
1: PWM pin output enable(only channel0~channel3 are available currently)</td><td>RW</td></tr>
<tr><td>6</td><td>PWM_IN_EN</td><td>PWM input mode enable.
Set to 1 to enable this function.
In this function, PWM pin need to set GPIO as function0 to receive external signal, EXT_EN, RTC_EN, PCK_EN need to set 0. And TCNT became a counter to count this signal’s both edges. (This bit in TCSR1, 2 are Reserved).</td><td>RW</td></tr>
<tr><td>5:3</td><td>PRESCALE</td><td>These bits select the TCNT count clock frequency. Don’t change this field when the channel is running.
<table>
<tr><th>Bit 2</th><th>Bit1</th><th>Bit 0</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Internal clock: CLK/1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Internal clock: CLK/4</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Internal clock: CLK/16</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Internal clock: CLK/64</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Internal clock: CLK/256</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Internal clock: CLK/1024</td></tr>
<tr><td colspan="3">110~111</td><td>Reserved</td></tr>
</table></td><td>RW</td></tr>
<tr><td>2</td><td>EXT_EN</td><td>Select EXTAL as the timer clock input.
0: Disable
1: Enable</td><td>RW</td></tr>
<tr><td>1</td><td>RTC_EN</td><td>Select RTCCLK as the timer clock input.
0: Disable
1: Enable</td><td>RW</td></tr>
<tr><td>0</td><td>PCK_EN</td><td>Select PCLK as the timer clock input.
0: Disable
1: Enable</td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: The input clock of timer and the PCLK should keep to the rules as follows:

Input clock of timer: IN_CLK	**Clock generated from the frequency divider (PRESCALE): DIV_CLK**
PCK_EN == 0, RTC_EN == 1 and EXT_EN == 0 (IN_CLK = RTCCLK)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 0, RTC_EN == 0 and EXT_EN == 1 (IN_CLK = EXTAL)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 1, RTC_EN == 0 and EXT_EN == 0 (IN_CLK = PCLK)	ANY

20.3.2 Timer Data FULL Register (TDFR)

The comparison data FULL registers TDFR is used to store the data to be compared with the content of the up-counter TCNT. This register can be directly read and written. (Default: indeterminate) But it is not suggested changing when counter is working in TCU2 mode.

	TDFR0, TDFR1, TDFR2, TDFR3, TDFR4, TDFR5, TDFR6, TDFR7																**BASE+0x040, 0x050, 0x060, 0x070, 0x080, 0x090, 0x0A0, 0x0B0**															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TDFR															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

20.3.3 Timer Data HALF Register (TDHR)

The comparison data HALF registers TDHR is used to store the data to be compared with the content of the up-counter TCNT. This register can be directly read and written. (Default: indeterminate) But it is not suggested changing when counter is working in TCU2 mode.

	TDHR0, TDHR1, TDHR2, TDHR3, TDHR4, TDHR5, TDHR6, TDHR7																**BASE+0x044, 0x054, 0x064, 0x074, 0x084, 0x094, 0x0A4, 0x0B4**															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TDHR															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

20.3.4 Timer Counter (TCNT)

TCNT is a 16-bit read/write register. The up-counter TCNT can be reset to 0 by software and counts up

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

using the prescaler output clock. When TCNT count up to equal to TDFR, it will reset to 0 and continue to count up.

TCU1: The counter data can be read out at any time. The data can be written at any time. This makes it possible to change the interrupt and/or clock output cycles temporarily. (Default: indeterminate)

TCU2: The counter data can be read out at any time, but you should read TSTR.REALn to check whether the data is real data or not. The data can only be written before counter is started, and the counter clock is pclk. But it can be cleared to 0 by setting TCSR.CLRZ to 1, and if the counter is really cleared, TCSR.CLRZ will be set to 0 by hardware.

	TCNT0, TCNT1, TCNT2, TCNT3, TCNT4, TCNT5, TCNT6, TCNT7																**BASE+0x048, 0x058, 0x068, 0x078, 0x088, 0x098, 0x0A8, 0x0B8**															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TCNT															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

20.3.5 Timer Counter Enable Register (TER)

The TER is a 16-bit read-only register. It contains the counter enable control bits for each channel. It is initialized to 0x0000 by any reset. It can only be set by register TESR and TECR. Since the timer enable control bits are located in the same addresses, two or more timers can be started at the same time.

	TER																**BASE+0x010**															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	OSTEN	Reserved							TCEN 7	TCEN 6	TCEN 5	TCEN 4	TCEN 3	TCEN 2	TCEN 1	TCEN 0
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	OSTEN	Enable the counter in OST. 0: Stop counting up 1: Begin counting up	R
14:8	Reserved	Writing has no effect, read as zero.	R
7	TCEN 7	Enable the counter in timer 7. 0: Stop counting up 1: Begin counting up	R
6	TCEN 6	Enable the counter in timer 6.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Stop counting up 1: Begin counting up	
5	TCEN 5	Enable the counter in timer 5. 0: Stop counting up 1: Begin counting up	R
4	TCEN 4	Enable the counter in timer 4. 0: Stop counting up 1: Begin counting up	R
3	TCEN 3	Enable the counter in timer 3. 0: Stop counting up 1: Begin counting up	R
2	TCEN 2	Enable the counter in timer 2. 0: Stop counting up 1: Begin counting up	R
1	TCEN 1	Enable the counter in timer 1. 1: Begin counting up 0: Stop counting up	R
0	TCEN 0	Enable the counter in timer 0. 0: Stop counting up 1: Begin counting up	R

20.3.6 Timer Counter Enable Set Register (TESR)

The TCCSR is a 32-bit write-only register. It contains the counter enable set bits for each channel. Since the timer enable control set bits are located in the same addresses, two or more timers can be started at the same time.

	TESR																BASE+0x014															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	OSTST	Reserved							TCST 7	TCST 6	TCST 5	TCST 4	TCST 3	TCST 2	TCST 1	TCST 0
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
15	OSTST	Set OSTEN bit of TER. 0: Ignore 1: Set OSTEN bit to 1	W
14:8	Reserved	Writing has no effect, read as zero.	R
7	TCST 7	Set TCEN 7 bit of TER. 0: Ignore 1: Set TCEN 5 bit to 1	W
6	TCST 6	Set TCEN 6 bit of TER.	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Ignore 1: Set TCEN 5 bit to 1	
5	TCST 5	Set TCEN 5 bit of TER. 0: Ignore 1: Set TCEN 5 bit to 1	W
4	TCST 4	Set TCEN 4 bit of TER. 1: Set TCEN 4 bit to 1 0: Ignore	W
3	TCST 3	Set TCEN 3 bit of TER. 0: Ignore 1: Set TCEN 3 bit to 1	W
2	TCST 2	Set TCEN 2 bit of TER. 1: Set TCEN 2 bit to 1 0: Ignore	W
1	TCST 1	Set TCEN 1 bit of TER. 0: Ignore 1: Set TCEN 1 bit to 1	W
0	TCST 0	Set TCEN 0 bit of TER. 0: Ignore 1: Set TCEN 0 bit to 1	W

20.3.7 Timer Counter Enable Clear Register (TECR)

The TECR is a 32-bit write-only register. It contains the counter enable clear bits for each channel. Since the timer enable clear bits are located in the same addresses, two or more timers can be stop at the same time.

	TECR																															BASE+0x018
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	OSTCL	Reserved							TCCL 7	TCCL 6	TCCL 5	TCCL 4	TCCL 3	TCCL 2	TCCL 1	TCCL 0
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
15	OSTCL	Set OSTEN bit of TER. 0: Ignore 1: Set OSTEN 5 bit to 0	W
14:8	Reserved	Writing has no effect, read as zero.	R
7	TCCL 7	Set TCEN 7 bit of TER. 0: Ignore 1: Set TCEN 6 bit to 0	W
6	TCCL 6	Set TCEN 7 bit of TER.	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Ignore 1: Set TCEN 6 bit to 0	
5	TCCL 5	Set TCEN 5 bit of TER. 0: Ignore 1: Set TCEN 5 bit to 0	W
4	TCCL 4	Set TCEN 4 bit of TER. 1: Set TCEN 4 bit to 0 0: Ignore	W
3	TCCL 3	Set TCEN 3 bit of TER. 0: Ignore 1: Set TCEN 3 bit to 0	W
2	TCCL 2	Set TCEN 2 bit of TER. 1: Set TCEN 2 bit to 0 0: Ignore	W
1	TCCL 1	Set TCEN 1 bit of TER. 0: Ignore 1: Set TCEN 1 bit to 0	W
0	TCCL 0	Set TCEN 0 bit of TER. 0: Ignore 1: Set TCEN 0 bit to 0	W

20.3.8 Timer Flag Register (TFR)

The TFR is a 32-bit read-only register. It contains the comparison match flag bits for all the channels. It can also be set by register TFSR and TFCR. It is initialized to 0x00000000 by any reset.

	TFR																															BASE+0x020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOEFLAG5	FIFOEFLAG4	FIFOEFLAG3	FIFOEFLAG0	HFLAG 7	HFLAG 6	HFLAG 5	HFLAG 4	HFLAG 3	HFLAG 2	HFLAG 1	HFLAG 0	OSTFLAG	Reserved			FIFOFLAG5	FIFOFLAG4	FIFOFLAG3	FIFOFLAG0	FFLAG 7	FFLAG 6	FFLAG 5	FFLAG 4	FFLAG 3	FFLAG 2	FFLAG 1	FFLAG 0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:24	FIFO empty FLAG	fifo empty flag: 0:fifo not empty 1:fifo empty	R
23:16	HFLAG 7~0	HALF comparison match flag. (TCNT = TDHR) 0: Comparison not match 1: Comparison match	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15	OSTFLAG	OST comparison match flag. (OSTCNT = OSTDR) 0: Comparison not match 1: Comparison match	R
14:12	Reserved	Writing has no effect, read as zero.	R
11:8	FIFOFLAG 3~0	FIFO comparison match flag. (TCNT = TFWD) 0: Comparison not match 1: Comparison match	R
7:0	FFLAG 7~0	FULL comparison match flag. (TCNT = TDFR) 0: Comparison not match 1: Comparison match	R

20.3.9 Timer Flag Set Register (TFSR)

The TFSR is a 32-bit write-only register. It contains the comparison match flag set bits for all the channels.

	TFSR																															BASE+0x024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOEST5	FIFOEST4	FIFOEST3	FIFOEST0	HFST 7	HFST 6	HFST 5	HFST 4	HFST 3	Reserved		HFST 0	OSTFST	Reserved			FIFOST5	FIFOST4	FIFOST3	FIFOST0	FFST 7	FFST 6	FFST 5	FFST 4	FFST 3	Reserved		FFST 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:24	FIFOEST 3~0	Set FIFOEFLAG bit of TFR 0: Ignore; 1: Set FIFOEFLAG n bit to 1.	W
23:19	HFST 7~3	Set HFLAG n bit of TFR. 0: Ignore 1: Set HFLAG n bit to 1	W
18:17	Reserved		
16	HFST0	Set HFLAG 0 bit of TFR. 0: Ignore 1: Set HFLAG n bit to 1	W
15	OSTFST	Set OSTFLAG n bit of TFR. 0: Ignore 1: Set OSTFLAG n bit to 1	W
14:12	Reserved	Writing has no effect, read as zero.	R
11:8	FIFOST 3~0	Set FIFOFLAG n bit of TFR. 0: Ignore; 1: Set FIFOFLAG n bit to 1.	W
7:3	FFST 7~0	Set FFLAG n bit of TFR. 0: Ignore	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Set FFLAG n bit to 1	
2:1	Reserved		
0	FFST0	Set FFLAG 0 bit of TFR. 0: Ignore 1: Set FFLAG n bit to 1	

20.3.10 Timer Flag Clear Register (TFCR)

The TFCR is a 32-bit write-only register. It contains the comparison match flag clear bits for all the channels.

	TFCR																															BASE+0x028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOECL5	FIFOECL4	FIFOECL3	FIFOECL0	HFCL 7	HFCL 6	HFCL 5	HFCL 4	HFCL 3	HFCL 2	HFCL 1	HFCL 0	OSTFCL	Reserved			FIFOCL5	FIFOCL4	FIFOCL3	FIFOCL0	FFCL 7	FFCL 6	FFCL 5	FFCL 4	FFCL 3	FFCL 2	FFCL 1	FFCL 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:24	FIFOECL 3~0	Set FIFOEFLAG n bit of TFR. 0: Ignore 1: Set FIFOEFLAG n bit to 0	
23:16	HFCL 7~0	Set HFLAG n bit of TFR. 0: Ignore 1: Set FFLAG n bit to 0	W
15	OSTFCL	Set OSTFLAG n bit of TFR. 0: Ignore 1: Set OSTFLAG n bit to 0	W
14:12	Reserved	Writing has no effect, read as zero.	R
11:8	FIFOCL 3~0	Set FIFOFLAG n bit of TFR. 0: Ignore 1: Set FIFOFLAG n bit to 0	W
7:0	FFCL 7~0	Set FFLAG n bit of TFR. 0: Ignore 1: Set FFLAG n bit to 0	W

20.3.11 Timer Mast Register (TMR)

The TMR is a 32-bit read-only register. It contains the comparison match flag bits for all the channels. It is initialized to 0x003F003F by any reset. It can only be set by register TMSR and TMCR.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TMR																															BASE+0x030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOEMASK5	FIFOEMASK4	FIFOEMASK3	FIFOEMASK0	HMASK 7	HMASK 6	HMASK 5	HMASK 4	HMASK 3	HMASK 2	HMASK 1	HMASK 0	OSTMASK	Reserved			FIFOMASK5	FIFOMASK4	FIFOMASK3	FIFOMASK0	FMASK 7	FMASK 6	FMASK 5	FMASK 4	FMASK 3	FMASK 2	FMASK 1	FMASK 0
RST	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:24	FIFOEMASK 3~0	FIFO empty interrupt mask. 0:empty interrupt not mask 1:empty interrupt mask	
23:16	HMASK 7~0	HALF comparison match interrupt mask. 0: Comparison match interrupt not mask 1: Comparison match interrupt mask	R
15	OSTMASK	OST comparison match interrupt mask. 0: Comparison match interrupt not mask 1: Comparison match interrupt mask	R
14:12	Reserved	Writing has no effect, read as zero.	R
11:8	FIFOMASK 3~0	FIFO comparison match interrupt mask. 0: Comparison match interrupt not mask 1: Comparison match interrupt mask	R
7:0	FMASK 7~0	FULL comparison match interrupt mask. 0: Comparison match interrupt not mask 1: Comparison match interrupt mask	R

20.3.12 Timer Mask Set Register (TMSR)

The TMSR is a 32-bit write-only register. It contains the comparison match flag set bits for all the channels.

	TMSR																															BASE+0x034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOEMST5	FIFOEMST4	FIFOEMST3	FIFOEMST0	HMST 7	HMST 6	HMST 5	HMST 4	HMST 3	HMST 2	HMST 1	HMST 0	OSTMST	Reserved			FIFOMST5	FIFOMST4	FIFOMST3	FIFOMST0	FMST 7	FMST 6	FMST 5	FMST 4	FMST 3	FMST 2	FMST 1	FMST 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

27:24	FIFOEMST 3~0	Set FIFOEMST n bit of TMR. 0: Ignore 1: Set FIFOEMASK n bit to 1	W
23:16	HMST 7~0	Set HMASK n bit of TMR. 0: Ignore 1: Set HMASK n bit to 1	W
15	OSTMST	Set OSTMASK n bit of TMR. 0: Ignore 1: Set OSTMASK n bit to 1	W
14:12	Reserved	Writing has no effect, read as zero.	R
11:8	FIFOMST 3~0	Set FIFOMST n bit of TMR. 0: Ignore 1: Set FMASK n bit to 1	W
7:0	FMST 7~0	Set FMASK n bit of TMR. 0: Ignore 1: Set FMASK n bit to 1	W

20.3.13 Timer Mask Clear Register (TMCR)

The TMCR is a 32-bit write-only register. It contains the comparison match flag clear bits for all the channels.

	TMCR																															BASE+0x038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				FIFOEMCL5	FIFOEMCL4	FIFOEMCL3	FIFOEMCL0	HMCL 7	HMCL 6	HMCL 5	HMCL 4	HMCL 3	HMCL 2	HMCL 1	HMCL 0	OSTMCL	Reserved			FIFOMCL5	FIFOMCL4	FIFOMCL3	FIFOMCL0	FMCL 7	FMCL 6	FMCL 5	FMCL 4	FMCL 3	FMCL 2	FMCL 1	FMCL 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:28	Reserved	Writing has no effect, read as zero.	R
27:24	FIFOEMCL 3~0	Set FIFOEMST n bit of TMR. 0: Ignore 1: Set FIFOEMASK n bit to 0	
23:16	HMCL 7~0	Set HMASK n bit of TMR. 0: Ignore 1: Set HMASK n bit to 0	W
15	OSTMCL	Set OSTMASK n bit of TMR. 0: Ignore 1: Set OSTMASK n bit to 0	W
14:12	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11:8	FIFOMCL 3~0	Set FIFOMCL n bit of TMR. 0: Ignore 1: Set FIFOMCL n bit to 0	W
7:0	FMCL 7~0	Set FMASK n bit of TMR. 0: Ignore 1: Set FMASK n bit to 0	W

20.3.14 Timer Stop Register (TSR)

The TSR is a 32-bit read-only register. It contains the timer stop control bits for each channel, WDT and OST. It is initialized to 0x00000000 by any reset. It can only be set by register TSSR and TSCR. If set, clock supplies to timer n / WDT / OST is stopped, and registers of the timer / WDT / OST cannot be accessed also.

	TSR																															BASE+0x01C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															WDTS	OSTS	Reserved							STOP 7	STOP 6	STOP 5	STOP 4	STOP 3	STOP 2	STOP 1	STOP 0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:17	Reserved	Writing has no effect, read as zero.	R
16	WDTS	0: The clock supplies to WDT is supplied 1: The clock supplies to WDT is stopped	R
15	OSTS	0: The clock supplies to OST is supplied 1: The clock supplies to OST is stopped	R
14:8	Reserved	Writing has no effect, read as zero.	R
7	STOP 7	0: The clock supplies to timer 7 is supplied 1: The clock supplies to timer 7 is stopped	R
6	STOP 6	0: The clock supplies to timer 6 is supplied 1: The clock supplies to timer 6 is stopped	R
5	STOP 5	0: The clock supplies to timer 5 is supplied 1: The clock supplies to timer 5 is stopped	R
4	STOP 4	0: The clock supplies to timer 4 is supplied 1: The clock supplies to timer 4 is stopped	R
3	STOP 3	0: The clock supplies to timer 3 is supplied 1: The clock supplies to timer 3 is stopped	R
2	STOP 2	0: The clock supplies to timer 2 is supplied 1: The clock supplies to timer 2 is stopped	R
1	STOP 1	0: The clock supplies to timer 1 is supplied 1: The clock supplies to timer 1 is stopped	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	STOP 0	0: The clock supplies to timer 0 is supplied 1: The clock supplies to timer 0 is stopped	R

20.3.15 Timer Stop Set Register (TSSR)

The TCSR is an 32-bit write-only register. It contains the timer stop set bits for each channel, WDT and OST. Since the timer stop control set bits are located in the same addresses, two or more timers can be started at the same time.

	TSSR																															BASE+0x02C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															WDTSS	OSTSS	Reserved							STPS 7	STPS 6	STPS 5	STPS 4	STPS 3	STPS 2	STPS 1	STPS 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:17	Reserved	Writing has no effect, read as zero.	R
16	WDTSS	Set WDTS bit of TSR. 0: Ignore 1: Set WDTS bit to 1	W
15	OSTSS	Set OSTS bit of TSR. 0: Ignore 1: Set OSTS bit to 1	W
14:8	Reserved	Writing has no effect, read as zero.	R
7	STPS 7	Set STOP 7 bit of TSR. 0: Ignore 1: Set STOP 7 bit to 1	W
6	STPS 6	Set STOP 6 bit of TSR. 0: Ignore 1: Set STOP 6 bit to 1	W
5	STPS 5	Set STOP 5 bit of TSR. 0: Ignore 1: Set STOP 5 bit to 1	W
4	STPS 4	Set STOP 4 bit of TSR. 1: Set STOP 4 bit to 1 0: Ignore	W
3	STPS 3	Set STOP 3 bit of TSR. 0: Ignore 1: Set STOP 3 bit to 1	W
2	STPS 2	Set STOP 2 bit of TSR. 0: Ignore	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Set STOP 2 bit to 1	
1	STPS 1	Set STOP 1 bit of SR. 0: Ignore 1: Set STOP 1 bit to 1	W
0	STPS 0	Set STOP 0 bit of TSR. 0: Ignore 1: Set STOP 0 bit to 1	W

20.3.16 Timer Stop Clear Register (TSCR)

The TSCR is an 32-bit write-only register. It contains the timer stop clear bits for each channel, WDT and OST. Since the timer stop clear bits are located in the same addresses, two or more timers can be stop at the same time.

	TSCR																															BASE+0x03C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															WDTSC	OSTSC	Reserved							STPC 7	STPC 6	STPC 5	STPC 4	STPC 3	STPC 2	STPC 1	STPC 0
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:17	Reserved	Writing has no effect, read as zero.	R
16	WDTSC	Set WDTS bit of TSR. 0: Ignore 1: Set WDTS bit to 0	W
15	OSTSC	Set OSTS bit of TSR. 0: Ignore 1: Set OSTS bit to 0	W
14:8	Reserved	Writing has no effect, read as zero.	R
7	STPC 7	Set STOP 7 bit of TSR. 0: Ignore 1: Set STOP 7 bit to 0	W
6	STPC 6	Set STOP 6 bit of TSR. 0: Ignore 1: Set STOP 6 bit to 0	W
5	STPC 5	Set STOP 5 bit of TSR. 0: Ignore 1: Set STOP 5 bit to 0	W
4	STPC 4	Set STOP 4 bit of TSR. 0: Ignore 1: Set STOP 4 bit to 0	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3	STPC 3	Set STOP 3 bit of TSR. 0: Ignore 1: Set STOP 3 bit to 0	W
2	STPC 2	Set STOP 2 bit of TSR. 0: Ignore 1: Set STOP 2 bit to 0	W
1	STPC 1	Set STOP 1 bit of TSR. 0: Ignore 1: Set STOP 1 bit to 0	W
0	STPC 0	Set STOP 0 bit of TSR. 0: Ignore 1: Set STOP 0 bit to 0	W

20.3.17 Timer Status Register (TSTR)

The TSTR is a 32-bit read-only register. It contains the status of channel in TCU2 mode. The register can be written by setting register TSTSR and TSTCR.

	TSR																															BASE+0x0F0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													REAL2	REAL1	Reserved														BUSY2	BUSY1	Reserved
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:19	Reserved	Writing has no effect, read as zero.	R
18	REAL 2	0: The value read from counter 2 is a false value 1: The value read from counter 2 is a real value	R
17	REAL1	0: The value read from counter 1 is a false value 1: The value read from counter 1 is a real value	R
16:3	Reserved	Writing has no effect, read as zero.	R
2	BUSY 2	0: The counter 2 is ready now 1: The counter 2 is busy now	R
1	BUSY1	0: The counter 1 is ready now 1: The counter 1 is busy now	R
0	Reserved	Writing has no effect, read as zero.	R

20.3.18 Timer Status Set Register (TSTSR)

The TSTSR is a 32-bit write-only register. It contains the timer status set bits for each channel.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TSTSR																															BASE+0x0F4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													REALS 2	REALS 1	Reserved														BUSYS 2	BUSYS 1	Reserved
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:19	Reserved	Writing has no effect, read as zero.	R
18	REALS 2	Set REAL 2 bit of TSTR. 0: Ignore 1: Set REAL 2 bit to 1	R
17	REALS 1	Set REAL 1 bit of TSTR. 0: Ignore 1: Set REAL 1 bit to 1	R
16:3	Reserved	Writing has no effect, read as zero.	R
2	BUSYS 2	Set BUSY 2 bit of TSTR. 0: Ignore 1: Set BUSY 2 bit to 1	R
1	BUSYS 1	Set BUSY 1 bit of TSTR. 0: Ignore 1: Set BUSY 1 bit to 1	R
0	Reserved	Writing has no effect, read as zero.	R

20.3.19 Timer Status Clear Register (TSTCR)

The TSTCR is a 32-bit write-only register. It contains the timer status clear bits for each channel.

	TSTCR																															BASE+0x0F8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved													REALC 2	REALC 1	Reserved														BUSYC 2	BUSYC 1	Reserved
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:19	Reserved	Writing has no effect, read as zero.	R
18	REALC 2	Clear REAL 2 bit of TSTR. 0: Ignore 1: Clear REAL 2 bit to 1	R
17	REALC 1	Clear REAL 1 bit of TSTR. 0: Ignore 1: Clear REAL 1 bit to 1	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16:3	Reserved	Writing has no effect, read as zero.	R
2	BUSYC 2	Clear BUSY 2 bit of TSTR. 0: Ignore 1: Clear BUSY 2 bit to 1	R
1	BUSYC 1	Clear BUSY 1 bit of TSTR. 0: Ignore 1: Clear BUSY 1 bit to 1	R
0	Reserved	Writing has no effect, read as zero.	R

20.3.20 Timer control mode Register (TCUMOD)

The TCUMOD is a 32-bit read write register. It contains the fifo control signal when TCU work in the TCU fifo mode , and only for the TCU0 、TCU3、TCU4、TCU5.

	TCUMOD0 ,TCUMOD3, TCUMOD4,TCUMOD5																**BASE+0x100, 0x110, 0x120, 0x130**															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																cyc_num						cyc_st	fifo_num					fifo_num_st	al_empty	fifo_clr	FIFOMOD
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:10	**cyc_num**	Set the fifo cycle numbers. when the TCU work in FIFO mode 2.	RW
9	cyc_st	Select the FIFO mode. 0: work in FIFO mode 1 1: work in FIFO mode 2	RW
8:4	fifo_num	Set the fifo data numbers can read out of the fifo. when the TCU work in FIFO mode 2.	RW
3	fifo_num_st	Select the FIFO mode. 0: work in FIFO mode 1 1: work in FIFO mode 2	RW
2	al_empty	please keep this bit value is 0.	RW
1	fifo_clr	0: do nothing 1:clr the fifo	
0	FIFOMOD	Set the TCU work in fifo mode. 0: TCU work in nonfifo mode 1: TCU work in fifo mode	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.3.21 Timer fifo write data (TFWD)

The comparison data FIFO registers TFWD is used to store the data to be compared with the content of the up-counter TCNT. This register can be directly read and written(Default: 0),but only can read the 16 bit data at once. And Only use in the fifo mode.

	TFWD0 , TFWD3, TFWD4, TFWD5																BASE+0x104, 0x114, 0x124, 0x134															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFWD_HIGN																TFWD_LOW															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

20.3.22 Timer fifo state Register (TFIFOSR)

The TFIFOSR is a 32-bit write-only register. It contains the timer fifo status bits for TCU work in fifo mode.

	TFIFOSR0 , TFIFOSR3, TFIFOSR4, TFIFOSR5																BASE+0x108, 0x118, 0x128, 0x138															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserve																				cyc_timer						fifo_entry					fifo_w_flag
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:12	Reserved	Writing has no effect, read as zero.	R
11: 6	**cyc_timer**	record the number of cycs had been done.(valid in fifo)	R
5 : 1	**fifo_entry**	record fifo's read address pointer	R
0	**fifo_w_flag**	record the fifo's state	R

20.4 Operation

20.4.1 Basic Operation in TCU1 Mode

The value of TDFR should be bigger than TDHR, and the minimum settings are TDHR = 0 and TDFR = 1. In this case, the timer output clock cycle is the input clock × 1/2. If TDHR > TDFR, no comparison TFHR signal is generated.

Before the timer counter begin to count up, we need to do as follows:

If you want to use PWM you should set TCSR.PWM_EN to be 0 before you initial TCU.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1 Initial the configuration.
   a Writing TCSR.INITL to initialize PWM output level.
   b Writing TCSR.SD to setting the shutdown mode (Abrupt shutdown or Graceful shutdown).
   c Writing TCSR.PRESCALE to set TCNT count clock frequency.
   d Setting TCNT, TDHR and TDFR.

2 Enable the clock.
   a Writing TCSR.PWM_EN to set whether enable PWM or disable PWM.
   b Writing TCSR.EXT_EN, TCSR.RTC_EN or TCSR.PCK_EN to 1 to select the input clock and enable the input clock. Only one of TCSR.EXT_EN, TCSR.RTC_EN and TCSR.PCK_EN can be set to 1.

After initialize the register of timer, we should start the counter as follows:

3 Enable the counter.
   Setting the TESR.TCST bit to 1 to enable the TCNT.

NOTE: The input clock and PCLK should follow the rules advanced before.

20.4.2 Disable and Shutdown Operation in TCU1 Mode

1 Setting the TECR.TCCL bit to 1 to disable the TCNT.

20.4.3 Basic Operation in TCU2 Mode

The value of TDFR should be bigger than TDHR, and the minimum settings are TDHR = 0 and TDFR = 1. In this case, the timer output clock cycle is the input clock × 1/2. If TDHR > TDFR, no comparison TFHR signal is generated.

Initial state is that TCSR.PRESCALE=0, TCSR.PWM_EN=0 and TCENR=0.

1 Reset the TCU.
   a Writing TCSR.PCK_EN to 1 to select pclk as the input clock.
   b Set TCSR.CLRZ to 1 to clear TCNT or set TCNT to an initial value.
   c Writing TCSR.PCK_EN to 0 to close the input clock.

2 Initial the configuration.
   a Setting TDHR and TDFR.
   b Writing TCSR.INITL to initialize PWM output level (if used PWM).
   c Writing TCSR.PRESCALE to set TCNT count clock frequency.
   d Writing TCSR.EXT_EN, TCSR.RTC_EN or TCSR.PCK_EN to 1 to select the input clock and enable the input clock. Only one of TCSR.EXT_EN, TCSR.RTC_EN and TCSR.PCK_EN can be set to 1.
   e Writing TCSR.PWM_EN to set whether enable PWM or disable PWM.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

After initialize the register of timer, we should start the counter as follows:

3 Setting the TESR.TCST bit to 1 to enable the TCNT.

NOTES:

1 You can clear the counter when counter is working.
   a Set TCSR.CLRZ to 1 to clear TCNT.
   b Wait till TSTR.BUSY = 0, that is the counter have been cleared.
2 You can enable PWM or disable PWM the counter when counter is working.
   a Set TCSR.PWM_EN to 1 to enable PWM.
   b Set TCSR.PWM_EN to 0 to disable PWM.

20.4.4 Disable and Shutdown Operation in TCU2 Mode

1 Writing TCSR.PWM_EN to 0 to disable PWM.
2 Setting the TECR.TCCL bit to 1 to disable the TCNT.
3 Wait till TSTR.BUSY = 0, that is the reset of counter is finished.

20.4.5 Read Counter in TCU2 Mode

If you want to read the data from register TCNT when the TCU is working, you can check TSTR.REAL whether it is good or not. It is suggested that:

1 If TSTR.REAL==1, the data read is available.
2 If TSTR.REAL==0, reread the counter till TSTR.REAL==1, the data read is available.
3 If TSTR.REAL is always 0, you can read some data, and lose some data that is quick different from the others. Then choose a data from them as the available data.

NOTES:

1 It suggested that (1), (2) is often used when the counter clock is very slow.
2 It suggested that (3) is often used when the counter clock is very fast.

20.4.6 Pulse Width Modulator (PWM)

Timer 0~7 can be used as Pulse Width Modulator (PWM). The PWM can be used to control the back light inverter or adjust bright or contrast of LCD panel.

FULL comparison match signal and HALF comparison match signal can determine an attribute of the PWM_OUT waveform. FULL comparison match signal specifies the clock cycle for the PWM module clock. HALF comparison match signal specifies the duty ratio for the PWM module clock.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.4.7 Trackball Input Waveform Detect

Timer 0, 3~7 can be used as a waveform edge counter to count both positive edge and negative edge of an external input waveform. For example, a trackball device's input. 4 timers will need to count all four directions (up, down, left, right). You need configure relate GPIO (set relate 4 PWM IO as input) and set relate TCSR.PWM_IN_EN to 1. Both relate TDFR and TDHR need to set to 0xFFFF, unless you need a special interrupt when the counter hit TDFR or TDHR. The counter will clear to 0 when hit TDFR.

Before the timer counter begin to count up, we need to do as follows:

1 Initial the configuration.
   - a Writing TCSR.SD to setting the shutdown mode (Abrupt shutdown or Graceful shutdown).
   - b Writing TCSR.PRESCALE to set to 0.
   - c Setting TCNT, TDHR and TDFR.

2 Enable the clock.
   - a Writing TCSR.PWM_EN to disable PWM.
   - b Writing TCSR.EXT_EN, TCSR.RTC_EN and TCSR.PCK_EN to 0, TCSR.PWM_IN_EN to 1 to select the input clock and enable the input clock.

After initialize the register of timer, we should start the counter as follows:

3 Enable the counter.
   Setting the TESR.TCST bit to 1 to enable the TCNT.

NOTE: The input clock and PCLK should follow the rules advanced before.

20.4.8 Basic Operation in FIFO Mode 1

1. set TCSR, please keep PWM_EN to 0.
2. set TCUMOD to 0x0000_0001.
3. push data into fifo, through write data in TFWD. fifo's depth is 16, do not over flow.
   do not push data like 0x0000_XXXX, or 0xXXXX_0000 X: nonzero value
4. set the TCSR.PWM_EN to 1.
5. set the TESR to enable the TCNT.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6. push some data into fifo is optional. Please insure the TFIFOSR. **fifo_w_flag is 0, before push data into fifo. and do not over flow.** do not push data like 0x0000_XXXX, or 0xXXXX_0000
7. when TFR.fifo_empty_flag is 1, symbol the fifo is empty..
8. when fifo match finished ,if you want to do it again, please stop the TCNT count.
9. set TCSR,please keep PWM_EN to 0.
10. set TFCR to clr the flag.
11. set TCUMOD.fifo_clr to 1.
12. set TCUMOD.fifo_clr to 0.
13. do step2 ~step7

example

```
1.  write_reg(0x04C,0x00000002);
2.  write_reg(0x100,0x00000001);
3.  write_reg(0x104,0x00010001);
     write_reg(0x104,0x00020002);
     write_reg(0x104,0x00030003);        //write fifo
4.  apb_write_reg(0x04C,0x00000082);
5.  apb_write_reg(0x014,0x00000001);
6.  if((read_reg(0x108) & 0x1) != 1)
    {
     write_reg(0x104,0x00040004);     //write fifo
    }
```

20.4.9 Basic Operation in FIFO Mode 2

1. set the TCSR, Please keep PWM_EN to 0.
2. set TCUMOD to 0x0000_0001.
3. push data into fifo, through write data in TFWD. fifo's depth is 16, do not over flow.
   do not push data like 0x0000_XXXX, or 0xXXXX_0000
4. set TCUMOD.
5. set the TCSR.PWM_EN to 1.
6. set the TESR to enable the TCNT.
7. if step 4 has been done, do not push data into fifo,
8. when TFIFOSR. **cyc_timer ==** TCUMOD. **cyc_num , It symbol finished.**
9. if you want do again, please stop the TCNT count
10. set TCUMOD.fifo_clr to 1.
11. set TCUMOD.fifo_clr to 0.
12. do step1 ~ step 8

example

```
1.  write_reg(0x04C,0x00000002);
2.  write_reg(0x100,0x00000001);
3.  write_reg(0x104,0x00010001);
     write_reg(0x104,0x00020002);
     write_reg(0x104,0x00030003);
     write_reg(0x104,0x00040004);
```


Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

```
    write_reg(0x104,0x00050005);
4.  write_reg(0x100,0x00002A59);
      // conduct ten cycles, each cycle would match five data from fifo
5.  write_reg(0x04C,0x00000082);
6.  write_reg(0x014,0x00000001);
7.  if(TFIFOSR. cyc_timer == TCUMOD. cyc_num)
    {
        finish
    }
```


Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

21 Operating System Timer

21.1 Overview

The OST (Operating System Timer) contains one 64-bit programmable timer. It can be used as operating system timer.

OST has the following features:

- OST includes:
  - 64-bit Counter
  - 32-bit Compare Data Register
  - Control Register
- Independent clock for each counter, selectable by software
  - PCLK, EXTAL and RTCCLK can be used as the clock for counter
  - The division ratio of the clock can be set to 1, 4, 16, 64, 256 and 1024 by software
- Match interrupt can be generated for OST using the compare data registers
  - Interrupt flag and interrupt mask is same with TCU in TCU spec

21.2 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute

   R - Read only

   W - Write only

   RW - read and write

   RCW - read and write, but clear to 0 by read

   RSW - read and write, but set to 1 by read

   RWC - read and write, clear to 0 by write 1, write 0 has no effect

   RWS - read and write, set to 1 by write 1, write 0 has no effect

   RC - read only, and clear to 0 by read

   RS - read only, and set to 1 by read

   SPEC - special access method, relate to its description
3. Reset Value

   1 - reset to 1

   0 - reset to 0

   ? - value unknown after reset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 21-1 Registers Memory Map-Address Base

Name	Base	Description
OST	0x10002000	Address base of OST

Table 21-2 OST Registers Configuration

Name	Description	RW	Reset Value	Offset Address	Access Size
OSTDR	Operating System Timer Data Register	RW	0x????????	0x0e0	32
OSTCNTL	Operating System Timer Counter Lower 32 Bits	RW	0x????????	0x0e4	32
OSTCNTH	Operating System Timer Counter Higher 32 Bits	RW	0x????????	0x0e8	32
OSTCSR	Operating System Timer Control Register	RW	0x0000	0x0ec	16
OSTCNTH BUF	Operating System Timer Counter Higher 32 Bits Buffer	R	0x????????	0x0fc	32

21.2.1 Operating System Control Register (OSTCSR)

The TCSR is a 16-bit read/write register. It contains the control bits for OST. It is initialized to 0x00 by any reset.

	OSTCSR																BASE+0x0ec															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	CNT_MD	Reserved					SD	Reserved			PRESCALE			EXT_EN	RTC_EN	PCK_EN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	CNT_MD	Counter mode choose bit. 0: When the value counter is equal to compare value, the counter will be cleared, and increase from 0. 1: When the value counter is equal to compare value, the counter will go on increasing till overflow, and then increase from 0.	
14:6	Reserved	Writing has no effect, read as zero.	R
9	SD	Shut Down (SD) the counter of OST. 0: Graceful shutdown (only used when CNT_MD = 0) 1: Abrupt shutdown	RW
5:3	PRESCALE	These bits select the TCNT count clock frequency.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Bit 2	Bit1	Bit 0	Description	
		0	0	0	Internal clock: CLK/1	
		0	0	1	Internal clock: CLK/4	
		0	1	0	Internal clock: CLK/16	
		0	1	1	Internal clock: CLK/64	
		1	0	0	Internal clock: CLK/256	
		1	0	1	Internal clock: CLK/1024	
		110~111			Reserved	
2	EXT_EN	Select EXTAL as the timer clock input. 0: Disable 1: Enable				RW
1	RTC_EN	Select RTCCLK as the timer clock input. 1: Enable 0: Disable				RW
0	PCK_EN	Select PCLK as the timer clock input. 1: Enable 0: Disable				RW

NOTE: The input clock of timer and the PCLK should keep to the rules as follows.

Input clock of timer: IN_CLK	**Clock generated from the frequency divider (PRESCALE): DIV_CLK**
PCK_EN == 0, RTC_EN == 1 and EXT_EN == 0 (IN_CLK = RTCCLK)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 0, RTC_EN == 0 and EXT_EN == 1 (IN_CLK = EXTAL)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 1, RTC_EN == 0 and EXT_EN == 0 (IN_CLK = PCLK)	ANY

21.2.2 Operating System Timer Data Register (OSTDR)

The operating system timer data register OSTDR is used to store the data to be compared with the content of the operating system timer up-counter OSTCNT. This register can be directly read and written. Please also refer to CNT_MD bit of register **OSTCSR** to understand the counter behavior. (Default: indeterminate)

	OSTDR																															**BASE+0x0e0**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTDR																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.2.3 Operating System Timer Counter (OSTCNTH, OSTCNTL)

The operating system timer counter (OSTCNT) is a 64-bit read/write counter. The up-counter OSTCNT can be set by software and counts up using the prescaler output clock. The data can be read out at any time. The counter data can be written at any time. (Default: indeterminate)

	OSTCNTH																															BASE+0x0e8
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTCNTH																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

	OSTCNTL																															BASE+0x0e4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTCNTL																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

21.2.4 Operating System Timer Counter high 32 bits buffer (OSTCNTHBUF)

The operating system timer counter high 32 bits buffer OSTCNTHBUF is used to store the high 32 bits of OSTCNT when its lower 32 bits are read by software. This register can be directly read. (Default: indeterminate)

	OSTCNTHBUF																															BASE+0x0fc
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTCNTHBUF																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

21.3 Operation

21.3.1 Basic Operation

Before the timer counter begin to count up, we need to do as follows:

1 Initial the configuration.
   - a Writing TCSR.SD to setting the shutdown mode (Abrupt shutdown or Graceful shutdown).
   - b Writing OSTCSR.PRESCALE to set OSTCNT count clock frequency.
   - c Setting OSTCNTL/H and OSTDR.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 Enable the clock.

Writing OSTCSR.EXT_EN, OSTCSR.RTC_EN or OSTCSR.PCK_EN to 1 to select the input clock and enable the input clock. Only one of OSTCSR.EXT_EN, OSTCSR.RTC_EN and OSTCSR.PCK_EN can be set to 1.

After initialize the register of timer, we should start the counter as follows:

3 Enable the counter.

Setting the TESR.OSTCST bit to 1 to enable the OSTCNT.

NOTE: The input clock and PCLK should follow the rules advanced before.

21.3.2 Disable and Shutdown Operation

1 Setting the TECR.OSTCCL bit to 1 to disable the OSTCNT.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

22 Interrupt Controller

22.1 Overview

This chapter describes the interrupt controller included in the XBurst Processor, explains its modes of operation, and defines its registers. The interrupt controller controls the interrupt sources available to the processor and contains the location of the interrupt source to allow software to determine source of all interrupts. It also determines whether an IRQ occurs or not and masks the interrupts.

Features:

- Total 64 interrupt sources
- Each interrupt source can be independently enabled
- Priority mechanism to indicate highest priority interrupt
- All the registers are accessed by CPU
- Unmasked interrupts can wake up the chip in sleep mode
- Another set of source, mask and pending registers to serve for PDMA

22.2 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

22.2.1 Registers Mapping

Name	Base	Description
INTC	0x10001000	Address base of INTC

The following table lists the registers of Interrupt Controller. All of these registers are 32bit, and each bit of the register represents or controls one interrupt source that list in following table.

Table 22-1 INTC Register

Name	Description	RW	Reset Value	Address Offset	Access Size
ICSR0	IRQ Source Register0	R	0x00000000	0x0000	32
ICMR0	IRQ Mask Register0	RW	0xFFFFFFFF	0x0004	32
ICMSR0	IRQ Mask Set Register0	W	0x????????	0x0008	32
ICMCR0	IRQ Mask Clear Register0	W	0x????????	0x000C	32
ICPR0	IRQ Pending Register0	R	0x00000000	0x0010	32
ICSR1	IRQ Source Register1	R	0x00000000	0x0020	32
ICMR1	IRQ Mask Register1	RW	0xFFFFFFFF	0x0024	32
ICMSR1	IRQ Mask Set Register1	W	0x????????	0x0028	32
ICMCR1	IRQ Mask Clear Register1	W	0x????????	0x002C	32
ICPR1	IRQ Pending Register1	R	0x00000000	0x0030	32
DSR0	IRQ Source Register0 for PDMA	R	0x00000000	0x0034	32
DMR0	IRQ Mask Register0 for PDMA	RW	0xFFFFFFFF	0x0038	32
DPR0	IRQ Pending Register0 for PDMA	R	0x00000000	0x003C	32
DSR1	IRQ Source Register1 for PDMA	R	0x00000000	0x0040	32
DMR1	IRQ Mask Register1 for PDMA	RW	0xFFFFFFFF	0x0044	32
DPR1	IRQ Pending Register1 for PDMA	R	0x00000000	0x0048	32

22.2.2 Interrupt Controller Source Register0 (ICSR0)

This register contains all the interrupts' status. A "1" indicates that the corresponding interrupt is pending . A "0" indicates that the interrupt is not pending now. The register is read only.

	ICSR0																															BASE + 0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Value of Bits in ICSR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

22.2.3 Interrupt Controller Source Register1 (ICSR1)

This register contains all the interrupts' status. A "1" indicates that the corresponding interrupt is pending . A "0" indicates that the interrupt is not pending now. The register is read only.

	ICSR1																															BASE + 0x0020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICSR1	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

Note: ETHC is only for fpga.

22.2.4 Interrupt Controller Mask Register0 (ICMR0)

This register is used to mask the interrupt input sources and defines which active sources are allowed to generate interrupt requests to the processor. Its value can be changed either by writing ICMSR and ICMCR or by writing itself. The masked interrupts are invisible to the processor.

	ICMR0																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICMR0	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

22.2.5 Interrupt Controller Mask Register1 (ICMR1)

This register is used to mask the interrupt input sources and defines which active sources are

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

allowed to generate interrupt requests to the processor. Its value can be changed either by writing ICMSR and ICMCR or by writing itself. The masked interrupts are invisible to the processor.

	ICMR1																															BASE + 0x0024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICMR1	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

Note: ETHC is only for fpga.

22.2.6 Interrupt Controller Mask Set Register0 (ICMSR0)

This register is used to set bits in the interrupt mask register. This register is write only.

	ICMSR0																															BASE + 0x0008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICMSR0	Description
0	Ignore.
1	Will set the corresponding interrupt mask bit.

22.2.7 Interrupt Controller Mask Set Register1 (ICMSR1)

This register is used to set bits in the interrupt mask register. This register is write only.

	ICMSR1																															BASE + 0x0028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Value of Bits in ICMSR1	Description
0	Ignore.
1	Will set the corresponding interrupt mask bit.

Note: ETHC is only for fpga.

22.2.8 Interrupt Controller Mask Clear Register0 (ICMCR0)

This register is used to clear bits in the interrupt mask register. This register is write only.

	ICMCR0																															BASE + 0x000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICMCR0	Description
0	Ignore.
1	Will clear the corresponding interrupt mask bit.

22.2.9 Interrupt Controller Mask Clear Register1 (ICMCR1)

This register is used to clear bits in the interrupt mask register. This register is write only.

	ICMCR1																															BASE + 0x002C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICMCR1	Description
0	Ignore.
1	Will clear the corresponding interrupt mask bit.

Note: ETHC is only for fpga.

22.2.10 Interrupt Controller Pending Register0 (ICPR0)

This register contains the status of the interrupt sources after masking. This register is read only.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ICPR0																															BASE + 0x0010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICPR0	Description
0	The corresponding interrupt is not active or is masked.
1	The corresponding interrupt is active and is not masked to the processor.

22.2.11 Interrupt Controller Pending Register1 (ICPR1)

This register contains the status of the interrupt sources after masking. This register is read only.

	ICPR1																															BASE + 0x0030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICPR1	Description
0	The corresponding interrupt is not active or is masked.
1	The corresponding interrupt is active and is not masked to the processor.

Note: ETHC is only for fpga.

22.2.12 Interrupt Source Register0 for PDMA (DSR0)

This register contains status of all interrupts. A “1” indicates the corresponding interrupt is pending. The register is read only.

	DSR0																															BASE + 0x0034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Value of Bits in DSR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

22.2.13 Interrupt Mask Register0 for PDMA (DMR0)

This register is used to mask the interrupt input sources and defines which active sources are allowed to generate interrupt requests to the processor. The masked interrupts are invisible to the processor.

	DMR0																															BASE + 0x0038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in DMR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

22.2.14 Interrupt Pending Register0 for PDMA (DPR0)

This register contains the status of the interrupt sources after masking. This register is read only.

	DPR0																															BASE + 0x003C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCD	DELAY_LINE	ISP	Reserved	TCU0	TCU1	TCU2	Reserved	AES	HASH	OTG	EHCI	EPDC	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	Reserved	PDMA	Reserved	SSI0	SSI1	IPU	OHCI	CSI	DSI	BCH	AIC0	DMIC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in ICPR0	Description
0	The corresponding interrupt is not active or is masked.
1	The corresponding interrupt is active and is not masked to the processor.

22.2.15 Interrupt Source Register1 to PDMA (DSR1)

This register contains status of all interrupts. A “1” indicates the corresponding interrupt is pending. The register is read only.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DSR1																															BASE + 0x0040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in DSR1	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

Note: ETHC is only for fpga.

22.2.16 Interrupt Mask Register1 for PDMA (DMR1)

This register is used to mask the interrupt input sources and defines which active sources are allowed to generate interrupt requests to the processor. The masked interrupts are invisible to the processor.

	DMR1																															BASE + 0x0044
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Value of Bits in DMR1	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

Note: ETHC is only for fpga.

22.2.17 Interrupt Pending Register1 for PDMA (DPR1)

This register contains the status of the interrupt sources after masking. This register is read only.

	DPR1																															BASE + 0x0048
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPU	VPU	PDMAM	I2C0	I2C1	I2C2	I2C3	Reserved	ETHC	EFUSE	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved			PCM0	NFI	Reserved	MSC0	MSC1	MSC2	URAT4	Reserved	RTC
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Value of Bits in ICPR0	Description
0	The corresponding interrupt is not active or is masked.
1	The corresponding interrupt is active and is not masked to the processor.

Note: ETHC is only for fpga.

22.3 Software Considerations

The interrupt controller is reflecting the status of interrupts sources in the peripheral.

Software should perform the task - determine the interrupt source from in ICPRx. In this chip, pending interrupts have two levels in structure. Interrupting module in the system that contains more than one interrupt sources need software to determine how to service it by reading interrupt status registers within it.

In the interrupt handler, the serviced interrupt source needs to be cleared in the interrupting device. In order to make certain the cleared source request status has been reflected at the corresponding ICPRx bit, software should wait enough time before exiting interrupt state.

The procedure is described following:

1. Interrupt generated.
2. CPU query interrupt sources, saves the current environment and then goes to interrupt common service routine.
3. Get ICPRx.
4. Find the highest priority interrupt and vector it. (The software decides which one has the highest priority)
5. Mask the chosen interrupt by writing the register ICMSRx.
6. Enable the system interrupt to allow the interrupt nesting. (software decided)
7. Execute the interrupt handler and unmask it by writing the register ICMCRx when exit the handler.
8. CPU restores the saved environment and exits the interrupt state.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

23 Watchdog Timer

23.1 Overview

The watchdog timer is used to resume the processor whenever it is disturbed by malfunctions such as noise and system errors. The watchdog timer can generate the reset signal.

Features:

- Generates WDT reset
- A 16-bit Data register and a 16-bit counter
- Counter clock uses the input clock selected by software
  - PCLK, EXTAL and RTCCLK can be used as the clock for counter
  - The division ratio of the clock can be set to 1, 4, 16, 64, 256 and 1024 by software

23.2 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 23-1 Registers Memory Map-Address Base

Name	Base	Description
WDT	0x10002000	Address base of WDT

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 23-2 WDT Registers Configuration

Name	Description	RW	Reset Value	Address offset	Access Size
TDR	Watchdog Timer Data Register	RW	0x????	0x000	16
TCER	Watchdog Counter Enable Register	RW	0x00	0x004	8
TCNT	Watchdog Timer Counter	RW	0x????	0x008	16
TCSR	Watchdog Timer Control Register	RW	0x0000	0x00C	16

23.2.1 Watchdog Control Register (TCSR)

The TCSR is a 16-bit read/write register. It contains the control bits for WDT. It is initialized to 0x00 by any reset.

	TCSR																															BASE+0x 00C
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved										PRESCALE			EXT_EN	RTC_EN	PCK_EN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:6	Reserved	Writing has no effect, read as zero.	R
5:3	PRESCALE	These bits select the TCNT count clock frequency. Bit 2 / Bit1 / Bit 0 / Description 0 / 0 / 0 / Internal clock: CLK/1 0 / 0 / 1 / Internal clock: CLK/4 0 / 1 / 0 / Internal clock: CLK/16 0 / 1 / 1 / Internal clock: CLK/64 1 / 0 / 0 / Internal clock: CLK/256 1 / 0 / 1 / Internal clock: CLK/1024 110~111 / Reserved	RW
2	EXT_EN	Select EXTAL as the timer clock input. 1: Enable 0: Disable	RW
1	RTC_EN	Select RTCCLK as the timer clock input. 1: Enable 0: Disable	RW
0	PCK_EN	Select PCLK as the timer clock input. 1: Enable 0: Disable	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: The input clock of timer and the PCLK should keep to the rules as follows:

Input clock of timer: IN_CLK	**Clock generated from the frequency divider (PRESCALE): DIV_CLK**
PCK_EN == 0, RTC_EN == 1 and EXT_EN == 0 (IN_CLK = RTCCLK)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 0, RTC_EN == 0 and EXT_EN == 1 (IN_CLK = EXTAL)	$f_{DIV_CLK} < ½ f_{PCLK}$
PCK_EN == 1, RTC_EN == 0 and EXT_EN == 0 (IN_CLK = PCLK)	ANY

23.2.2 Watchdog Enable Register (TCER)

The TCER is an 8-bit read/write register. It contains the counter enable control bits for watchdog. It is initialized to 0x00 by any reset.

	TCER																															**BASE+0x 004**
Bit																										7	6	5	4	3	2	1	0
																										Reserved							TCEN
RST																										0	0	0	0	0	0	0	0

Bits	**Name**	**Description**	**RW**
7:1	Reserved	Writing has no effect, read as zero.	R
0	TCEN	Counter enable control. 0: Timer stop 1: Timer running	RW

23.2.3 Watchdog Timer Data Register (TDR)

The watchdog timer data register TDR is used to store the data to be compared with the content of the watchdog timer up-counter TCNT. This register can be directly read and written. (Default: indeterminate)

	TDR																															**BASE+0x 000**
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TDR															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.2.4 Watchdog Timer Counter (TCNT)

The watchdog timer counter (TCNT) is a 16-bit read/write counter. The up-counter TCNT can be reset to 0 by software and counts up using the prescaler output clock. When TCNT count up to equal to TDR, the comparison match signal will be generated and a WDT reset is generated. The data can be read out at any time. The counter data can be written at any time. (Default: indeterminate)

	TCNT																BASE+0x 008															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TCNT															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

23.3 Watchdog Timer Function

The following describes steps of using WDT:

1 Setting the PRESCALE of input clock in register TCSR.
2 Set register TDR and TCNT.
3 Select the input clock and enable the input clock in register TCSR.

After initialize the register of timer, we should start the counter as follows:

4 Set TCEN bit in TCER to 1. The counter TCNT begins to count.
5 If TCNT = TDR, a WDT reset will be generated.

NOTES:

1 The input clock and PCLK should follow the rules advanced before.
2 The clock of WDT can be stopped by setting register TSR, and register TSR can only be set by register TSSR or TSCR. The content of register TSR, TSSR and TSCR can be found in TCU spec.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24 PDMA Controller

24.1 Overview

Programmable DMA controller (PDMAC) is dedicated to smartly transfer data between on-chip peripherals (MSC, AIC, UART, etc.), external memories, and memory-mapped external devices.

24.2 Features

- Support up to 32 independent DMA channels
- Descriptor or No-Descriptor Transfer mode compatible with previous JZ SOC
- A simple Xburst-1 CPU supports smart transfer mode controlled by programmable firmware
- Transfer data units: 1-byte, 2-byte, 4-byte, 16-byte, 32-byte, 64-byte, 128-byte
- Transfer number of data unit: 1 ~ 2^{24} - 1
- Independent source and destination port width: 8-bit, 16-bit, 32-bit
- Fixed three priorities of channel groups: 0~3, highest; 4~11: mid; 12~31: lowest
- A dedicated bus interface - BIF interconnects with on-chip BCH
- A dedicated bus interface - NIF interconnects with on-chip NEMC or off-chip NEMC.
- An extra INTC IRQ can be bound to one programmable DMA channel
- Dedicated Security ROM and Security RAM supporting enhanced security requirements.

24.3 Block Diagram

Figure 24-1 Block Diagram of PDMA

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.4 Memory Mapped Register Descriptions

24.4.1 DMA Channel Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clears to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - Reset to 1
   0 - reset to 0
   ? - Value unknown after reset

Table 24-1 Registers Memory Map-Address Base

Name	Base	Description
Channel n	0x13420000 + n*0x20	PDMA channel n.

Table 24-2 DMA Channel Registers (n=0~31)

Name	Description	Reset Value	Address offset	Access Size (bit)
DSAn	Channel n Source Address	0x?	0x00 + n*0x20	32
DTAn	Channel n Target Address	0x?	0x04 + n*0x20	32
DTCn	Channel n Transfer Count	0x?	0x08 + n*0x20	32
DRTn	Channel n Request Source	0x?	0x0C + n*0x20	32
DCSn	Channel n Control/Status	0x?	0x10 + n*0x20	32
DCMn	Channel n Command	0x?	0x14 + n*0x20	32
DDAn	Channel n Descriptor Address	0x?	0x18 + n*0x20	32
DSDn	Channel n Stride Difference	0x?	0x1C + n*0x20	32

24.4.2 Global Control Registers

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 24-3 Registers Memory Map-Address Base

Name	Base	Description
GCR	0x13420000	PDMA global control registers.

Table 24-4 Global Control Registers

Name	Description	Reset Value	Address offset	Access Size (bit)
DMAC	DMA Control	0x0	0x1000	32
DIRQP	DMA Interrupt	0x0	0x1004	32
DDR	DMA Doorbell	0x0	0x1008	32
DDRS	DMA Doorbell Set	0x0	0x100C	32
DCKE	DMA Clock Enable	0x0	0x1010	32
DCKES	DMA Clock Enable Set	0x0	0x1014	32
DCKEC	DMA Clock Enable Clear	0x0	0x1018	32
DMACP	DMA Channel Programmable	0x0	0x101C	32
DSIRQP	Channel soft IRQ to MCU	0x0	0x1020	32
DSIRQM	Channel soft IRQ mask	0xffffffff	0x1024	32
DCIRQP	Channel IRQ to MCU	0x0	0x1028	32
DCIRQM	Channel IRQ to MCU mask	0xffffffff	0x102C	32
DMCS	MCU Control and Status	0xE1	0x1030	32
DMNMB	MCU Normal MailBox	0x?	0x1034	32
DMSMB	MCU Security MailBox	0x?	0x13421038	32
DMINT	MCU Interrupt	0x3	0x1342103C	32

NOTES:

Grey ones are obsolete registers defined in previous JZ SOC. They are relative to clock gating and have no real function, and they are not supported any longer.

24.5 DMA Channel Register Definition

24.5.1 DMA Source Address (DSAn, n = 0 ~ 31)

	DSAn																															
																	BASE + 0x0 + (n*0x20)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	SA	Source physical address.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.5.2 DMA Target Address (DTAn, n = 0 ~ 31)

	DTAn																															BASE + 0x4 + (n*0x20)
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	TA	Target physical address.	RW

24.5.3 DMA Transfer Count (DTCn, n = 0 ~ 31)

	DTCn																															BASE + 0x8 + (n*0x20)
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								TC																							
RST	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:24	Reserved	Write has no effect, read as zero.	R
23:0	TC	TC records the number of data unit to be transferred. Moreover, when Stride transfer mode is enabled, TC composes of two parts: the lower 16 bits is the number of data unit for sub-block transfer, the higher 8 bits is the number of sub-block. TC automatically counts down to 0 at the end.	RW

24.5.4 DMA Request Types (DRTn, n = 0 ~ 31)

	DRTn																															BASE + 0xC + (n*0x20)
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										RT					
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?

Bits	Name	Description	RW
31:6	Reserved	Write has no effect, read as zero.	R
5:0	RT	Transfer request type.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 24-5 Transfer Request Types

RT5-0	Description
000000	Reserved.
000001	Reserved.
000010	I2S2 transmit-fifo-empty transfer request.
000011	I2S2 receive-fifo-full transfer request.
000100	I2S1 transmit-fifo-empty transfer request.
000101	I2S1 receive-fifo-full transfer request.
000110	I2S0 transmit-fifo-empty transfer request.
000111	I2S0 receive-fifo-full transfer request.
001000	Auto-request. (external address → external address)
001001	SADC receive-fifo-full transfer request
001010	External request with DREQn. (external address ←→ external device with DACKn)
001011	TCU channel n. (overflow interrupt, external address→external address space)
001100	UART4 transmit-fifo-empty transfer request. (external address → UTHR)
001101	UART4 receive-fifo-full transfer request. (URBR → external address)
001110	UART3 transmit-fifo-empty transfer request. (external address → UTHR)
001111	UART3 receive-fifo-full transfer request. (URBR → external address)
010000	UART2 transmit-fifo-empty transfer request. (external address → UTHR)
010001	UART2 receive-fifo-full transfer request. (URBR → external address)
010010	UART1 transmit-fifo-empty transfer request. (external address → UTHR)
010011	UART1 receive-fifo-full transfer request. (URBR → external address)
010100	UART0 transmit-fifo-empty transfer request. (external address → UTHR)
010101	UART0 receive-fifo-full transfer request. (URBR → external address)
010110	SSI0 transmit-fifo-empty transfer request.
010111	SSI0 receive-fifo-full transfer request.
011000	SSI1 transmit-fifo-empty transfer request.
011001	SSI1 receive-fifo-full transfer request.
011010	MSC0 transmit-fifo-empty transfer request.
011011	MSC0 receive-fifo-full transfer request.
011100	MSC1 transmit-fifo-empty transfer request.
011101	MSC1 receive-fifo-full transfer request.
011110	MSC2 transmit-fifo-empty transfer request.
011111	MSC2 receive-fifo-full transfer request.
100000	PCM0 transmit-fifo-empty transfer request.
100001	PCM0receive-fifo-full transfer request.
100010	PCM1 transmit-fifo-empty transfer request.
100011	PCM1receive-fifo-full transfer request.
100100	SMB0 transmit-fifo-empty transfer request.
100101	SMB0 receive-fifo-full transfer request.
100110	SMB1 transmit-fifo-empty transfer request.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

100111	SMB1 receive-fifo-full transfer request.
101000	SMB2 transmit-fifo-empty transfer request.
101001	SMB2 receive-fifo-full transfer request.
101010	SMB3 transmit-fifo-empty transfer request.
101011	SMB3 receive-fifo-full transfer request.
101100	SMB4 transmit-fifo-empty transfer request.
101101	SMB4 receive-fifo-full transfer request.
101110	DES transmit-fifo-empty transfer request.
101111	DES receive-fifo-full transfer request.
110000 ~ 111111	reserved

NOTES:

1 Only auto request can be concurrently set in all channels with different source and target address.

2 For on-chip device DMA request, the corresponding source or target address that map to on-chip device must be set as fixed.

24.5.5 DMA Channel Control/Status (DCSn, n = 0 ~ 31)

	DCSn																BASE + 0x10 + (n*0x20)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NDES	DES8	TOC														CDOA											AR	TT	HLT	TOE	CTE
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	NDES	Descriptor or No-Descriptor Transfer Select. 0: Descriptor Transfer; 1: No-descriptor Transfer.	RW
30	DES8	Descriptor 8 Word. 0: 4-word descriptor; 1: 8-word descriptor.	RW
29:16	TOC	Time out counter. Available when TOE=1，undefined scratch field	RW
15:8	CDOA	Copy of offset address of last completed descriptor from that in DMA command register. (The field is Ignored in No-Descriptor Transfer)	RW
7:5	Reserved	Write has no effect, read as zero.	R
4	AR	Address Error. Hardware set it to 1 and software clear it to 0. 0: no address error; 1: address error occurred.	RW
3	TT	Transfer Terminate. 0: Descriptor or No-Descriptor DMA transfer does not end 1: No-Link Descriptor or No-Descriptor DMA transfer end	RW
2	HLT	DMA halt. Hardware sets it to 1 when an abnormal case occurs during	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		transfer, then software has to clear it to 0 for re-using the channel later.	
1	TOE	Time out enable for transaction of a data unit. 0 – no time out monitor for a data unit's transfer 1 – after a data unit has already been transferred but not completed yet, TOC field monitors the maximum cycles to be cost, and once exceeding the preset value occurs, a time-out error arises and HLT will be set 1 but TT keeps 0, moreover, if DCMn.TIE=1, a time-out interrupt arises.	RW
0	CTE	Channel transfer enable: 0: disable; 1: enable.	RW

NOTES:

1. For a working channel, when an address error occurs or an on-going transfer is stopped by software writing 0 to CTE, HLT becomes 1 but TT keeps 0 that denotes an abnormal situation occurs.
2. Software can stop any working channel at any time. However, after setting 0 to CTE, software must read TT, HLT and CTE to ensure the stop behavior has been done for the channel, that is,
   a) TT=0, HALT=0, CTE =1, not halt yet, need polling again
   b) TT=1, HALT=0, CTE=0, the channel just ends its work when software wants to stop it.
   c) TT=0, HALT=1, CTE=0, the channel has been stopped but its transfer not complete yet.

24.5.6 DMA Channel Command (DCMn, n = 0 ~ 31)

	DCMn																															BASE + 0x14 + (n*0x20)
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved				SID		DID		SAI	DAI	Reserved		RDIL				SP		DP		Reserved	TSZ			Reserved					STDE	TIE	LINK
RST	0	0	0	0	0	0	0	0	?	?	0	0	?	?	?	?	?	?	?	?	0	?	?	?	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:28	Reserved	Write has no effect, read as zero	R
27:26	SID	Source identification. Only be used for special channel 0/1. 0: Source is TCSM. 1: Source is BCH (for channel 0) / NEMC (for channel 1) 2: Source is DDR (for channel 1) / Reserved (for channel 0) 3: Reserved.	RW
25:24	DID	Destination identification. Only be used for special channel 0/1. 0: Destination is TCSM. 1: Destination is BCH (for channel 0) / NEMC (for channel 1) 2: Destination is DDR (for channel 1) / Reserved (for channel 0) 3: Reserved.	RW
23	SAI	Source Address Increment. 0: no increment; 1: increment.	RW
22	DAI	Destination Address Increment. 0: no increment; 1: increment.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

19:16	RDIL	Recommended data unit size (unit: byte) for triggering device's DMA request when TSZ is autonomy.	RW
15:14	SP	Source port width. 00: 32-bit; 01: 8-bit; 10: 16-bit; 11: reserved.	RW
13:12	DP	Destination port width. 00: 32-bit; 01: 8-bit; 10: 16-bit; 11: reserved.	RW
11	Reserved	Write has no effect, read as zero.	R
10:8	TSZ	Transfer Data Size of a data unit. 000: 32-bit; 001: 8-bit; 010: 16-bit; 011: 16-byte; 100: 32-byte; 101: 64-byte; 110: 128-byte; 111: autonomy;	RW
7:3	Reserved	Write has no effect, read as zero.	R
2	STDE	Stride Disable/Enable. 0: address stride disable; 1: address stride enable.	RW
1	TIE	Transfer Interrupt Enable (TIE). 0: disable interrupt; 1: enable interrupt when TT is set to 1.	RW
0	LINK	Descriptor Link Enable. 0: disable; 1: enable. (the field is ignored in No-Descriptor Transfer mode)	RW

NOTES:

SP and DP are only available for device FIFO or NEMC. But when transfer data between device and DDR memory, must set the port width of DDR side as the same as the device side; however, when both source and destination are in DDR memory, both SP and DP must be 32-bit by setting 0.

Table 24-6 Available RDIL

RDIL	Description
0	Reserved
1	Recommended data unit is 1 bytes
2	Recommended data unit is 2 bytes
3	Recommended data unit is 3 bytes
4	Recommended data unit is 4 bytes
5	Recommended data unit is 8 bytes
6	Recommended data unit is 16 bytes
7	Recommended data unit is 32 bytes
8	Recommended data unit is 64 bytes
9	Recommended data unit is 128 bytes
10~15	Reserved

NOTES:

1. if TSZ is autonomy, total bytes to be transferred equal the value of DCTn.TC (so maximum transfer bytes for autonomy are 16MB -1).
2. Programmer must care the detail of the FIFO of the device binding with relative DMA channel to set the correct and best recommended data unit value according to FIFO's data width and depth, refer to following table.

FIFO form	Best recommended RDIL
8x8bits	4 (half of total entries just accommodate 4 bytes)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8x16bits	5 (half of total entries just accommodate 8 bytes)
32x8bits	6 (half of total entries just accommodate 16 bytes)
16x16bits	6 (half of total entries just accommodate 16 bytes)
64x8bits	7 (half of total entries just accommodate 32 bytes)
64x32bits	9 (half of total entries just accommodate 128 bytes)
128x16bits	9 (half of total entries just accommodate 128 bytes)
128x32bits	9 (because maximum data unit is 128 bytes)

Moreover, programmer must be carefully obey following rules for correctness and best performance concern of DMA transfer:

1. ***auto-request, both source and destination device are DDR memory***. No alignment constraint for address, however, both source port width and destination port width must be set 32bit, and transferred bytes in a bus transaction denoted by TSZ (when TSZ !=7) must not be less than 4. Setting autonomy (TSZ=7) is the best choice for transfer efficiency[*0]
2. ***auto-request, both source and destination device are NEMC*****,** then
   a) if both source and destination port width are 8bit, no alignment constraint for address, setting autonomy (TSZ=7) is the best choice for transfer efficiency.
   b) If either source port or destination port width is not 8bit, must not set autonomy to TSZ. The relative source or destination address must be 2-byte (port width is 16bit) or 4-byte (port width is 32bit) aligned. Moreover, the byte number denoted by TSZ must not be less than the least common multiple of the byte number denoted by source port width and destination port width. Larger TSZ means better transfer efficiency.
3. ***auto-request, source device is NEMC while destination device is DDR memory*****,** then
   a) if source port width is 8bit. no alignment constraint for address, setting autonomy (TSZ=7) is the best choice for transfer efficiency.
   b) if source port width is 16bit, source address must be 2-byte aligned and TSZ must be the times of 2-byte (so must not set autonomy to TSZ); no alignment constraint for destination address, larger TSZ means better transfer efficiency.
   c) If source port width is 32bit, source address must be 4-byte aligned and TSZ must be the times of 4-byte (so must not set autonomy to TSZ); no alignment constraint for destination address, larger TSZ means better transfer efficiency.
4. ***auto-request, source device is DDR memory while destination device is NEMC***, then
   a) if destination port width is 8bit, no alignment constraint for address, setting autonomy (TSZ=7) is the best choice for transfer efficiency
   b) if destination port width is 16bit, destination address must be 2-byte aligned and TSZ must be the times of 2-byte (so must not set autonomy to TSZ); no alignment constraint for source address, larger TSZ means better transfer efficiency.
   c) If destination port width is 32bit, destination address must be 4-byte aligned and TSZ must be the times of 4-byte (so must not set autonomy to TSZ); no alignment constraint for source address, larger TSZ means better transfer efficiency.
5. ***external device fifo receive request, destination device is DDR memory***, then no alignment constraint for destination address, and
   a) if non-autonomy (TSZ !=7), for transferred bytes in a bus transaction denoted by TSZ, it

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when source port width is 16bit or times of 4-byte when source port width is 32bit.

b) If autonomy (TSZ=7), for transferred bytes in a bus transaction denoted by RDIL, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when source port width is 16bit or times of 4-byte when source port width is 32bit.

6. ***external device fifo receive request, destination device is NEMC***, then

a) if destination port width is 8bit, no alignment constraint for destination address, and if non-autonomy (TSZ !=7), for transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when source port width is 16bit or times of 4-byte when source port width is 32bit.
If autonomy (TSZ=7), for transferred bytes in a bus transaction denoted by RDIL, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when source port width is 16bit or times of 4-byte when source port width is 32bit.

b) If destination port width is 16bit, must not set autonomy to TSZ and destination address must be 2-byte aligned. For transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must not be less than the least common multiple of the byte number denoted by source port width and destination port width.

c) If destination port width is 32bit, must not set autonomy to TSZ and destination address must be 4-byte aligned. For transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must not be less than the least common multiple of the byte number denoted by source port width and destination port width.

7. ***external device fifo transmit request, source device is DDR memory***, then no alignment constraint for source address, and

a) if destination port width is 8bit, setting autonomy (TSZ=7) is the best choice for transfer efficiency

b) if destination port width is 16bit, when setting autonomy (TSZ=7), total transferred bytes must be the times of 2-byte, and for transferred bytes in a bus transaction denoted by RDIL, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be the times of 2-byte; when setting non-autonomy (TSZ!=7), for transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be the times of 2-byte.

c) if destination port width is 32bit, when setting autonomy (TSZ=7), total transferred bytes must be the times of 4-byte, and for transferred bytes in a bus transaction denoted by RDIL, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be the times of 4-byte; when setting non-autonomy (TSZ!=7), for transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be the times of 4-byte.

8. ***external device fifo transmit request, source device is NEMC***, then

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

a) if source port width is 8bit, no alignment constraint for source address, and if non-autonomy (TSZ !=7), for transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when destination port width is 16bit or times of 4-byte when destination port width is 32bit.

If autonomy (TSZ=7), for transferred bytes in a bus transaction denoted by RDIL, it must not exceed the critical value of triggering DMA request[*1] meanwhile must be times of 2-byte when destination port width is 16bit or times of 4-byte when destination port width is 32bit.

b) If source port width is 16bit, must not set autonomy to TSZ and source address must be 2-byte aligned. For transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must not be less than the least common multiple of the byte number denoted by source port width and destination port width.

c) If source port width is 32bit, must not set autonomy to TSZ and source address must be 4-byte aligned. For transferred bytes in a bus transaction denoted by TSZ, it must not exceed the critical value of triggering DMA request[*1] meanwhile must not be less than the least common multiple of the byte number denoted by source port width and destination port width.

NOTES:

*0 - when TSZ is not autonomy, total bytes to be transferred are bytes represented by TSZ * TC, and each bus transaction can only transfer bytes represented by TSZ, so larger data unit is better for transfer efficiency. But since in practical, in most cases, total bytes to be transferred are not the times of the best data unit, or source and/or destination address violates alignment, thus **setting autonomy for TSZ is recommended** when condition is ok.

*1 - Please refer to relative device's spec for detail information of how to set critical trigger value.

24.5.7 DMA Descriptor Address (DDAn, n = 0 ~ 31)

This register is ignored in No-Descriptor Transfer mode.

	DDAn																															
																		BASE + 0x18 + (n*0x20)														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBA																				DOA								Reserved			
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0

Bits	Name	Description	RW
31:12	DBA	Descriptor Base Address.	RW
11:4	DOA	Descriptor Offset Address.	RW
3:0	Reserved	Write has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.5.8 DMA Stride Difference (DSDn, n = 0 ~ 31)

This register is ignored in No-Descriptor Transfer mode.

When address stride transfer mode is enabled in Descriptor mode, after a sub-block's transfer defined in DTCn completes, the source or target stride difference will be added to perform the new sub-block's start address, and the transfer will keep going until the transfer ends which means TC in DTCn reaches 0.

	DSDn																BASE + 0x1C + (n*0x20)															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSD																SSD															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:16	TSD	Target Stride Difference (next sub-block's start address – current sub-block's end address - 1), value range is -32768 ~ 32767	RW
15:0	SSD	Source Stride Difference (next sub-block's start address – current sub-block's end address - 1), value range is -32768 ~ 32767	RW

24.6 DMA Global Register Definition

24.6.1 DMA Control

	DMAC																BASE+0x1000															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FMSC	FSSI	FTSSI	FUART	FAIC	Reserved					INTCC					INTCE	Reserved												HLT	AR	CH01	DMAE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	FMSC	0: 0.5-data/clk for MSC on APB;1: 1-data/clk for MSC on APB.	RW
30	FSSI	0: 0.5-data/clk for SSI on APB;1: 1-data/clk for SSI on APB.	RW
29	FTSSI	0: 0.5-data/clk for TSSI on APB;1: 1-data/clk for TSSI on APB.	RW
28	FUART	0: 0.5-data/clk for UART on APB;1: 1-data/clk for UART on APB.	RW
27	FAIC	0: 0.5-data/clk for AIC on APB;1: 1-data/clk for AIC on APB.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

21:17	INTCC	Which channel is bound with INTC_IRQ. Only available when INTCE is enabled	RW
16	INTCE	Permit INTC_IRQ to be bound to one of programmable channel. 0, INTC_IRQ is ignored; 1, INTC_IRQ is bound to channel n, n equals to INTCC.	RW
15:4	Reserve	Write has no effect, read as zero.	R
3	HLT	Global halt status, halt occurs in any channel, the bit should be set to 1 by hardware and cleared to 0 by software. 0: no halt occurred 1: halt occurred in one or more channels	RW
2	AR	Global address error status, address error occurs in any channel, the bit should be set to 1 by hardware and cleared to 0 by software. 0: no address error occurred 1: address error occurred	RW
1	CH01	Special channel 0 and channel 1 enable. 1 –enable 0 – disable.	RW
0	DMAE	Global DMA enable. 0: disable DMA 1: enable DMA	RW

NOTES:

1. any of FMSC/FSSI/FTSSI/FUART/FAIC has been set，the corresponding DMA transfer on APB for MSC(MSC1, MSC2), SSI(SSI1), UART0~4, AIC will be in fast mode.
2. DMAE is a global switch, so software must be careful to toggle it from 0 to 1 or 1 to 0. It is **UNPREDICTABLE** to set 0 to DMAE abruptly when some channels are still working.
3. For more detail of CH01, refer to later chapter of special channel 0 and channel 1

24.6.2 DMA Interrupt Pending (DIRQP)

	DIRQP																															BASE+0x1004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	IRQn	IRQn denotes pending IRQ to main CPU for corresponding channel. 0: normal DMA transfer is in-progress or the channel is idle 1: an address error occurs (only available for UARTn) or normal DMA transfer done; moreover, if the relative channel is programmable, MCU can only set 1 to it. Hardware can only set 1 to it and main CPU can only set 0 to it.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.6.3 DMA Doorbell (DDB)

	DDB																															BASE+0x 1008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DB31	DB30	DB29	DB28	DB27	DB26	DB25	DB24	DB23	DB22	DB21	DB20	DB19	DB18	DB17	DB16	DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	DBn	DMA Doorbell. Software sets it to 1 and hardware clears it to 0. 0: disable DMA controller to fetch the first descriptor or DMA controller clears it to 0 as soon as it starts to fetch the descriptor 1: Writing 1 to DDS will set the corresponding DBn bit to 1 and enable DMA controller to fetch the first descriptor For example, write 0x00000001 to DDS, DB0 bit is set to 1 and enable DMA channel 0 to fetch its first descriptor. Write 0 to DDS, no meaning.	RWS

24.6.4 DMA Doorbell Set (DDS)

	DBS																															BASE+0x 100C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBS31	DBS30	DBS29	DBS28	DBS27	DBS26	DBS25	DBS24	DBS23	DBS22	DBS21	DBS20	DBS19	DBS18	DBS17	DBS16	DBS15	DBS14	DBS13	DBS12	DBS11	DBS10	DBS9	DBS8	DBS7	DBS6	DBS5	DBS4	DBS3	DBS2	DBS1	DBS0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	DBSn	DMA Doorbell Set for each channel. Read as zero, and when write: 0: ignore 1: Set the corresponding DBn bit to 1	W

24.6.5 DMA Channel Programmable (DMACP)

	DMACP																															BASE+0x 101C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCP31	DCP30	DCP29	DCP28	DCP27	DCP26	DCP25	DCP24	DCP23	DCP22	DCP21	DCP20	DCP19	DCP18	DCP17	DCP16	DCP15	DCP14	DCP13	DCP12	DCP11	DCP10	DCP9	DCP8	DCP7	DCP6	DCP5	DCP4	DCP3	DCP2	DCP1	DCP0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	DCPn	Channel programmable enable. 0, compatible with previous JZ SOC; 1, firmware controlled channel	RW

24.6.6 DMA Soft IRQ Pending

	DSIRQP																															BASE+0x 1020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SIRQ31	SIRQ30	SIRQ29	SIRQ28	SIRQ27	SIRQ26	SIRQ25	SIRQ24	SIRQ23	SIRQ22	SIRQ21	SIRQ20	SIRQ19	SIRQ18	SIRQ17	SIRQ16	SIRQ15	SIRQ14	SIRQ13	SIRQ12	SIRQ11	SIRQ10	SIRQ9	SIRQ8	SIRQ7	SIRQ6	SIRQ5	SIRQ4	SIRQ3	SIRQ2	SIRQ1	SIRQ0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	SIRQn	Channel soft IRQ to MCU. Write is ignored 0, no soft IRQ to MCU; 1, pending soft IRQ to MCU. Note that the register is read-only for other masters but is entirely controlled (R/W) by the MCU of PDMA	R

24.6.7 DMA Soft IRQ Mask

	DSIRQM																															BASE+0x 1024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SIRQM31	SIRQM30	SIRQM29	SIRQM28	SIRQM27	SIRQM26	SIRQM25	SIRQM24	SIRQM23	SIRQM22	SIRQM21	SIRQM20	SIRQM19	SIRQM18	SIRQM17	SIRQM16	SIRQM15	SIRQM14	SIRQM13	SIRQM12	SIRQM11	SIRQM10	SIRQM9	SIRQM8	SIRQM7	SIRQM6	SIRQM5	SIRQM4	SIRQM3	SIRQM2	SIRQM1	SIRQM0
RST	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31:0	SIRQMn	Channel soft IRQ mask. Write is ignored 0, not mask corresponding channel's soft IRQ; 1, mask corresponding channel's soft IRQ Note that the register is read-only for other masters but is entirely controlled (R/W) by the MCU of PDMA	R

24.6.8 DMA Channel IRQ Pending to MCU

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DCIRQP																															BASE+0x 1028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CIRQ31	CIRQ30	CIRQ29	CIRQ28	CIRQ27	CIRQ26	CIRQ25	CIRQ24	CIRQ23	CIRQ22	CIRQ21	CIRQ20	CIRQ19	CIRQ18	CIRQ17	CIRQ16	CIRQ15	CIRQ14	CIRQ13	CIRQ12	CIRQ11	CIRQ10	CIRQ9	CIRQ8	CIRQ7	CIRQ6	CIRQ5	CIRQ4	CIRQ3	CIRQ2	CIRQ1	CIRQ0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	CIRQn	Channel IRQ Pending to MCU. The register is read-only 0, no channel IRQ to MCU; 1, pending channel IRQ to MCU When channel n is not programmable, the CIRQn is ignored by hardware. When channel n is programmable and DCMn.TIE is active 1, DIRQP.IRQn can not be set automatically any more after address error occurs or TT becomes active 1, instead, if DCIRQMn==0, CIRQn may be set 1 immediately to raise an IRQ to MCU. Moreover, when an channel m with above attribute is bound with the INTC_IRQ, if DCIRQMm==0, an active INTC_IRQ request will trigger an active CIRQn immediately	R

24.6.9 DMA Channel IRQ to MCU Mask

	DCIRQM																															BASE+0x 102C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CIRQM31	CIRQM30	CIRQM29	CIRQM28	CIRQM27	CIRQM26	CIRQM25	CIRQM24	CIRQM23	CIRQM22	CIRQM21	CIRQM20	CIRQM19	CIRQM18	CIRQM17	CIRQM16	CIRQM15	CIRQM14	CIRQM13	CIRQM12	CIRQM11	CIRQM10	CIRQM9	CIRQM8	CIRQM7	CIRQM6	CIRQM5	CIRQM4	CIRQM3	CIRQM2	CIRQM1	CIRQM0
RST	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31:0	CIRQMn	Mask of Channel IRQ to MCU. The register is read-only for other master but is entirely controlled (R/W) by the MCU of PDMA 0, not mask corresponding channel's IRQ to MCU; 1, mask corresponding channel's IRQ to MCU When channel n is not programmable, the CIRQMn is ignored by hardware. When channel n is programmable and DCMn.TIE is active 1, CIRQMn can mask corresponding channel's IRQ to MCU.	R

24.6.10 Programmable Channel Bound With INTC_IRQ

In this SOC, all peripheral IRQs originally being issued to main CPU now can be taken over by PDMA through binding special INTC_IRQ to one of programmable channel. Note that INTC_IRQ from INTC may be different to normal IRQ of INTC to main CPU, please refer to the spec of INTC for

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

detail of how to configure INTC_IRQ sources. In practical, maybe it is a better tradeoff to let MCU of PDMA take over trivial transactions of some devices from main CPU.

24.6.11 Special Channel 0 and Channel 1

When a product integrating this SOC need use raw NAND chipset properly through PDMA, the only method is setting special channel 0 and 1, following key bit fields must be preset:

1. DMAC.CH01 must be set 1
2. DCM0.TIE and DCM1.TIE must be set 1
3. DCM0.LINK and DCM1.LINK must be set 0
4. DCS0.NDES and DCS1.NDES must be set 1
5. DCM0.STDE and DCM1.STDE must be set 0
6. DCM0.SAI and DCM0.DAI must be set 1
7. DCM1.SAI and DCM1.DAI must be set 1
8. DCM0.SP and DCM0.DP must be set 0
9. DCM1.SP and DCM1.DP must be set 0
10. DCM0.TSZ and DCM1.TSZ must be set 7
11. DMACP.DCP0 and DMACP.DCP1 must be set 1 after relative firmware has been prepared.

24.7 MCU

MCU in PDMA is a mini CPU compatible with XBurst-1 ISA without implementing CACHE, MMU, DEBUG, FPU and MXU. It is very similar as the AUX in VPU but has different memory mapped control and status register.

24.7.1 MCU Control & Status

	DMCS																															BASE+0x 1030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SLEEP	SCMD	Reserved						SC _OFF																BCH			BTB_INV	SC_CALL			SW_RST
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1

Bits	Name	Description	RW
31	SLEEP	Sleep status of MCU. Read only by main CPU.	R
30	SCMD	Security Mode. Read only by main CPU.	R
29:24	reserved	Write has no effect, read as zero.	R
23:8	SC_OFF	Set the offset of the caller's data structure for SC_CALL	RW
7	BCH_DB	Block syndrome of BCH decoding, low level active	R
6	BCH_DF	BCH decoding finished, low level active	R
5	BCH_EF	BCH encoding finished, low level active	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4	BTB_INV	Writing 1 invalidates BTB in MCU. Writing 0 has no effect, read as zero.	W
3	SC_CALL	Security Call. Writing 1 triggers a security exception which directs MCU to run security code located in the on-chip security ROM., read as zero.	W
2:1	reserved	Write has no effect, read as zero.	R
0	SW_RST	Software reset. 1, MCU keeps at reset state; 0, do not soft-reset MCU any more	RW

NOTES:

BCH_DB, BCH_DF and BCH_EF are only available when special channel 0 and channel 1 is used for implementing storage of raw NAND chipsets. Please refer to BCH spec for detail.

24.7.2 MCU Normal MailBox

	DMNMB																															BASE+0x 1034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	MB	Contents of Normal Mailbox. Writing any value to the register will trigger a normal mailbox IRQ to main CPU when relative MASK field in DMINT is not set	RW

24.7.3 MCU Security MailBox

	DMSMB																															BASE+0x 1038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	MB	Contents of Security Mailbox. Writing any value to the register will trigger a security mailbox IRQ to main CPU when relative MASK field in DMINT is not set. Note that writing the register is inhibited by HW when DMCS.SCMD == 0	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.7.4 MCU Interrupt

	DMINT																															BASE+0x 103C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved														S_IP	N_IP	Reserved														S_IMSK	N_IMSK
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bits	Name	Description	RW
31:18	Reserved	Write has no effect, read as zero.	R
17	S_IP	Security Mailbox IRQ pending. 1-pending IRQ. Hardware only set 1 by writing DMSMB when DMCS.SCMD == 1. Can be clear to 0 in any mode.	RW
16	N_IP	Normal Mailbox IRQ pending. 1-pending IRQ. Hardware only set 1 by writing DMNMB. Can be clear to 0 in any mode.	RW0
15:2	Reserved	Write has no effect, read as zero.	R
1	S_IMSK	Security Mailbox IRQ mask. 1-mask; 0-not mask Can only be modified in security mode.	RW
0	N_IMSK	Normal Mailbox IRQ mask. 1-mask; 0-not mask	RW

24.7.5 Multiple Bank Tightly Coupled Sharing Memory

In MCU, there is a Tightly Coupled Sharing Memory (TCSM) with 32KB size and total 8 banks. Multiple- bank TCSM permits several masters accessing different banks in parallel. TCSM works at the same speed as MCU and both instruction fetch and data read/write to TCSM by MCU can be performed in one cycle.

BANK0~BANK7 can be configured with "Deep Sleep" mode, all the data read out are "0" during this mode, When out of "Deep Sleep" mode, the data in ram can recover.. BANK4~BANK7 can also be configured with "Shutdown" mode, but BANK0~BANK3 can not, all the data read out are "0" during this mode. The data is still "0" when out of "Shutdown" mode, which means the data has lost and can not be recovered. Please reference "CPM spec" about how to configure "Deep Sleep" and "Shutdown".

Table 24-7 TCSM space

	VA (only available for MCU)	PA (not remapped)	PA (remapped)
BANK0	F4000000h ~ F4000FFFh	13422000h ~ 13422FFFh	F3422000h ~ F3422FFFh
BANK1	F4001000h ~ F4001FFFh	13423000h ~ 13423FFFh	F3423000h ~ F3423FFFh
BANK2	F4002000h ~ F4002FFFh	13424000h ~ 13424FFFh	F3424000h ~ F3424FFFh
BANK3	F4003000h ~ F4003FFFh	13425000h ~ 13425FFFh	F3425000h ~ F3425FFFh
BANK4	F4004000h ~ F4004FFFh	13426000h ~ 13426FFFh	F3426000h ~ F3426FFFh

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

BANK5	F4005000h ~ F4005FFFh	13427000h ~ 13427FFFh	F3427000h ~ F3427FFFh
BANK6	F4006000h ~ F4006FFFh	13428000h ~ 13428FFFh	F3428000h ~ F3428FFFh
BANK7	F4007000h ~ F4007FFFh	13429000h ~ 13429FFFh	F3429000h ~ F3429FFFh

24.7.6 CP0 Registers of MCU

Available CP0 registers of MCU are not entirely compatible with MIPS PRA spec, they are listed below.

CP0 Name (number)	**Description**	***slt***
STATUS (12)	Bit 31~27:: reserved, read as zero Bit26: readable; only 1 can bet written into*1 Bit25: readable; only 1 can bet written into*1 Bit24: only being active during running Security Call routine *2 Bit2: ERL, Bit1: EXL, Bit0: IE	*0*
INTCTL (12)	Bit31~3: reserved, read as zero; Bit2: IRQ pending of soft IRQ to MCU; Bit1: IRQ pending of channel IRQ to MCU; Bit0: IRQ pending of INTC_IRQ Note that the register is read-only by MCU	*1*
CAUSE (13)	Bit31:BD， whether the interrupted PC is located in branch delay slot Bit30 ~ 0: reserved, read as zero	*0*
SeCAUSE (13)	Bit31:BD, similar as CAUSE, but being active when an IRQ interrupted a running Security Call routine. Bit30 ~ 0: reserved, read as zero	*1*
EPC (14)	IRQ Exception program counter	*0*
SePC (24)	Security Exception program counter	*0*
ErrorPC (30)	*Reset exception program counter*	*0*

NOTE:

*1 : once value 1 is set, only chip reset signal can clear it to 0.

*2 : STATUS.Bit24 is hard wired outside of PDMA.

24.7.7 Normal Exceptions Accepted by MCU

There are two kinds of normal exceptions that can be accepted by MCU: RESET and IRQ.

Note that RESET exception has the handler entry: 0xF4000000, while IRQ has another one: 0xF4000100.

Once the MCU accepts a normal exception, according to the definition of XBurst-1 Programming Manual, current interrupted PC will be pushed into CP0.EPC or CP0.ErrPC, and further CP0.STATUS.ERL or CP0.SATUS.EXL will be automatically set 1. Firmware can take use of MTC0/MFC0 to access these three CP0 registers: EPC, ErrPC and STATUS. Firmware must use ERET instruction to return to the interrupted locale to resume executing.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.7.8 How to Boot MCU Up

After power-on reset, MCU keeps at reset state due to DMCS.SW_RST==1. To boot MCU up, first of all, RESET exception handler must be loaded into TCSM at the physical address range beginning from 0x13422000 or 0xF3422000 for remapped case. Programmer can use main CPU or some Non-programmable DMA channel to finish this loading work. Then clear DMCS.SW_RST to 0 by main CPU so that MCU can acknowledge a RESET exception, now MCU begins to run from the PC 0xF4000000. Following is a simple example.

1. Prepare a simple RESET exception handler, total 3 instructions:

```
1:  WAIT   //sleep
B   1b     //endless loop
NOP        //delay slot
```

2. Load the handler into TCSM at the physical address range beginning from 0x13422000 or 0xF3422000 for remapped case.
3. Clear DMCS.SW_RST to 0 by main CPU

Now MCU will execute WAIT instruction and then enter sleep state

24.7.9 Security features

In additional to the above security used registers, several security objects and functions, such as security ROM, RAM and security call, are added to PDMA and its MCU, to compose the JZ4780 security engine. For the details, please reference to security engine section.

24.8 DMA manipulation

24.8.1 Descriptor Transfer Mode

24.8.1.1 Non-Stride (1-Dimensional) Transfer Mode

To do proper Descriptor DMA transfer in a channel, do following steps:

1 Check whether the status of DMA controller is available, that is, for global control (DMAC), ensure that DMAC.AR=0 and DMAC.HLT=0; and then for expected channel n, ensure that DCSn.AR=0, DCSn.HLT=0, DCSn.TT=0, DTCn=0.

2 guarantee DCSn.NDES=0, and select 4 word or 8 word descriptor by DCSn.DES8.

3 Build descriptor in memory. Write the first descriptor's address in DDAn and the address must be 16Bytes aligned for 4word descriptor and 32Bytes aligned for 8word descriptor, otherwise, the hardware behavior later is **UNPREDICTABLE**. The address includes two parts: Base and Offset address. If the descriptor is linked, the 32-bit address of next descriptor is composed of 20-bit Base address in DDAn and 8-bit Offset address in DESn.CDOA. See Table 24-8 Descriptor Structure for the detailed descriptor structure.

4 Don't update DRTn manually by software, instead, hardware will manage the request type of descriptor's transfer automatically.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5 Set 1 to the corresponding bit in DDB to initiate descriptor fetch.
6 Set DMAC.DMAE=1 and expected DCSn.CTE=1 to activate channel n.
7 Hardware will automatically clear the corresponding bit in DDB as soon as it starts to fetch the descriptor.
8 Wait for DMA request from peripherals to start channel n's DMA transfer.
9 After DMA transfers described by the current descriptor complete, if current DCMn.LINK =0, then DCSn.TT will be set to 1 immediately. Further, if DCMn.TIE=1, it will issue an interrupt request. However, if DCMn.LINK=1, next descriptor will be automatically fetched to repeat step 8 and 9 but ignore setting DCSn.TT until a final descriptor with settting of DCMn.LINK=0 has been done.
10 When channel n's transfer ends normally (DCSn.TT=1) or abnormally (DCSn.HLT=1), to correctly reuse the channel later, software must close the channel first by setting 0 to DCSn.CTE, and then safely clear DCSn.TT or DCSn.HLT to 0.

Table 24-8 Descriptor Structure

Word	Bit	Name	Function
1st (DES0)	31	EACKS	External DMA DACKn output polarity select
	30	EACKM	External DMA DACKn output Mode select
	29-28	ERDM	External DMA request detection Mode
	27-24	Reserved	--
	23	SAI	Source Address Increment
	22	DAI	Target Address Increment
	21-20	Reserved	--
	19-16	RDIL	Request Detection Interval Length
	15-14	SP	Source port width
	13-12	DP	Target port width
	11	Reserved	--
	10-8	TSZ	Transfer Data Size
	7-3	Reserved	--
	2	STDE	Stride transfer enable
	1	TIE	Transfer Interrupt Enable
	0	LINK	Descriptor Link Enable
2nd (DES1)	31-0	DSA	Source Address
3rd (DES2)	31-0	DTA	Target Address
4th (DES3)	31-24	DOA	Descriptor Offset address
	23-0	DTC	Transfer Counter
5th (DES4)	31-16	TSD	Target Stride Address
	15-0	SSD	Source Stride Address
6th(DES5)	31-6	Reserved	--
	5-0	DRT	DMA Request Type
7th(DES6)	31-0	Reserved	--
8th(DES7)	31-0	Reserved	--

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 24-2 Descriptor Transfer Flow

24.8.1.2 Stride (2-dimensional) Transfer Mode

Stride transfer mode means total transfer counts are subdivided into dozens of or even hundreds of sub-blocks with same size, and each adjacent two sub-blocks has fixed nonzero address gap between the tail of preceding one and the head of subsequent one.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 24-3 Example for Stride Transfer Mode

Manipulation steps of Stride Transfer Mode are similar to Non-stride Transfer Mode. However, each new sub-block's start address (source or destination) is performed by adding the next adjacent byte address of very completed sub-block's end address to corresponding stride difference.

24.8.2 No-Descriptor Transfer Mode

To do proper No-Descriptor DMA transfer, do following steps:

1. Check whether the status of DMA controller is available, that is, for global control (DMAC), ensure that DMAC.AR=0 and DMAC.HLT=0; and then for expected channel n, ensure that DCSn.AR=0, DCSn.HLT=0, DCSn.TT=0, DTCn=0.
2. Forx channel n, initialize DSAn, DTAn, DTCn, DRTn, DCSn, DCMn properly.
3. Set DMAC.DMAE=1 and DCSn.CTE=1 to launch DMA transfer.

For a channel with auto-request (DRTn.RT=0x8), the transfer begins automatically when the DCSn.CTE bit and DMAC.DMAE bit are set to 1. While for a channel with other request types, the transfer does not start until a transfer request is issued and detected.

For any channel n, The DTCn value is decremented by 1 for each successful transaction of a data unit. When the specified number of transfer data unit has been completed (DTCn = 0), the transfer ends normally. Meanwhile if DCMn.TIE=1, corresponding bit of DIRQP will be set to 1 to raise an interrupt request. However, during the transfer, if a DMA address error occurs, the transfer should be suspended, both DCSn.AR and DMAC.AR are set to 1 as well as corresponding bit of DIRQP to raise an interrupt request despite of DCMn.TIE.

Sometimes, for example, an UART parity error occurs for a channel that is transferring data between such UART and another terminal. In the case, both DCSn.HLT and DMAC.HLT are set to 1 and the transfer is suspended. Software should identify halt status by checking such two bits and re-configure DMA to let DMA rerun properly later.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.9 DMA Requests

DMA transfer requests are normally generated from either the data transfer source or target, but also they can be issued by on-chip peripherals that are neither the source nor the target. There are two DMA transfer request types: auto-request, and on-chip peripheral request. For any channel n, its transfer request type is determined through DRTn.

24.9.1 Auto Request

When there is no explicit transfer request signal available, for example, memory-to-memory transfer or memory to some on-chip peripherals like GPIO, the auto-request mode allows the DMA to automatically generate a transfer request signal internally. Therefore, when DMA initialization done, once the DMAC.DMAE and DCSn.CTE are set to 1, the transfer begins immediately in channel n which DRTn=0x8.

24.9.2 On-Chip Peripheral Request

In the mode, transfer request signals come from on-chip peripherals or even off-chip devices. When a device issues a peripheral request to its corresponding channel n, the transferred byte number for the request is equals to:

1. byte number determined by the bytes denoted by DCMn.TSZ when DCMn.TSZ != 7
2. byte number determined by the bytes denoted by RDIL when DCMn.TSZ == 7

Be careful that the above number must not exceed the device's expected one otherwise FIFO under-run or over-run may occur. However, if the number is too small than the expected one, DMA's bus efficiency will become worse.

24.10 How to Use Programmable DMA Channel

When system needs to setup one programmable channel for later use of smart data transfer, do following steps:

1. Load elaborate firmware to TCSM, should use physical address 0x13422xxx. Common code fragment of IRQ exception handler must be located at physical address range beginning from 0x13422100.
2. Prepare a DMA channel n using expected transfer mode as description in preceding chapters
3. Must set DCMn.TIE=1 and DMACP.DCPn=1
4. Must ensure DCIRQM.CIRQMn==0
5. Launch the channel n as description in preceding chapters

Now the channel n becomes a working programmable channel. It is the responsibility of local channel IRQ's handler to remove corresponding channel IRQ source by setting 0 to DCSn.TT.

Moreover, if programmer wants to let the channel can be triggered by local soft IRQ, DSIRQM.SIRQMn must be set 0, then at expected moment, the MCU sets 1 to DSIRQP.SIRQn to trigger a soft IRQ. It is the responsibility of this soft IRQ's handler to remove the IRQ by setting 0 to DSIRQP.SIRQn.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

25 SAR A/D Controller

25.1 Overview

The A/D is CMOS low-power dissipation 12bit SAR analog to digital converter. It is developed as an embedded high resolution ADC targeting to the CMOS process and has wide application in portable electronic devices, high-end home entertainment center, communication systems and so on.

The SAR A/D controller is dedicated to control A/D to work at two different modes: Battery (check the battery power), and two auxiliary input. Battery and two auxiliary input can transfer the data to memory through CPU.

Features:

- 3 Channels
- Resolution: 12-bit
- Integral nonlinearity: ±1 LSB
- Differential nonlinearity: ±0.5 LSB
- Speed: up to 200Ksps
- Max Frequency: 2M
- Low power dissipation: 1.5mW(worst)
- Support voltage measurement (Through pin VBAT)
- Support two auxiliary input (Through pin AUX1, AUX2)

25.2 Pin Description

Table 25-1 SADC Pin Description

Name	I/O	Description
VBAT	AI	VBAT direct input *[1] *[2]
AUX1	AI	Auxiliary Input *[3]
AUX2	AI	Auxiliary Input *[3]

NOTE:

*[1]: The VBAT allowed input voltage range is from 0V to 1.2V

*[2]: Users who already deployed resistor networks on board level can use VBAT to direct measure the battery value.

*[3]: The AUX1 and AUX2 allowed input voltage range is from 0V to AVDADC

25.3 Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

25.3.1 Registers Memory Map

Table 25-2 Registers Memory Map-Address Base

Name	Base	Description
ADC	0x10070000	Address base of ADC controller

Table 25-3 SADC Register Description

Name	Description	Reset Value	Address Offset	Access Size
ADENA	ADC Enable Register	0x80	0x000	8
ADCFG	ADC Configure Register	0x00040000	0x004	32
ADCTRL	ADC Control Register	0x3F	0x008	8
ADSTATE	ADC Status Register	0x00	0x00C	8
ADVDAT	ADC VBAT Data Register	0x0000	0x01C	16
ADADAT	ADC AUX Data Register	0x0000	0x020	16
ADCLK	ADC Clock Divide Register	0x00041000	0x028	32
ADSTB	ADC stable Register	0x00000000	0x034	32

25.3.2 Registers and Fields Description

25.3.2.1 ADC Enable Register (ADENA)

The register ADENA is used to trigger A/D to work.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ADENA																															BASE + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								POWER	INUSG1	Reserved		INUSG0		VBATEN	AUXEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:8	Reserved	Writing has no effect, read as zero.	R
7	POWER	SADC Power control bit. 1: SADC power down 0: SADC power on When POWER is set from 1 to 0, you should wait at least 2ms to enable SADC.	RW
6	INUSG1	Inner usage, keep to 0	RW
5:4	Reserved	Writing has no effect, read as zero.	R
3:2	INUSG0	Inner usage, keep to 0	RW
1	VBATEN	VBAT Enable Control. VBATEN can be set to 1 to sample the voltage of battery, and when the value of voltage is ready, VBATEN will be cleared by hardware auto.	RW
0	AUXEN	AUX n Enable Control. AUXEN can be set to 1 to sample the voltage of AUX1, AUX2, and when the value of voltage is ready, AUXEN will be cleared by hardware auto.	RW

NOTES:

1 TVBATEN and AUXEN can be set to 1 at the same time. The priority of the three mode is AUX > VBAT .

2

25.3.2.2 ADC Configure Register (ADCFG)

The register ADCFG is used to configure the A/D.

	ADCFG																															BASE + 0x004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								INUSG2		INUSG1						INUSG0	Reserved												CMD		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:22	INUSG2	Inner usage, keep to 0	RW
21:16	INUSG1	Inner usage, keep to 6'b000100	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15	INUSG0	Inner usage, keep to 0	RW
14:3	Reserved	Writing has no effect, read as zero.	RW
2:0	CMD	CMD is used to choose the current sample command when ADENA.AUXEN is set to 1. <table><tr><th>CMD</th><th>Function</th></tr><tr><td>000</td><td>Reserved</td></tr><tr><td>001</td><td>Measure AUX1 voltage</td></tr><tr><td>010</td><td>Measure AUX2 voltage</td></tr><tr><td>011</td><td rowspan="4">Reserved</td></tr><tr><td>100</td></tr><tr><td>101</td></tr><tr><td>110</td></tr><tr><td>111</td><td>Measure AUX0 voltage(shared with VBAT channel, but ADC reference voltage is AVDADC)</td></tr></table>	RW

25.3.2.3 ADC Control Register (ADCTRL)

The register ADCTRL is used to control A/D to work.

	ADCTRL																															BASE + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										INUSG				VRDYM	ARDYM
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5:2	INUSG	Inner usage, keep to 4'b1111	RW
1	VRDYM	VBAT data ready interrupt mask. 0: enabled 1: masked	RW
0	ARDYM	AUX data ready interrupt mask. 0: enabled 1: masked	RW

25.3.2.4 ADC Status Register (ADSTATE)

The register ADSTATE is used to keep the status of A/D.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ADSTATE																															BASE + 0x00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														VRDY	ARDY
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:2	Reserved	Writing has no effect, read as zero.	R
1	VRDY	VBAT data ready interrupt flag. Write 1 to this bit, the value will be clear. 1: active 0: not active	RWC
0	ARDY	AUX data ready interrupt flag. Write 1 to this bit, the value will be clear. 1: active 0: not active	RWC

25.3.2.5 ADC VBAT Data Register (ADVDAT)

The read-only ADVDAT is a 16-bit register, it keep the sample data of VBAT. 0~11 bits are data.

	ADBDAT																															BASE + 0x01C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				VDATA											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	VDATA	Data of VBAT A/D convert. When write to the register, DATA will be clear to 0.	RW

Notes:

1 The measured voltage V_{BAT} can directly calculate as follows:

$$V_{BAT} = \frac{VDATA}{4096} \bullet 1.2V$$

2 The better accurate voltage can be calculated use the trim data stored in OTP

25.3.2.6 ADC AUX Data Register (ADADAT)

The read-only ADADAT is a 16-bit register, it keep the sample data. 0~11 bits are data.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ADSDAT																															BASE + 0x020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				ADATA											
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	ADATA	Data of AUX. When write to the register, DATA will be clear to 0.	RW

The measured voltage V_{AUX} (V_{AUX1} and V_{AUX2}) is as follows:

$$V_{SAD} = \frac{ADATA}{4096} \bullet AVDADC$$

25.3.2.7 ADC Clock Divide Register (ADCLK)

The register ADCLK is used to set the A/D's clock dividing number.

	ADCLK																															BASE + 0x028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INUSG1																INUSG0								CLKDIV							
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	INUSG1	Inner usage, keep to 16'h4	RW
15:8	INUSG0	Inner usage, keep to 8'h10	RW
7:0	CLKDIV	Dividing number to get ADC clock from device clock. The A/D works at the frequency between 20Ksps and 200Ksps. If CLKDIV = N, Then the freq of adc_clk = dev_clk / (N+1). 0 ≤ N ≤ 255	RW

25.3.2.8 ADC Stable Register (ADCSTB)

ADC Stable register is used for insert some ADC clock cycle to wait ADC stable when sampling VBAT and AUX voltage.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ADCSTB																																BASE + 0x034
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																STB_DLY															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:0	STB_DLY	Stable delay cycle .	RW

25.4 SAR A/D Controller Guide

The following describes steps of using SAR-ADC.

25.4.1 Power Down Mode

1 Then initial value of ADENA.POWER is 1, and the state of SADC is in dower down state.

2 When you want to use SADC, you should first set ADENA.POWERON to 0 to power on SADC. And you should wait for at least 2ms, then you can enable VBAT or AUX.

3 When you want to power down SADC to get lower power, you should disable VBAT and AUX, and then set ADENA.POWER to 1.

25.4.2 VBAT Sample Operation

1 Set ADCLK.CLKDIV to set A/D clock frequency.

2 Set ADENA.VBATEN to 1 to enable the channel.

3 When ADSTATE.VRDY = 1, you can read the sample data from ADVDAT. And the VBATEN will be set to 0 auto.

25.4.3 AUX Sample Operation

1 Set ADCFG.CMD to choose one CMD. (AUX1 or AUX2)

2 Set ADCLK.CLKDIV to set A/D clock frequency.

3 Set ADENA.AUXEN to 1 to enable the channel.

4 When ADSTATE. ARDY = 1, you can read the sample data from ADADAT. And the AUXEN will be set to 0 auto.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

26 Real Time Clock

26.1 Overview

The Real-Time Clock (RTC) unit can be operated in either chip main power is on or the main power is down but the RTC power is still on. In this case, the RTC power domain consumes only a few micro watts power.

The RTC contains a real time and alarm logic, and the power down and wakeup control logic.

26.1.1 Features

RTC module has following features:

- Need input 32768Hz clock
- RTCLK selectable from the oscillator or from the divided clock of EXCLK, so that 32k crystal can be absent if the hibernating mode is not needed
- 32-bits second counter
- Programmable and adjustable counter to generate accurate 1 Hz clock
- Alarm interrupt, 1Hz interrupt
- Stand alone power supply, work in hibernating mode
- Power down controller
- Alarm wakeup
- External pin wakeup with up to 2s glitch filter

26.1.2 Signal Descriptions

RTC has 6 signal IO pins and 2 power pin. They are listed and described in.

Pin Names	Pin Loc	IO	IO Cell Char.	Pin Description	Power
RTCLK		AI	32768Hz	RTCLK: 32768Hz clock input	VDDRTC33
NC		AO		XRTCLK: Reserved NC: Not Connect	-
PWRON		O	8mA	PWRON: Power on/off control of main power	VDDRTC33
WKUP_		I	Schmitt	WKUP_: Wake signal after main power down	VDDRTC33
PPRST_		I	Schmitt	PPRST_: RTC power on reset and RESET-KEY reset input	VDDRTC33
VDDRTC33		P		VDDRTC33: 3.3V power for RTC and hibernating mode controlling that never power down (normally you can use 1.8V instead to reduce power consumption)	-

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Pin Names	Pin Loc	IO	IO Cell Char.	Pin Description	Power
VDDRTC11		P		VDDRTC11: 1.1V power for RTC core that never power down	-
CLK32K PD14		O IO	8mA pulldown-pe	32768Hz clock output PD14: GPIO group D bit14. When main power down, this pin is controlled by RTC register	VDDRTC33

RTCLK pin. need input 32.768KHz rtc clock. If do not use any clock, hibernate mode will be NOT available any more, and the time will lose if power down.

PWRON pin: this pin is used to control the main power on/off. Output high voltage means on and low voltage means off.

WKUP_ pin: hibernating mode wakeup input. (low active)

PPRST_ pin: This pin should be set to low voltage only in two cases.

- When RTC power is turned on. (so that whole chip is power on)
- A RESET-KEY is pressed.

CLK32PD14 pin: output 32.768KHz RTC clock and can used as GPIO.

26.2 Registers

26.2.1 Registers Memory Map

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 26-1 Registers Memory Map-Address Base

Name	Base	Description
RTC	0x10003000	Address base of RTC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 26-2 Registers for real time clock

Name	Description	Reset Value	Address offset	Access Size
RTCCR	RTC Control Register	0x000000??*1*2	0x000	32
RTCSR	RTC Second Register	0x????????	0x004	32
RTCSAR	RTC Second Alarm Register	0x????????	0x008	32
RTCGR	RTC Regulator Register	0x0???????	0x00C	32

NOTES:

1 *1: Unless otherwise stated, the reset value is for PPRST_ and Hibernating wakeup reset. WDT reset doesn't change the value.

2 *2: The reset value can be either of 0x00000081, 0x00000091, 0x00000089, 0x00000099.

Table 26-3 Registers for hibernating mode

Name	Description	Reset Value	Address offset	Access Size
HCR	Hibernate Control Register	0x00000000*1	0x020	32
HWFCR	Wakeup filter counter Register in Hibernate mode	0x0000???0	0x024	32
HRCR	Hibernate reset counter Register in Hibernate mode	0x00000??0	0x028	32
HWCR	Wakeup control Register in Hibernate mode	0x2d52d2d0*1	0x02C	32
HWRSR	Wakeup Status Register in Hibernate mode	0x00000???*1	0x030	32
HSPR	Scratch pattern register	0x????????	0x034	32
WENR	Write enable pattern register	0x00000000	0x03C	32
CKPCR	Configure the CLK32K pin value	0x00000000	0x040	32
WKUPPINCR	Configure the extend press WKUP pin to reset chip	0x00000064	0x048	32

NOTE:

*1: Unless otherwise stated, the reset value is for PPRST_ and Hibernating wakeup reset. WDT reset doesn't change the value.

All these registers, include those for real time clock and for hibernating mode control, except otherwise stated, are implemented in RTCLK clock domain. When write to these registers, it needs about 1 ~ 2 RTCLK cycles to actually change the register's value and needs another RTCLK cycle to allow the next write access. A bit RTCCR.WRDY is used to indicate it. When RCR.WRDY is 1, it

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

means the previous write is finished, a right value can be read from the target register, and a new write access can be issued. So before any write access, please make sure RCR.WRDY = 1.

26.2.2 Registers and Fields Description

26.2.2.1 RTC Control Register (RTCCR)

RTCCR contains bits to configure the real time clock features. Unless otherwise stated, the reset value is for PPRST_ and Hibernating wakeup reset. WDT reset doesn't change the value.

	RTCCR																															BASE + 0x000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								WRDY	1HZ	1HZIE	AF	AIE	AE	SELEXC	RTCE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1*[1]	0	0*[1]	?	0	?	0*[1]	1

NOTE:

*[1]: These bits are reset in all resets: PPRST_ input pin reset, hibernating reset and WDT reset.

Bits	Name	Description	RW
31:7	Reserved	Writing has no effect, read as zero.	R
7	WRDY	Write ready flag. It is 0 when a write is currently processing and the value has not been written to the writing target register. No write to any RTC registers can be issued in this case, or the result is undefined. The read value from the target register is also undefined. The reading is meaningful and another write can be issued when it is 1. This bit is read only and write to it is ignored.	R
6	1HZ	1Hz flag. This bit is set by hardware once every 1 second through the 1Hz pulse if the real time clock is enabled (RTCCR.RTCE = 1). This bit can be cleared by software. Write 1 to this bit is ignored.	RW
5	1HZIE	1Hz interrupt enable. Writing to this bit takes effect immediately without delay. **1HZIE** — **Description** 0 — 1Hz interrupt is disabled. 1 — 1Hz interrupt is enabled. RTC issues interrupt when 1HZ bit is set.	RW
4	AF	Alarm flag. This bit is set by hardware when alarm match (RTCSR = RTCSAR) is found and alarm is enabled (RTCCR.AE = 1) and the real time clock is enabled (RTCCR.RTCE = 1). This bit can be cleared by software. Write 1 to this bit is ignored. Writing to this bit takes effect immediately.	RW
3	AIE	Alarm interrupt enable.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			AIE / **Description** 0 / Alarm interrupt is disabled. 1 / Alarm interrupt is enabled. RTC issues interrupt when AF is set.	
2	AE	Alarm enable. **AE** / **Description** 0 / Alarm function is disabled. 1 / Alarm function is enabled.		RW
1	SELEXC	The divided EXCLK is selected as RTCLK in rtc-hiber module. **SELEXC** / **Description** 0 / OSC32K or RTCLK input clock is selected as RTCLK in rtc-hiber module. 1 / The divided EXCLK is selected as RTCLK in rtc -hiber module. NOTE: If do not use any 32Khz clock (either input clock or using crystal), hibernate mode will be NOT available any more, and the time will lose if power down. CPM.OPCR.ERCS must be 0, when using SELEXC = 1. When the main chip power down, SELEXC will be 0 in internal circuit, in this time, RTCLK will use OSC32K clock.		RW
0	RTCE	Real time clock enable. **RTCE** / **Description** 0 / Real time clock function is disabled. 1 / Real time clock function is enabled.		RW

26.2.2.2 RTC Second Register (RTCSR)

RTCSR is a 32-bit width second counter. It can be read and write by software. It is increased by 1 at every 1Hz pulse if the real time clock is enabled (RTCCR.RTCE = 1). When read, it should be read continued more than once and take the value if the adjacent results are the same. RTCSR is not initialized by any reset.

	RTCSR																															**BASE + 0x004**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTCSR																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

26.2.2.3 RTC Second Alarm Register (RTCSAR)

RTCSAR serves as a second alarm register. Alarm flag (RTCCR.AF) is set to 1 when the RTCSR equals the RTCSAR in the condition of alarm is enabled (RTCCR.AE = 1) and the real time clock is

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

enabled (RTCCR.RTCE = 1). RTCSAR can be read and write by software and is not initialized by any reset.

	RTCSAR																															BASE + 0x008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTCSAR																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

26.2.2.4 RTC Regulator Register (RTCGR)

RTCGR is serves as the real time clock regulator, which is used to adjust the interval of the 1Hz pulse.

	RTCGR																															BASE + 0x00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK	Reserved					ADJC										NC1HZ															
RST	0*[1]	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

NOTE:

*[1]: This bit is reset in all resets: PPRST_ input pin reset, hibernating reset and WDT reset.

Bits	Name	Description	RW
31	LOCK	Lock bit. This bit is used to safeguard the validity of the data written into the RTCGR register. Once it is set, write to RTCGR is ignored. This bit can only be set by software and cleared by (any type of) resets. **LOCK** / **Description** 0 / Write to RTCGR is allowed. 1 / Write to RTCGR is forbidden.	RW
30:26	Reserved	Writing has no effect, read as zero.	R
25:16	ADJC	This field specifies how many times it needs to add one 32kHz cycle for the 1Hz pulse interval in every 1024 1Hz pulses. In other words, among every 1024 1Hz pulses, ADJC number of them are trigged in every (NC1HZ + 2) 32kHz clock cycles, (1024 – ADJC) number of them are trigged in every (NC1HZ + 1) 32kHz clock cycles.	RW
15:0	NC1HZ	This field specifies the number plus 1 of the working 32kHz clock cycles are contained in the 1Hz pulse interval. In other words, 1Hz pulse is trigged every (NC1HZ + 1) 32kHz clock cycles, if RTCGR.ADJC = 0.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.2.2.5 Hibernate Control Register (HCR)

HCR contains the bit to control the main chip power on/off. Unless otherwise stated, the reset value is for PPRST_ and Hibernating wakeup reset.

	HCR																															BASE + 0x020
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															PD
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:1	Reserved	Writing has no effect, read as zero.	R
0	PD	Power down or power on bit. Besides writing by CPU, this bit will be set to 1 if an unknown reason main power supply off is detected. This bit controls the PWRON pin level. When co-working with some external components, this bit is used for power management of this chip. It is supposed when 1 is written to this bit, the main power supply of the chip, except RTC power, will be shut down immediately. After this bit is set to 1, all registers in RTC module, except RTCCR.1HZ and RTCCR.1HZIE, cannot be changed by write access. This bit is cleared by reset pin reset and hibernating reset. The later one is asserted by wakeup procedure. PD | PWRON | Description 0 | VDDRTC33 | No power down, keep power on. 1 | 0 V | Power down enable, turn power off.	RW

26.2.2.6 HIBERNATE mode Wakeup Filter Counter Register (HWFCR)

The HIBERNATE mode Wakeup Filter Counter Register (HWFCR) is a 32-bit read/write register .It filter the glitch generated by a dedicated wakeup pin. The HWFCR is not initialized by any reset.

	HWFCR																															BASE + 0x024
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																HWFCR											Reserved				
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:5	HWFCR	Wakeup pin effective minimum time in number of 32 RTCLK cycles, used as glitch filter logic. Maximum of 2 seconds if the RTCLK is 32768Hz	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		If this value is configured to 0, and the WKUP pin keeps low longer than 15 RTCLK periods, it will wakes up RTC from Hibernate. The glitch filter time equals to HWFCR*32 RTCLK period.	
4:0	Reserved	Writing has no effect, read as zero.	R

26.2.2.7 Hibernate Reset Counter Register (HRCR)

The Hibernate Reset Counter Register is a 32-bit read/write register that specifies hibernate reset assertion time. The HRCR is initialized by PPRST_ .

	HRCR																															BASE + 0x028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																	HRCR				Reserved										
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:15	Reserved	Writing has no effect, read as zero.	R
14:11	HRCR	HIBERNATE Reset asserting time. Number of 2048 RTCLK cycles. Default value 0x1, assert 125 ms. Maximum 1 second if the RTCLK is 32768Hz. If this value is configured to 0, it will generate 2048 RTCLK HIBERNATE Reset, about 62.5 milliseconds.	RW
10:0	Reserved	Writing has no effect, read as zero.	R

26.2.2.8 HIBERNATE Wakeup Control Register (HWCR)

The HIBERNATE Wakeup Control Register is a 32-bit read/write register that controls real time clock alarm wake up enable. The reset value only for PPRST_ .

	HWCR																															BASE + 0x02C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDET																													Reserved		EALM
RST	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	0	0*[1]

NOTE:

*[1]: This bit is reset in PPRST_ input pin reset and hibernating reset .

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:3	EPDET	Power detect enable. 0x5aa5a5a: enable(default) 0x1a55a5a5: disable Only 0x1a55a5a5 can disable power detect, other values will enable the power detect.	RW
2:1	Reserved	Writing has no effect, read as zero.	R
0	EALM	RTC Alarm wakeup enable. 0: disable 1: enable	RW

26.2.2.9 HIBERNATE Wakeup Status Register (HWRSR)

The HIBERNATE Wakeup Status Register is a 32-bit read/write register that reflects wakeup status bits.

	HWRSR																															BASE + 0x030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																							APD	Reserved		HR	PPR	Reserved		PIN	ALM
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	0	0	1*[2]	1*[1]	0	0	?	?

NOTES:

*[1]: This reset value only for PPRST_. It is undefined in case of other resets.

*[2]: This reset value only for HRST_. It is undefined in case of other resets.

Bits	Name	Description	RW
31:9	Reserved	Writing has no effect, read as zero.	R
8	APD	Accident power down. When the software has not set to HIBERNATE state, the core power is down, then an accident power down is detected. APD is set and remains set until software clears it. This bit can only be written with 0. Write with 1 is ignored. **HR** — **Description** 0 — Accident power down has not occurred since the last time the software clears this bit. 1 — Accident power down has occurred since the last time the software clears this bit.	RW
7:6	Reserved	Writing has no effect, read as zero.	R
5	HR	Hibernate Reset. When a Hibernate reset detected, HR is set and remains set until software clears it. This bit can only be written with 0. Write with 1 is ignored. **HR** — **Description** 0 — Hibernate reset has not occurred since the last time the	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			software clears this bit.	
		1	Hibernate reset has occurred since the last time the software clears this bit.	
4	PPR	PAD PIN Reset. When a PPRST_ is detected, PPR is set and remains set until software clears it. This bit can only be written with 0. Write with 1 is ignored.		RW
		PPR	**Description**	
		0	PPRST_ reset has not occurred since last time the software clears this bit.	
		1	PPRST_ reset has occurred since last time the software clears this bit.	
3:2	Reserved	Writing has no effect, read as zero.		R
1	PIN	Wakeup Pin Status bit. The bit is cleared when chip enters hibernating mode. It is set when exit the hibernating mode by wakeup pin. This bit can only be written with 0. Write with 1 is ignored.		RW
0	ALM	RTC Alarm Status bit. The bit is cleared when chip enters hibernating mode. It is set when exit the hibernating mode by alarm. This bit can only be written with 0. Write with 1 is ignored.		RW

26.2.2.10 Hibernate Scratch Pattern Register (HSPR)

This is a scratch register used to hold a pattern. The software can check the pattern is kept to know whether RTC power has ever been down and whether it is needed to setup the real time clock.

	HSPR																															**BASE + 0x034**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	**Name**	**Description**	**RW**
31:0	PAT	The pattern.	RW

26.2.2.11 Write Enable Pattern Register (WENR)

This is a write protect register to may the rtc registers can write only in certain condition.

	WENR																															**BASE + 0x03C**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WEN	Reserved															WENPAT															
RST	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	WEN	The write enable flag. If the WENPAT is 0xA55A then this bit will be 1. When the WEN changes to 1, the RTCCR, RTCSR, RTCSAR, RTCGR, HCR, HWFCR, HRCR, HWCR, HWRSR, HSPR registers could be changed. But RTCCR.SELEXC, RTCCR.HZIE, RTCCR.WRDY may change in any time. This bit is read only and write to it is ignored. This bit only reset by PPRST_ and HRST_. There is an exception, when system does NOT have RTC 32Khz crystal. MUST write 1 to RTCCR.SELEXC before write to any value to any other registers. <table><tr><th>WEN</th><th>Description</th></tr><tr><td>0</td><td>Other RTC registers is locked, write these registers will be ignored.</td></tr><tr><td>1</td><td>Other RTC registers can be changed.</td></tr></table>	R
30:16	Reserved	Writing has no effect, read as zero.	R
15:0	WENPAT	The write enable pattern. Before writing any value to RTCCR, RTCSR, RTCSAR, RTCGR, HCR, HWFCR, HRCR, HWCR, HWRSR, HSPR registers, write 0xA55A to WENPAT to set these register writable. If this value is ok, WEN will change to 1. But RTCCR.SELEXC and RTCCR.HZIE are writable in any time. These bits are write-only, always read as 0.	W

26.2.2.12 CLK32K Pin control register (CKPCR)

This is a CLK32K pin control register used to configure the CLK32K pin value. The CKPCR is initialized by PPRST_ .

	CKPCR																															BASE + 0x040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																										CK32RD	CK32PULL	Reserved	CK32CTL		CK32D
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	0	0	0	0	0

Bits	Name	Description	RW
31:3	Reserved	Writing has no effect, read as zero.	R
5	CK32RD	Read this bit will return CLK32K pin status.	R
4	CK32PULL	Pull up configures. <table><tr><th>CK32PULL</th><th>Description</th></tr><tr><td>0</td><td>Pull Down is enabled.</td></tr></table>	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1	Pull Down is disabled.	
3	Reserved	Writing has no effect, read as zero.		R
2:1	CK32CTL	Output RTCLK to CLK32K pin.		RW
		CK32CTL	**Description**	
		00	CLK32K pin is set to general input. The pin value can be read by CK32RD bit or GPIO PD14 bit. The input pin should not be left floating, if pull up (CK32PULL) is disabled.	
		01	CLK32K pin is set to general output. The pin output value is set by CK32D bit.	
		10	GPIO PD14 controls CLK32K pin. The pin output is set by GPIO PD14 bit. The CLK32K pin is output CK32D bit in HIBERNATE mode.	
		11	Output RTCLK to CLK32K pin. If this set, CLK32K pin will always output 32K clock, even in HIBERNATE mode.	
0	CK32D	When CK32CTL is configured to general output, Write to this pin will output to CLK32K pin. And if CK32CTL configured to GPIO controls, the CK32D value will output at CLK32K pin in hibernate mode.		RW

26.2.2.13 WKUP_PIN_RST control register (WKUPPINCR)

This is a register used to control the enable and set the judge time to reset the chip when extend press the WKUP pin. The are only initialized by PPRST_ and HRST_.

	WKUPPINCR																															**BASE + 0x048**
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								P_JUD_LEN				P_RST_EN			
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0

Bits	Name	Description	RW
31:8	Reserved	Writing has no effect, read as zero.	R
7:4	P_JUD_LEN	WKUP up pin extend press length judge value The judge time equals to FLT_LEN multiply RTC 1HZ period, and only the active press length equal or greater than judge time, the chip reset can assert.	RW
3: 0	P_RST_EN	Enable extend press WKUP pin to reset chip others: disable	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		4'hB: enable	

26.3 Time Regulation

Because of the inherent inaccuracy of crystal and other variables, the time counter may be inaccurate. This requires a slight adjustment. The application processor, through the RTCGR, lets you adjust the 1Hz time base to an error of less than 1ppm. Such that if the Hz clock were set to be 1Hz, there would be an error of less than 5 seconds per month.

To determine the value programmed into the RTCGR, you must first measure the output frequency at the oscillator multiplex (approximately 32 kHz) using an accurate time base, such as a frequency counter. This clock is externally visible by selecting the alternate function of GPIO.
To gain access to the clock, program this pin as an output and then switch to the alternate function.
To trim the clock, divide the output of the oscillator by an integer value and fractional adjust it by periodically deleting clocks from the stream driving this integer divider.

After the true frequency of the oscillator is known, it must be split into integer and fractional portions. The integer portion of the value (minus one) is loaded into the NC1HZ field of the RTCGR.

The fractional part of the adjustment is done by periodically deleting clocks from the clock stream driving the Hz divider. The trim interval period is hardwired to be 1024 1Hz clock cycles (approximately 17 minutes). The number of clocks (represented by ADJC field of RTCGR) are deleted from the input clock stream per trim interval. If ADJC is programmed to be zero, then no trim operations occur and the RTC is clocked with the raw 32 kHz clock. The relationship between the Hz clock frequency and the nominal 32 kHz clock (f1 and f32K, respectively) is shown in the following equation.

$$f1 = \frac{2^{10} \times (NC1HZ + 1)}{2^{10} \times (NC1HZ + 1) + ADJC} \times \frac{f32k}{NC1HZ + 1}$$

f1 = actual frequency of 1Hz clock
f32k = frequency of either 32.768KHz crystal output or 3.6864MHz crystal output further divided down to 32.914KHz

26.4 Operate Mode

26.4.1 Registers Access

26.4.1.1 Registers Read

All registers can read directly, but the RTCSR and RTCSAR should be read continued more than once and take the value if the adjacent results are the same. Please check and make sure RTCCR.WRDY equals to 1 before execute a register read.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.4.1.2 Registers Write

Only the RTCCR.SELEXC, RTCCR.HZIE, RTCGR.LOCK, WENR.WENPAT bits can write directly. Others bits need following the below procedure

1) Check RTCCR.WRDY, make sure it equals to 1
2) Write 0xa55a to WENR register to enable other registers write.
3) Waiting WENR.WEN change to 1
4) Check RTCCR.WRDY, make sure it equals to 1
5) Write the target register
6) Waiting RTCCR.WRDY equals to 1 to make sure the current write processing was finished.

26.4.2 Normal Mode

26.4.2.1 Power Detect

When PPRST_ reset, the HWCR.EPDET set to 0x5aa5a5a, and the power detect is enable. If you want To disable the power detect, you should set HWCR.EPDET to 0x1a55a5a5. If power detect is enable, when CORE power supply was lose, the RTC will into hibernate mode. And if power detect is not enable, the RTC can't enter hibernate mode even CORE power supply was lose.

26.4.2.2 Power On Timing Diagram

The RTC power on procedure like the following figure.

Figure 26-1 Core Power On

26.4.3 HIBERNATE Mode

The purpose of hibernate mode is to save the power. When in hibernate mode, CORE will no power consumption.

If you want to use hibernate mode, please first make sure RTCCR.SELEXC is 0.
When Software writes 1 to PD bit of HCR, the system at once enters HIBERNATE mode. The powers of CORE and IO are disconnected by PWRON pin, no power consumption in core and IO. When a wakeup event occurs, the core enters normal mode through a hibernate reset.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.4.3.1 Procedure to Enter HIBERNATE mode

Before enter HIBERNATE mode, software must complete following steps:

1. Finish the current operation and preserve all data to flash.
2. Configure the wake-up sources properly by configure HWCR.
3. Set HIBERNATE MODE. (Set PD bit in HCR to 1)

26.4.3.2 Procedure to Wake-up from HIBERNATE mode

1. The internal hibernate reset signal will be asserted if one of the wake-up sources is issued.
2. Check RSR to determine what caused the reset.
3. Check PIN/ALM bits of HWRSR in order to know whether or not the power-up is caused by which wake-up from HIBERNATE mode.
4. Configure the SDRAM memory controller.
5. Recover the data from flash.

26.5 Clock select

There could be two clock input to RTC internal clock called rtclk. One is OSC32k clock; the other is EXCLK/512.

The software MUST make sure the RTC run in valid clock configuration.

Table 26-4 Clock select registers

RTCCR.SELEXC	CPM.ERCS	Description	Valid
0	0	RTC use OSC32K clock.	OK
0	1		OK
1	0	RTC use EXCLK/512 clock.	OK
1	1	RTC will lost clock. (Not Valid)	NO

Figure 26-2 RTC clock selection path

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Changing RTCLK sequence:

1. There are both 32KHz crystal and 24Mhz EXCLK crystal connected, so RTCLK input path has 32Khz clock.
   In this case, there is no need to change internal clock, so do NOT change SELEXC all the time.
2. There is no 32KHz crystal connected but only 24Mhz EXCLK crystal connected, so RTCLK input path has no clock.
   In this case, should flow the sequence below to change internal clock:
   a. Set OPCR.ERCS of CPM to 1; close EXCLK/512 to RTC.
   b. Set CLKGR.RTC of CPM to 0; open PCLK to RTC.
   c. Set RTCCR.SELEXC to 1; change internal clock to EXCLK/512.
   d. Wait two clock period of clock.
   e. Clear OPCR.ERCS of CPM to 0; open EXCLK/512 to RTC.
   f. Configure all RTC registers but RTCCR.SELEXC.
   g. Check RTCCR.SELEXC == 1.
   h. IF YES, finish this sequence; IF NO, do step (a) again.

NOTE: If using HIBERNATE mode, MUST have both 32KHz crystal (or input 32Khz clock) and 24Mhz EXCLK crystal connected, or RTC time will be insignificant.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

27 EFUSE Slave Interface (EFUSE)

27.1 Overview

The EFUSE is provided 4K programmable bit, Total 4K bits are separated into thirteen segments, as below table show.

128bit	128bit	128bit	32bit	32bit	32bit	16bit	16bit	128bit	128bit	128bit	128bit	3072bit

Each segment can be programmed separately.
Each segment has a protect bit, which has higher priority than program segment selection.

Programming frequency is wide, and programming time can adjust use the EFUSE register.

Initial value of EFUSE is 0, when programmed to 1, it doesn't able to program back to 0 anymore.
Do not attempt to program any bit that already programmed to 1, such action will result unpredictable status to whole EFUSE block.

Programming voltage supply pin VDDQ:

- In program mode, supply VDDQ with 2.5V.
  Maximum accumulative time for VDDQ pin exposed under 2.5V+/-10% should be less than 1 sec.
- In read mode, leave VDDQ to 0V or floating.

27.2 Registers

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SPEC - special access method, relate to its description

3. Reset Value

1 - reset to 1

0 - reset to 0

? - value unknown after reset

27.2.1 Registers Memory Map

Table 27-1 Registers Memory Map-Address Base

Name	Base	Description
EFUSE	0x13540000	Address base of EFUSE

Table 27-2 EFUSE Register Description

Name	Description	Reset Value	Address Offset	Access Size
EFUCTRL	EFUSE Control Register	0x001F0000	0x0000	32
EFUCFG	EFUSE Configure Register	0x00240000	0x0004	32
EFUSTATE	EFUSE Status Register	0x00????00	0x0008	32
EFUDATA0	EFUSE Data 0 Register	0x0000????	0x000C	32
EFUDATA1	EFUSE Data 1 Register	0x00000000	0x0010	32
EFUDATA2	EFUSE Data 2 Register	0x00000000	0x0014	32
EFUDATA3	EFUSE Data 3 Register	0x00000000	0x0018	32
EFUDATA4	EFUSE Data 4 Register	0x00000000	0x001C	32
EFUDATA5	EFUSE Data 5 Register	0x00000000	0x0020	32
EFUDATA6	EFUSE Data 6 Register	0x00000000	0x0024	32
EFUDATA7	EFUSE Data 7 Register	0x00000000	0x0028	32

27.2.2 Registers and Fields Description

27.2.2.1 EFUSE Control Register (EFUCTRL)

	EFUCTRL																															BASE + 0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved		Address									Length					PG_EN	Reserved													WR_EN	RD_EN
RST	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:21	Address	Indicate the start address wants to program or read Aligned to Byte	RW
20:16	Length	Indicate the length wants to program or read, efuse operate length equals to this value add 1. Byte number(min: 1byte, max: 32byte, default: 32byte)	RW
15	PG_EN	Programming EFUSE enable 1: Enable 0: Disable	RW
14:2	Reserved	Writing has no effect, read as zero.	RW
1	WR_EN	Write data to EFUSE data buffer enable 1: Enable 0: Disable This bit will change to 0 automatically when write procedure done	RW
0	RD_EN	Read data to EFUSE data buffer enable 1: Enable 0: Disable This bit will change to 0 automatically when read procedure done	RW

27.2.2.2 EFUSE Configure Register (EFUCFG)

	EFUCFG																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INT_EN	Reserved							RD_ADJ				RD_STROBE				WR_ADJ				WR_STROBE											
RST	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	INT_EN	Interrupt enable when read/write EFUSE done. 1: Enable 0: Disable	RW
30:25	Reserved	Writing has no effect, read as zero.	R
23:20	RD_ADJ	Adjust number of h2clk cycles when EFUSE reading unsigned value, h2clk period multiply (RD_ADJ + 1) should greater than 6.5 ns.	RW
19:16	RD_STROBE	Adjust number of h2clk cycles when EFUSE reading STROBE Signed value, h2clk period multiply(RD_ADJ + 5 + RD_STROBE) should greater than 35 ns, default 4.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:12	WR_ADJ	Adjust number of h2clk cycles when EFUSE programming unsigned value, h2clk period multiply (WR_ADJ + 1) should greater than 6.5 ns.	RW
11:0	WR_STROBE	Adjust number of h2clk cycles when EFUSE programming STROBE Signed value, h2clk period multiply(WR_ADJ + 1666 + WR_STROBE) should between 9us and 11us, typical 10us	RW

27.2.2.3 EFUSE Status Register (EFUSTATE)

	EFUSTATE																															BASE + 0x0008
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved								UK_PRT	MD5_PRT	AU_DET	INST_PRT	TM3_PRT	WDT_EN	TM12_PRT	TM0_PRT	CUT_PRT	CHID_PRT	Reserved		NENF_SEL	DIS_JTAG	AUBT_EN	SCBT_EN	Reserved						WR_DONE	RD_DONE
RST	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	0	0	?	?	?	?	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
30:24	Reserved	Writing has no effect, read as zero.	R
23	USERKEY_PRT	User key segment write protect 1: protect, this segment can't write 0: no protect, this segment can write	R
22	MD5_PRT	MD5 segment write protect 1: protect, this segment can't write 0: no protect, this segment can write	R
21	AUTO_DET_EN	Boot Rom auto detect external clock frequency enable 1: enable auto detect 0: disable auto detect	R
20	INST_PRT	CPU instructions segment write protect 1: protect, this segment can't write 0: no protect, this segment can write	R
19	TRIM3_PRT	Trim3 segment write protect bit 1: protect, this segment can't write 0: no protect, this segment can write	R
18	WDT_EN	Enable WDT reset function in Boot Rom 1: enable 0: disable	R
17	TRIM1_2_PRT	Trim1 and Trim2 segment write protect bit 1: protect, this segment can't write 0: no protect, this segment can write	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16	TRIM0_PRT	Trim0 segment write protect bit 1: protect, this segment can't write 0: no protect, this segment can write	R
15	CUSTID_PRT	Customer ID segment write protect bit 1: protect, this segment can't write 0: no protect, this segment can write	R
14	CHIPID_PRT	CHIP ID segment protect bit 1: protect, this segment can't write 0: no protect, this segment can write	R
13:12	Reserved	Writing has no effect, read as zero.	R
11	NEMC_NFI_SEL	NEMC and NFI controller select *Please refer NEMC and NFI spec for detail	R
10	DIS_JTAG	Disable JTAG 1: disable JTAG 0: enable JTAG	R
9:	CDCBOOT_EN	CDC boot enable 1: enable CDC boot 0: disable CDC boot	R
8	SECBOOT_EN	Security boot enable 1: enable Security boot 0: Normal boot	R
7:2	Reserved	Writing has no effect, read as zero.	RW
1	WR_DONE	Write data to EFUSE operate done flag 1: done 0: not done *This bit only can write to 0, and write to 0 will clear this flag.	RW
0	RD_DONE	Read EFUSE data operate done flag 1: done 0: not done *This bit only can write to 0, and write to 0 will clear this flag	RW

27.2.2.4 EFUSE Data Register (EFUDATAn)

	EFUDATAn (n=0,1,2,3,4,5,6,7)																BASE + 0x000C ~ BASE + 0x0028															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	DATA	EFUSE DATA register. *EFUSE DATA0 register mapped low operate address, others mapped increased address.	RW

NOTE:

1. The EFUSE 4K bits are mapped to chip memory address, from 0x13540200 to 0x135403FF.

2. The EFUSE 4K programmable bits are separated into thirteen segments as below table. The first Segment used to store Ingenic chip id, second segment store 128bit chip number, third segment used to store customer id, forth to seventh segment reserved to store trim data, eighth segment store segment protect bit, ninth segment used to store root key, tenth segment used to store chip key, eleven used to sore user key, twelve used to store md5, the last segment used for Ingenic to program instructions and fix boot rom.

Segment	1	2	3	4	5	6	7	8	9	10	11	12	13
Mapped Address	200 | 20F	210 | 21F	220 | 22F	230 | 233	234 | 237	238 | 23B	23C | 23D	23E | 23F	240 | 24F	250 | 25F	260 | 26F	270 | 27F	280 | 3FF
Bits	128	128	128	32	32	32	16	16	128	128	128	128	3072

3. The protect bit segment store others segment write protect bit, if protect bit programmed to 1, corresponding segment can not be able to program. The protect bit map as below table

	Bit	Mapped To EFUSE Status Register
The Protect bit Segment	15	USERKEY_PRT
	14	MD5_PRT
	13	AUTO_DET_EN
	12	INST_PRT
	11	TRIM3_PRT
	10	WDT_EN
	9	TRIM_1_2_PRT
	8	TRIM0_PRT
	7	USERID_PRT
	6	CHIPID_PRT
	5	Reserved, need fix to 0
	4	Reserved, need fix to 0
	3	NEMC_NFI_SEL
	2	DISJTAG
	1	CDCBOOT_EN
	0	SECBOOT_EN

4. The EFUSE can read or write maximum 256bit in each time, but we strongly recommend

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

operate each segment separately.

5. The segment(Root key/Chip key/User key/MD5) can only read/write by SC-ROM.
6. Trim1 and Trim2 segment shared one protect bit, named TRIM_1_2_PRT.
7. The Trim3 segment used to store SARADC calibration data, so please don't program these addresses.

27.3 Flow

27.3.1 Program EFUSE Flow

1. Set EFUCFG register(reading and programming timing).
2. Write want program data to EFUSE data buffer 0-7 registers
3. Set control register, indicate want to program address, data length.
4. Write control register PG_EN bit to 1
5. Connect VDDQ pin to 2.5V
6. Write control register WR_EN bit to 1
7. Wait status register WR_DONE set to 1.
8. Disconnect VDDQ pin from 2.5V.
9. Write control register PG_EN bit to 0.

27.3.2 Program Security Key Flow

1. Set EFUCFG register(reading and programming timing).
2. Write control register PG_EN bit to 1
3. Connect VDDQ pin to 2.5V
4. Invoke SC-ROM controller to write data to Security key mapped address
5. Wait status register WR_DONE set to 1
6. jump to step 4, program all data like this
7. Disconnect VDDQ pin from 2.5V.
8. Write control register PG_EN bit to 0.

NOTE:

*[1]: Do NOT pre-charge VDDQ too early before EFUSE program started.
The maximum 2.5V supply time to VDDQ must be strictly controlled less than 1sec.
Use on-chip counter and GPIO to coordinate external supply source.

*[2]: Only the 1.3.2 flow can be used to program the User Key and md5 Segment, The Root key and Chip key Segment will be programmed at CP by Ingenic, other segments only can program use 1.3.1 flow.

*[3]: The SC-ROM Controller can write Key segment 32bits once time.

27.3.3 Read EFUSE Flow

1. Set EFUCFG register(reading and programming timing).
2. Set control register to indicate what to read data address, read data length.
3. Set read enable register.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4. Wait status register RD_DONE set to 1 or EFUSE interrupted, software application can read EFUSE data buffer 0 – 7 registers.

27.3.4 Read Security Key/Random Number Flow

1. Invoke SC-ROM controller to read corresponding map address.

NOTE:

*[1]: EFSUE controller will read EFUSE protect bit segment automatically when chip reset, and update the EFUSE state register.

*[2]: Only 1.3.4 flow can be used to read all Keys, other segments only can read use 1.3.3 flow.

*[3]: The SC-ROM Controller can read Keys value 32bits once time.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 6

PERIPHERALS

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28 General-Purpose I/O Ports

28.1 Function Overview

General Purpose I/O Ports (GPIO) is used in generating and capturing application-specific input and output signals. Each port can be programmed as an output, an input or function port that serves certain peripheral. As input, pull up/down can be enabled/disabled for the port and the port also can be configured as level or edge tripped interrupt source.

Features:

- Each port can be configured as an input, an output or an alternate function port
- Each port can be configured as an interrupt source of low/high level or rising/falling edge triggering. Every interrupt source can be masked independently
- Each port has an internal pull-up or pull-down resistor connected. The pull-up/down resistor can be disabled
- GPIO output 7 interrupts, 1 for every group, to INTC

28.1.1 GPIO Port A Summary

Table 28-1 GPIO Port A summary

Bit	PA	Pull	Shared Function Port Selected by				
N	N	(U/D)	0	1	2	3	Note
0	00	U	sd0(io)	-	-	-	
1	01	U	sd1(io)	-	-	-	
2	02	U	sd2(io)	-	-	-	
3	03	U	sd3(io)	-	-	-	
4	04	U	sd4(io)	msc0_d4(io)	-	-	
5	05	U	sd5(io)	msc0_d5(io)	-	-	
6	06	U	sd6(io)	msc0_d6(io)	-	-	
7	07	U	sd7(io)	msc0_d7(io)	-	-	
8	08	U	sd8(io)	-	-	-	
9	09	U	sd9(io)	-	-	-	
10	10	U	sd10(io)	-	-	-	
11	11	U	sd11(io)	-	-	-	
12	12	U	sd12(io)	-	-	-	
13	13	U	sd13(io)	-	-	-	
14	14	U	sd14(io)	-	-	-	
15	15	U	sd15(io)	-	-	-	
16	16	U rst-pe	rd_o(o)	smb2_sda(io)	-	-	
17	17	U	we_o(o)	smb2_sck_o(o)	-	-	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		rst-pe					
18	18	U rst-pe	fre_o(o)	msc0_clk_o(o)	ssi0_clk(io)	-	
19	19	U rst-pe	fwe_o(o)	msc0_cmd(io)	ssi0_dt(io)	-	
20	20	U	-	msc0_d0(io)	ssi0_dr(io)	frb0_i(i)	1, ***13***
21	21	U rst-pe	cs1_o(o)	msc0_d1(io)	ssi0_gpc_o(io)	-	
22	22	U rst-pe	cs2_o(o)	msc0_d2(io)	ssi0_ce1_o(io)	-	
23	23	U rst-pe	cs3_o(o)	msc0_d3(io)	ssi0_ce0(io)	frb1_i(i)	***13***
24	24	U rst-pe	cs4_o(o)	-	-	-	
27	27	U	wait_i(i)	-	-	-	
29	29	U	ndqs(io)	-	-	-	
30	30	-	-	-	-	-	6
31	31	-	-	-	-	-	7,11

28.1.2 GPIO Port B Summary

Table 28-2 GPIO Port B summary

Bit N	PB N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	D rst-pe	sa0_o(o)	-	-	-	8
1	01	D rst-pe	sa1_o(o)	-	-	-	9
2	02	U	sa2_o(o)	-	uart4_rxd_i(i)	-	
3	03	U	sa3_o(o)	-	uart4_cts_i(i)	-	
4	04	U	sa4_o(o)	-	uart4_rts_o(o)	-	
5	05	U	sa5_o(o)	-	uart4_txd_o(o)	-	
7	07	U	-	smb3_sda(io)	isp_sda(io)	-	
8	08	U	-	smb3_sck(io)	isp_sck(io)	-	
20	20	U	ssi0_dr_i(i)	bclk(io)	uart4_rxd_i(i)	msc2_d0(io)	
21	21	U	ssi0_dt_o(o)	lrclk(io)	uart4_txd_o(o)	msc2_d1(io)	***13***
28	28	U	ssi0_clk_o(o)	sdati_i(i)	uart4_rts_o(o)	msc2_clk_o(o)	
29	29	U rst-pe	ssi0_ce0_o(o)	sdato_o(o)	uart4_cts_i(i)	msc2_cmd_o(o)	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30	30	U	ssi0_gpc_o(o)	sdato1_o(o)	-	msc2_d2(io)	*13*
31	31	U rst-pe	ssi0_ce1_o(o)	sysclk_o(o)	-	msc2_d3(io)	*13*

28.1.3 GPIO Port C Summary

Table 28-3 GPIO Port C summary

Bit N	PC N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	U	lcd_b0_o(o)	lcd_rev_o(o)	-	-	
1	01	U	lcd_b1_o(o)	lcd_ps_o(o)	-	-	
2	02	U	lcd_b2_o(o)	cim0_d8(io)	slcd_d0_o(o)	epd_d0_o(o)	
3	03	U	lcd_b3_o(o)	cim0_d9(io)	slcd_d1_o(o)	epd_d1_o(o)	
4	04	U	lcd_b4_o(o)	-	slcd_d2_o(o)	epd_d2_o(o)	
5	05	U	lcd_b5_o(o)	-	slcd_d3_o(o)	epd_d3_o(o)	
6	06	U	lcd_b6_o(o)	-	slcd_d4_o(o)	epd_d4_o(o)	
7	07	U	lcd_b7_o(o)	-	slcd_d5_o(o)	epd_d5_o(o)	
8	08	U	lcd_pclk_o(o)	-	slcd_d6_o(o)	epd_d6_o(o)	
9	09	U	lcd_de_o(o)	-	slcd_d7_o(o)	epd_d7_o(o)	
10	10	U rst-pe	lcd_g0_o(o)	lcd_spl_o(o)	uart2_txd_o(o)	-	
11	11	U	lcd_g1_o(o)	-	-	-	
12	12	U	lcd_g2_o(o)	cim0_d0(io)	slcd_d8_o(o)	epd_gdclk_o(o)	
13	13	U	lcd_g3_o(o)	cim0_d1(io)	slcd_d9_o(o)	epd_gdsp_o(o)	
14	14	U	lcd_g4_o(o)	cim0_d2(io)	slcd_d10_o(o)	epd_gdoe_o(o)	*13*
15	15	U	lcd_g5_o(o)	cim0_d3(io)	slcd_d11_o(o)	epd_sdclk_o(o)	*13*
16	16	U	lcd_g6_o(o)	cim0_d4(io)	slcd_d12_o(o)	epd_sdoe_o(o)	*13*
17	17	U	lcd_g7_o(o)	cim0_d5(io)	slcd_d13_o(o)	epd_sdle_o(o)	*13*
18	18	U	lcd_hsyn(io)	cim0_d6(io)	slcd_d14_o(o)	epd_sdce0_o(o)	*13*
19	19	U	lcd_vsyn(io)	cim0_d7(io)	slcd_d15_o(o)	epd_sdce1_o(o)	*13*
20	20	U	lcd_r0_o(o)	lcd_cls_o(o)	uart2_rxd_i(i)	-	
21	21	U	lcd_r1_o(o)	-	-	-	
22	22	U	lcd_r2_o(o)	smb3_sck(io)	isp_rst_o(o)	epd_pwr0_o(o)	
23	23	U	lcd_r3_o(o)	smb3_sda(io)	isp_pwd_o(o)	epd_pwr1_o(o)	
24	24	U	lcd_r4_o(o)	cim0_mclk_o(o)	isp_clk_o(o)	epd_pwr2_o(o)	
25	25	U	lcd_r5_o(o)	cim0_vsyn(io)	slcd_cs_o(o)	epd_pwr3_o(o)	
26	26	U	lcd_r6_o(o)	cim0_hsyn(io)	slcd_rs_o(o)	epd_bd0_o(o)	
27	27	U	lcd_r7_o(o)	cim0_pclk(io)	slcd_te_i(i)	epd_bd1_o(o)	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.1.4 GPIO Port D Summary

Table 28-4 GPIO Port D summary

Bit N	PDN	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	U	-	cim0_d0(io)	isp_rst_o(o)	-	*13*
1	01	U	-	cim0_d1(io)	isp_fsino_o(o)	-	
2	02	U	-	cim0_d2(io)	isp_pwm_o(o)	-	
d3	03	U	-	cim0_d3(io)	isp_pwd_o(o)	-	
4	04	U	-	cim0_d4(io)	isp_shutter_o(o)	-	
5	05	U	-	cim0_d5(io)	isp_frex_o(o)	-	
6	06	U	-	cim0_d6(io)	isp_fsini_i(i)	-	
7	07	U	-	cim0_d7(io)	isp_strobe_i(i)	-	
8	08	U	-	cim0_mclk_o(o)	isp_clk_o(o)	-	
9	09	U	-	cim0_vsyn(io)	isp_gc(io)	-	
10	10	U	-	cim0_hsyn(io)	-	-	
11	11	U	-	cim0_pclk(io)	-	-	
12	12	U	-	cim0_d8(io)	-	-	
13	13	U	-	cim0_d9(io)	-	-	
14	14	D rst-pe	-	-	-	clk32k_o(o)	10
15	15	D rst-pe	exclko_o(o)	-	-	-	
17	17	U	-	-	-	-	2,5
18	18	U	-	-	-	-	3,5
19	19	U	-	-	-	-	4,5
20	20	U	ssi0_dr_i(io)	-	msc1_d0(io)	-	
21	21	U	ssi0_dt_o(io)	-	msc1_d1(io)	-	
22	22	U	ssi0_gpc_o(io)	-	msc1_d2(io)	-	
23	23	U rst-pe	ssi0_ce1_o(io)	-	msc1_d3(io)	-	
24	24	U	ssi0_clk_o(io)	-	msc1_clk_o(o)	-	
25	25	U rst-pe	ssi0_ce0_o(io)	-	msc1_cmd(io)	-	
26	26	U	-	-	uart1_rxd_i(i)	-	
27	27	U	-	-	uart1_cts_i(i)	-	
28	28	U rst-pe	-	-	uart1_rts_o(o)	-	
29	29	U rst-pe	-	-	uart1_txd_o(o)	-	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30	30	U	-	smb0_sda(io)	-	-	
31	31	U	-	smb0_sck(io)	-	-	

28.1.5 GPIO Port E Summary

Table 28-5 GPIO Port E summary

Bit N	PE N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	U	pwm0(io)	smb2_sda(io)	-	-	
1	01	D rst_pe	pwm1_o(o)	-	-	clk32k_o(o)	
2	02	D rst_pe	pwm2_o(o)	-	-	exclko_o(o)	
3	03	U	pwm3(io)	smb2_sck (io)	-	sysclk_o(o)	
10	10	D rst-pe	drv_vbus_o(o)	dmic1_in_i(i)	-	-	*13*
20	20	U	ssi1_dr_i(i)	msc0_d0(io)	msc1_d0(io)	msc2_d0(io)	
21	21	U	ssi1_dt_o(o)	msc0_d1(io)	msc1_d1(io)	msc2_d1(io)	
22	22	U	ssi1_gpc_o(o)	msc0_d2(io)	msc1_d2(io)	msc2_d2(io)	
23	23	U	ssi1_ce1_o(o)	msc0_d3(io)	msc1_d3(io)	msc2_d3(io)	
28	28	U	ssi1_clk_o(o)	msc0_clk_o(o)	msc1_clk_o(o)	msc2_clk_o(o)	
29	29	U	ssi1_ce0_o(o)	msc0_cmd_o(o)	msc1_cmd_o(o)	msc2_cmd_o(o)	
30	30	U	-	smb1_sda(io)	-	-	
31	31	U	-	smb1_sck(io)	-	-	

28.1.6 GPIO Port F Summary

Table 28-6 GPIO Port F summary

Bit N	PF N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	U	-	-	uart0_rxd_i(i)	-	
1	01	U	-	uart4_rxd_i(i)	uart0_cts_i(i)	-	
2	02	U	-	uart4_txd_o(o)	uart0_rts_o(o)	-	
3	03	U	-	-	uart0_txd_o(o)	-	
6	06	D	-	sdato2_o(o)	uart2_txd_o(o)	dmic0_clk_o(o)	
7	07	D	-	sdato3_o(o)	uart2_rxd_i(i)	dmic0_in_i(i)	
10	10	D	-	dmic1_clk_o(o)	-	–	
11	11	D	-	dmic1_in_i(i)	-	–	
12	12	U	pcm0_do_o(o)	-	-	-	
13	13	U	pcm0_clk_o(o)	-	-	-	
14	14	U	pcm0_syn(io)	-	-	-	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

15	15	U	pcm0_di_i(i)	-	-	-	

28.1.7 GPIO Port Z - Shadow Group

The group is a dummy group which is not mapping to any on-chip ports. It is used to avoid 3rd (intermittent) state error during configuration. Please refer 1.2.19 and 1.3.2 for detailed information.

NOTES:

1 If NAND flash is used, this pin must be used as NAND FRB.

2 PD17: GPIO group D bit 17 is used as BOOT_SEL0 during boot and input during GPIO

3 PD18: GPIO group D bit 18 is used as BOOT_SEL1 during boot and input during GPIO.

4 PD19: GPIO group D bit 19 is used as BOOT_SEL2 during boot and input during GPIO.

5 BOOT_SEL2, BOOT_SEL1, BOOT_SEL0 are used to select boot source and function during the processor boot.

6 PA30: GPIO group A bit 30 can only be used as input and interrupt, no pull-up and pull-down. It is also used to select the function between PS2 function and JTAG function of JTAG/UART3/PS2 Pins(TCK_UART3_RTS_PS2_MCLK, TMS_UART3_CTS_PS2_MDATA, TDI_UART3_RxD_PS2_KCLK and TDO_UART3_TxD_PS2_KDATA), which share the same set of pins.
When PA30.function1 is false, select JTAG function.
When PA30.function1 is true, select PS2 function.

7 PA31: GPIO group A bit 31. No corresponding pin exists for this GPIO.
It is only used to select the function between UART and JTAG of JTAG/UART3/PS2 Pins (TCK_UART3_RTS_PS2_MCLK,TMS_UART3_CTS_PS2_MDATA,TDI_UART3_RxD_PS2_KCLK and TDO_UART3_TxD_PS2_KDATA), which share the same set of pins, by using register PASEL [31].
When PA31.function1 is false, select JTAG function.
When PA31.function1 is true, select UART function.

8 If NAND flash is used, this pin must be used as NAND CLE.

9 If NAND flash is used, this pin must be used as NAND ALE.

10 PD14 : this pin is generally used as RTCLK output. If this pin is intended to be used as GPIO or interrupt, CK32CTL(CKPCR[2:1] in RTC) should be configured to 2’b10.Please refer to RTC spec.

11 The pull enable of PA31 is used to control UART and JTAG of JTAG/UART3/ Pins.

13 With Glitch Filter implemented.

28.2 Registers Description

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

RCW - read and write, but clear to 0 by read
RSW - read and write, but set to 1 by read
RWC - read and write, clear to 0 by write 1, write 0 has no effect
RWS - read and write, set to 1 by write 1, write 0 has no effect
RC - read only, and clear to 0 by read
RS - read only, and set to 1 by read
SPEC - special access method, relate to its description

3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

Table 28-8 summarized all memory-mapped registers, which can be programmed to operate GPIO port and alternate function port sharing configuration.

All registers are in 32-bits width. Usually, 1 bit in the register affects a corresponding GPIO port and every GPIO port can be operated independently.

Table 28-7 Registers Memory Map-Address Base

Name	Base	Description
PA	0x10010000	Address base of GPIO Port A
PB	0x10010100	Address base of GPIO Port B
PC	0x10010200	Address base of GPIO Port C
PD	0x10010300	Address base of GPIO Port D
PE	0x10010400	Address base of GPIO Port E
PF	0x10010500	Address base of GPIO Port F
PZ	0x10010F00	Address base of GPIO Group Z (Dummy group)

Table 28-8 GPIO Registers

Name	Description	Reset Value	Address offset	Size
P*x*PIN	PORT A PIN Level Register	0x????????	0x00	32
P*x*INT	PORT A Interrupt Register	0x00000000	0x10	32
P*x*INTS	PORT A Interrupt Set Register	0x00000000	0x14	32
P*x*INTC	PORT A Interrupt Clear Register	0x00000000	0x18	32
P*x*MSK	PORT A Interrupt Mask Register	0x????????	0x20	32
P*x*MSKS	PORT A Interrupt Mask Set Register	0x00000000	0x24	32
P*x*MSKC	PORT A Interrupt Mask Clear Register	0x00000000	0x28	32
P*x*PAT1	PORT A Pattern 1 Register	0x????????	0x30	32
P*x*PAT1S	PORT A Pattern 1 Set Register	0x00000000	0x34	32
P*x*PAT1C	PORT A Pattern 1 Clear Register	0x00000000	0x38	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

P*x*PAT0	PORT A Pattern 0 Register	0x00000000	0x40	32
P*x*PAT0S	PORT A Pattern 0 Set Register	0x00000000	0x44	32
P*x*PAT0C	PORT A Pattern 0 Clear Register	0x00000000	0x48	32
P*x*FLG	PORT A FLAG Register	0x00000000	0x50	32
P*x*FLGC	PORT A FLAG Clear Register	0x00000000	0x58	32
P*x*PEN	PORT A PULL Disable Register	0x00000000	0x70	32
P*x*PENS	PORT A PULL Disable Set Register	0x00000000	0x74	32
P*x*PENC	PORT A PULL Disable Clear Register	0x00000000	0x78	32
PZGIDLD	Grout ID to be load	0x00000000	0xB0	32
PZ_A_CFG0 PZ_B_CFG0 PZ_C_CFG0 PZ_D_CFG0 PZ_E_CFG0 PZ_F_CFG0	Glictch Filter Configure Register 0 for each Port	0x00000000	0xC0 0xD0 0xE0 0xF0 0x0C 0x4C	32
PZ_A_CFG1 PZ_B_CFG1 PZ_C_CFG1 PZ_D_CFG1 PZ_E_CFG1 PZ_F_CFG1	Glictch Filter Configure Register 1 for each Port	0x00000000	0xC4 0xD4 0xE4 0xF4 0x1C 0x5C	32
PZ_A_CFG2 PZ_B_CFG2 PZ_C_CFG2 PZ_D_CFG2 PZ_E_CFG2 PZ_F_CFG2	Glictch Filter Configure Register 2 for each Port	0x00000000	0xC8 0xD8 0xE8 0xF8 0x2C 0x6C	32
PZ_A_CFG3 PZ_B_CFG3 PZ_C_CFG3 PZ_D_CFG3 PZ_E_CFG3 PZ_F_CFG3	Glictch Filter Configure Register 3 for each Port	0x00000000	0xCC 0xDC 0xEC 0xFC 0x3C 0x7C	32

NOTE: P*x***** in the description of register as follows means PA****, PB****, PC****, PD****, PE**** and PF****.

28.2.1 PORT PIN Level Registers (PxPIN)

PAPIN, PBPIN, PCPIN, PDPIN, PEPIN and PFPIN are six 32-bit PORT PIN level registers. They are read-only registers.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PxPIN																																PxBase + 0x00
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PINL31	PINL30	PINL29	PINL28	PINL27	PINL26	PINL25	PINL24	PINL23	PINL22	PINL21	PINL20	PINL19	PINL18	PINL17	PINL16	PINL15	PINL14	PINL13	PINL12	PINL11	PINL10	PINL09	PINL08	PINL07	PINL06	PINL05	PINL04	PINL03	PINL02	PINL01	PINL00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
N	PINL n	Where n = 0 ~ 31 and PINL n = PINL0 ~ PINL31. The PORT PIN level can be read by reading PINL n bit in register PXPIN.	R

28.2.2 PORT Interrupt Registers (PxINT)

PAINT, PBINT, PCINT, PDINT, PEINT and PFINT are six 32-bit interrupt registers.

	PxINT																																PxBase + 0x10
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INT31	INT30	INT29	INT28	INT27	INT26	INT25	INT24	INT23	INT22	INT21	INT20	INT19	INT18	INT17	INT16	INT15	INT14	INT13	INT12	INT11	INT10	INT09	INT08	INT07	INT06	INT05	INT04	INT03	INT02	INT01	INT00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
N	INT n	Where n = 0 ~ 31 and INT n = INT31 ~ INT00. Interrupt enable. 0: Corresponding pin is used as device functions or normal gpio 1: Corresponding pin is used as interrupt	RW

28.2.3 PORT Interrupt Set Registers (PxINTS)

PAINTS, PBINTS, PCINTS, PDINTS, PEINTS and PFINTS are six 32-bit interrupt set registers.

	PxINTS																																PxBase + 0x14
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTS31	INTS30	INTS29	INTS28	INTS27	INTS26	INTS25	INTS24	INTS23	INTS22	INTS21	INTS20	INTS19	INTS18	INTS17	INTS16	INTS15	INTS14	INTS13	INTS12	INTS11	INTS10	INTS09	INTS08	INTS07	INTS06	INTS05	INTS04	INTS03	INTS02	INTS01	INTS00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
N	INTS n	Writing 1 to INTS n will set INT n to 1 in register PXINT. Writing 0 to INTS n will no use.	WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.2.4 PORT Interrupt Clear Registers (PxINTC)

PAINTC, PBINTC, PCINTC, PDINTC, PEINTC and PFINTC are six 32-bit interrupt clear registers.

	PxINTC																															PxBase + 0x18
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTC31	INTC30	INTC29	INTC28	INTC27	INTC26	INTC25	INTC24	INTC23	INTC22	INTC21	INTC20	INTC19	INTC18	INTC17	INTC16	INTC15	INTC14	INTC13	INTC12	INTC11	INTC10	INTC09	INTC08	INTC07	INTC06	INTC05	INTC04	INTC03	INTC02	INTC01	INTC00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	INTC n	Writing 1 to INTC n will set INT n to 0 in register PXINT. Writing 0 to INTC n will no use.	WC

28.2.5 PORT Mask Registers (PxMSK)

PAMSK, PBMSK, PCMSK, PDMSK, PEMSK and PFMSK are six 32-bit PORT MASK registers.

	PxMSK																															PxBase + 0x20
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSK31	MSK30	MSK29	MSK28	MSK27	MSK26	MSK25	MSK24	MSK23	MSK22	MSK21	MSK20	MSK19	MSK18	MSK17	MSK16	MSK15	MSK14	MSK13	MSK12	MSK11	MSK10	MSK09	MSK08	MSK07	MSK06	MSK05	MSK04	MSK03	MSK02	MSK01	MSK00
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	R/W
n	MSK n	Where n = 0 ~ 31 and MSK n = MSK31 ~ MSK0. When PXINT n = 1: 0: Enable the pin as an interrupt source 1: Disable the pin as an interrupt source When PXINT n = 0: 0: Corresponding pin will be used as device function 1: Corresponding pin will be used as gpio	RW

28.2.6 PORT Mask Set Registers (PxMSKS)

PAMSKS, PBMSKS, PCMSKS, PDMSKS, PEMSKS and PFMSKS are six 32-bit PORT MASK set registers.

	PxMSKS																															PxBase + 0x24
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSKS31	MSKS30	MSKS29	MSKS28	MSKS27	MSKS26	MSKS25	MSKS24	MSKS23	MSKS22	MSKS21	MSKS20	MSKS19	MSKS18	MSKS17	MSKS16	MSKS15	MSKS14	MSKS13	MSKS12	MSKS11	MSKS10	MSKS09	MSKS08	MSKS07	MSKS06	MSKS05	MSKS04	MSKS03	MSKS02	MSKS01	MSKS00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
n	MSKS n	Writing 1 to MSKS n will set MSK n to 1 in register PXMSK. Writing 0 to MSKS n will no use.	WC

28.2.7 PORT Mask Clear Registers (PxMSKC)

PAMSKC, PBMSKC, PCMSKC, PDMSKC, PEMSKC and PFMSKC are six 32-bit PORT MASK clear registers.

	PxMSKC																															PxBase + 0x28
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSKC31	MSKC30	MSKC29	MSKC28	MSKC27	MSKC26	MSKC25	MSKC24	MSKC23	MSKC22	MSKC21	MSKC20	MSKC19	MSKC18	MSKC17	MSKC16	MSKC15	MSKC14	MSKC13	MSKC12	MSKC11	MSKC10	MSKC09	MSKC08	MSKC07	MSKC06	MSKC05	MSKC04	MSKC03	MSKC02	MSKC01	MSKC00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	MSKC n	Writing 1 to MSKC n will set MSK n to 0 in register PXMSK. Writing 0 to MSKC n will no use.	WC

28.2.8 PORT PAT1/Direction Registers (PxPAT1)

PAPAT1, PBPAT1, PCPAT1, PDPAT1, PEPAT1 and PFPAT1 are six 32-bit PORT pattern1/direction registers.

	PxPAT1																															PxBase + 0x30
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT131	PAT130	PAT129	PAT128	PAT127	PAT126	PAT125	PAT124	PAT123	PAT122	PAT121	PAT120	PAT119	PAT118	PAT117	PAT116	PAT115	PAT114	PAT113	PAT112	PAT111	PAT110	PAT109	PAT108	PAT107	PAT106	PAT105	PAT104	PAT103	PAT102	PAT101	PAT100
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	R/W
n	PAT1 n	Where n = 0 ~ 31 and PAT1 n = PAT131 ~ PAT10. When PINT n = 1 (Interrupt function): 0: Level trigger interrupt 1: Edge trigger interrupt When PINT n = 0 and PMSK = 0 (Device function): 0: Corresponding pin is used as device 0 or device 1 function 1: Corresponding pin is used as device 2 or device 3 function When PINT n = 0 and PMSK = 0 (GPIO function): 0: Corresponding pin is used as gpio output 1: Corresponding pin is used as gpio input	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.2.9 PORT PAT1/Direction Set Registers (PxPAT1S)

PAPAT1S, PBPAT1S, PCPAT1S, PDPAT1S, PEPAT1S and PFPAT1S are six 32-bit PORT pattern1/direction set registers.

	PxPAT1S																															PxBase + 0x34
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT1S31	PAT1S30	PAT1S29	PAT1S28	PAT1S27	PAT1S26	PAT1S25	PAT1S24	PAT1S23	PAT1S22	PAT1S21	PAT1S20	PAT1S19	PAT1S18	PAT1S17	PAT1S16	PAT1S15	PAT1S14	PAT1S13	PAT1S12	PAT1S11	PAT1S10	PAT1S09	PAT1S08	PAT1S07	PAT1S06	PAT1S05	PAT1S04	PAT1S03	PAT1S02	PAT1S01	PAT1S00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	PAT1S n	Writing 1 to PAT1S n will set PAT1 n to 1 in register PXPAT1. Writing 0 to PAT1S n will no use.	WC

28.2.10 PORT PAT1/Direction Clear Registers (PxPAT1C)

PAPAT1C, PBPAT1C, PCPAT1C, PPAT1C, PEPAT1C and PFPAT1C are six 32-bit PORT pattern1 or direction clear registers.

	PxPAT1S,																															PxBase + 0x38
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT1C31	PAT1C30	PAT1C29	PAT1C28	PAT1C27	PAT1C26	PAT1C25	PAT1C24	PAT1C23	PAT1C22	PAT1C21	PAT1C20	PAT1C19	PAT1C18	PAT1C17	PAT1C16	PAT1C15	PAT1C14	PAT1C13	PAT1C12	PAT1C11	PAT1C10	PAT1C09	PAT1C08	PAT1C07	PAT1C06	PAT1C05	PAT1C04	PAT1C03	PAT1C02	PAT1C01	PAT1C00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	PAT1C n	Writing 1 to PAT1C n will set PAT1 n to 0 in register PXPAT1. Writing 0 to PAT1C n will no use.	WC

28.2.11 PORT PAT0/Data Registers (PxPAT0)

PAPAT0, PBPAT0, PCPAT0, PDPAT0, PEPAT0 and PFPAT0 are six 32-bit PORT Pattern 0 or DATA registers.

	PxPAT0,																															PxBase + 0x40
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT031	PAT030	PAT029	PAT028	PAT027	PAT026	PAT025	PAT024	PAT023	PAT022	PAT021	PAT020	PAT019	PAT018	PAT017	PAT016	PAT015	PAT014	PAT013	PAT012	PAT011	PAT010	PAT009	PAT008	PAT007	PAT006	PAT005	PAT004	PAT003	PAT002	PAT001	PAT000
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
N	PAT0 n	Where n = 0 ~ 31 and PAT0 n = PAT00 ~ PAT031.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		When PINTn = 1 and PPAT1 = 0: 0: Port is low level triggered interrupt input 1: Port is low high triggered interrupt input When PINTn = 1 and PPAT1 = 1: 0: Port is falling edge triggered interrupt input 1: Port is rising edge triggered interrupt input When PINTn = 0 and PMSK = 0 and PPAT1 = 0: 0: Port is pin of device 0 1: Port is pin of device 1 When PINTn = 0 and PMSK = 0 and PPAT1 = 1: 0: Port is pin of device 2 1: Port is pin of device 3 When PINTn = 0 and PMSK = 1 and PPAT1 = 0: 0: Port is GPIO output 0 1: Port is GPIO output 1	

28.2.12 PORT PAT0/Data Set Registers (PxPAT0S)

PAPAT0S, PBPAT0S, PCPAT0S, PDPAT0S, PEPAT0S and PFPAT0S are six 32-bit PORT Pattern0 or DATA set registers.

	PxPAT0S																															PxBase + 0x44
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT0S31	PAT0S30	PAT0S29	PAT0S28	PAT0S27	PAT0S26	PAT0S25	PAT0S24	PAT0S23	PAT0S22	PAT0S21	PAT0S20	PAT0S19	PAT0S18	PAT0S17	PAT0S16	PAT0S15	PAT0S14	PAT0S13	PAT0S12	PAT0S11	PAT0S10	PAT0S09	PAT0S08	PAT0S07	PAT0S06	PAT0S05	PAT0S04	PAT0S03	PAT0S02	PAT0S01	PAT0S00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	PAT0S n	Writing 1 to PAT0S n will set PAT0 n to 1 in register PXPAT0. Writing 0 to PAT0S n will no use.	WC

28.2.13 PORT PAT0/Data Clear Registers (PxPAT0C)

PAPAT0C, PBPAT0C, PCPAT0C, PDPAT0C, PEPAT0C and PFPAT0C are six 32-bit PORT Pattern 0 or DATA clear registers.

	PxPAT0C																															PxBase + 0x48
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT0C31	PAT0C30	PAT0C29	PAT0C28	PAT0C27	PAT0C26	PAT0C25	PAT0C24	PAT0C23	PAT0C22	PAT0C21	PAT0C20	PAT0C19	PAT0C18	PAT0C17	PAT0C16	PAT0C15	PAT0C14	PAT0C13	PAT0C12	PAT0C11	PAT0C10	PAT0C09	PAT0C08	PAT0C07	PAT0C06	PAT0C05	PAT0C04	PAT0C03	PAT0C02	PAT0C01	PAT0C00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
n	PAT0C n	Writing 1 to PAT0C n will set PAT0 n to 0 in register PXPAT0. Writing 0 to PAT0C n will no use.	WC

28.2.14 PORT FLAG Registers (PxFLG)

PAFLG, PBFLG, PCFLG, PDFLG, PEFLG and PFFLG are six 32-bit GPIO FLAG registers. They are read-only registers.

	PxFLG																															PxBase + 0x50
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLAG31	FLAG30	FLAG29	FLAG28	FLAG27	FLAG26	FLAG25	FLAG24	FLAG23	FLAG22	FLAG21	FLAG20	FLAG19	FLAG18	FLAG17	FLAG16	FLAG15	FLAG14	FLAG13	FLAG12	FLAG11	FLAG10	FLAG09	FLAG08	FLAG07	FLAG06	FLAG05	FLAG04	FLAG03	FLAG02	FLAG01	FLAG00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	FLAG n	Where n = 0 ~ 31 and FLAG n = FLAG00 ~ FLAG31. FLAG n is interrupt flag bit for checking the interrupt whether to happen. When GPIO is used as interrupt function and the interrupt happened, the FLAG n in PXFLG will be set to 1.	R

28.2.15 PORT FLAG Clear Registers (PxFLGC)

PAFLGC, PBFLGC, PCFLGC, PDFLGC, PEFLGC and PFFLGC are six 32-bit GPIO FLAG Clear registers. They are read-only registers.

	PxFLGC																															PxBase + 0x58
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLAGC31	FLAGC30	FLAGC29	FLAGC28	FLAGC27	FLAGC26	FLAGC25	FLAGC24	FLAGC23	FLAGC22	FLAGC21	FLAGC20	FLAGC19	FLAGC18	FLAGC17	FLAGC16	FLAGC15	FLAGC14	FLAGC13	FLAGC12	FLAGC11	FLAGC10	FLAGC09	FLAGC08	FLAGC07	FLAGC06	FLAGC05	FLAGC04	FLAGC03	FLAGC02	FLAGC01	FLAGC00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	FLAGC n	When GPIO is used as interrupt function and when write 1 to the bit, the bit FLAG n in PXFLG will be cleared.	RC

28.2.16 PORT PULL Disable Registers (PxPE)

PAPE, PBPE, PCPE, PDPE, PEPE and PFPE are six 32-bit PORT PULL disable registers.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PxPE																															PxBase + 0x70
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PULL31	PULL30	PULL29	PULL28	PULL27	PULL26	PULL25	PULL24	PULL23	PULL22	PULL21	PULL20	PULL19	PULL18	PULL17	PULL16	PULL15	PULL14	PULL13	PULL12	PULL11	PULL10	PULL09	PULL08	PULL07	PULL06	PULL05	PULL04	PULL03	PULL02	PULL01	PULL00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
N	PULL n	Where n = 0 ~ 31 and PULL n = PULL0 ~ PULL31. PULL n is used for setting the port to be PULL UP or PULL DOWN enable. 0: An internal pull up or pull down resistor connects to the port. Up or down is pin dependence 1: No pull up or pull down resistor connects to the port	RW

28.2.17 PORT PULL Set Registers (PxPES)

PAPES, PBPES, PCPES, PDPES, PEPES and PFPES are six 32-bit PORT PULL set registers. They are write-only registers.

	PxPES																															PxBase + 0x74
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PULLS31	PULLS30	PULLS29	PULLS28	PULLS27	PULLS26	PULLS25	PULLS24	PULLS23	PULLS22	PULLS21	PULLS20	PULLS19	PULLS18	PULLS17	PULLS16	PULLS15	PULLS14	PULLS13	PULLS12	PULLS11	PULLS10	PULLS09	PULLS08	PULLS07	PULLS06	PULLS05	PULLS04	PULLS03	PULLS02	PULLS01	PULLS00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
n	PULLS n	Writing 1 to PULLS n will set PULL n to 1 in register PXPE. Writing 0 to PULLS n will no use.	WC

28.2.18 PORT PULL Clear Registers (PxPEC)

PAPEC, PBPEC, PCPEC, PDPEC, PEPEC and PFPEC are six 32-bit PORT PULL clear registers.

	PxPES																															PxBase + 0x78
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PULLC31	PULLC30	PULLC29	PULLC28	PULLC27	PULLC26	PULLC25	PULLC24	PULLC23	PULLC22	PULLC21	PULLC20	PULLC19	PULLC18	PULLC17	PULLC16	PULLC15	PULLC14	PULLC13	PULLC12	PULLC11	PULLC10	PULLC09	PULLC08	PULLC07	PULLC06	PULLC05	PULLC04	PULLC03	PULLC02	PULLC01	PULLC00
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
n	PULLC n	Writing 1 to PULLC n will set PULL n to 0 in register PXPE. Writing 0 to PULLC n will no use.	WC

28.2.19 GPIOZ Group ID to Load (PzGID2LD)

This register controls which Port Group to be load from GPIOZ.

	PzGID2LD																															0xB0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																														GPID		
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	R/W
3	GPID	0: GPIO Port A 1: GPIO Port B 2: GPIO Port C 3: GPIO Port D 4: GPIO Port E 5: GPIO Port F 6: GPIO Port G	W

28.2.20 PORT GLITCH FILTER Configure (PxCFG0~3)

These four registers used for suppress the glitch on port signal. The range of glitch width is:

1 x pclk ~ 15 x pclk

	PACFG0, PBCFG0, PCCFG0, PDCFG0, PECFG0, PFCFG0																PzBase + 0xC0, 0xD0,0xE0, 0xF0, 0x0C, 0x4C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PxCFG0_31	PxCFG0_30	PxCFG0_29	PxCFG0_28	PxCFG0_27	PxCFG0_26	PxCFG0_25	PxCFG0_24	PxCFG0_23	PxCFG0_22	PxCFG0_21	PxCFG0_20	PxCFG0_19	PxCFG0_18	PxCFG0_17	PxCFG0_16	PxCFG0_15	PxCFG0_14	PxCFG0_13	PxCFG0_12	PxCFG0_11	PxCFG0_10	PxCFG0_9	PxCFG0_8	PxCFG0_7	PxCFG0_6	PxCFG0_5	PxCFG0_4	PxCFG0_3	PxCFG0_2	PxCFG0_1	PxCFG0_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	PACFG1, PBCFG1, PCCFG1, PDCFG1, PECFG1, PFCFG1																PzBase + 0xC4, 0xD4,0xE4, 0xF4, 0x1C, 0x5C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PxCFG1_31	PxCFG1_30	PxCFG1_29	PxCFG1_28	PxCFG1_27	PxCFG1_26	PxCFG1_25	PxCFG1_24	PxCFG1_23	PxCFG1_22	PxCFG1_21	PxCFG1_20	PxCFG1_19	PxCFG1_18	PxCFG1_17	PxCFG1_16	PxCFG1_15	PxCFG1_14	PxCFG1_13	PxCFG1_12	PxCFG1_11	PxCFG1_10	PxCFG1_9	PxCFG1_8	PxCFG1_7	PxCFG1_6	PxCFG1_5	PxCFG1_4	PxCFG1_3	PxCFG1_2	PxCFG1_1	PxCFG1_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PACFG2, PBCFG2, PCCFG2, PDCFG2, PECFG2, PFCFG2																PzBase + 0xFC8, 0xFD8,0xFE8, 0xFF8, 0xF2C, 0xF6C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PxCFG2_31	PxCFG2_30	PxCFG2_29	PxCFG2_28	PxCFG2_27	PxCFG2_26	PxCFG2_25	PxCFG2_24	PxCFG2_23	PxCFG2_22	PxCFG2_21	PxCFG2_20	PxCFG2_19	PxCFG2_18	PxCFG2_17	PxCFG2_16	PxCFG2_15	PxCFG2_14	PxCFG2_13	PxCFG2_12	PxCFG2_11	PxCFG2_10	PxCFG2_9	PxCFG2_8	PxCFG2_7	PxCFG2_6	PxCFG2_5	PxCFG2_4	PxCFG2_3	PxCFG2_2	PxCFG2_1	PxCFG2_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	PACFG3, PBCFG3, PCCFG3, PDCFG3, PECFG3, PFCFG3																PzBase + 0xFCC, 0xFDC,0xFEC, 0xFFC, 0xF3C, 0xF7C															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PxCFG3_31	PxCFG3_30	PxCFG3_29	PxCFG3_28	PxCFG3_27	PxCFG3_26	PxCFG3_25	PxCFG3_24	PxCFG3_23	PxCFG3_22	PxCFG3_21	PxCFG3_20	PxCFG3_19	PxCFG3_18	PxCFG3_17	PxCFG3_16	PxCFG3_15	PxCFG3_14	PxCFG3_13	PxCFG3_12	PxCFG3_11	PxCFG3_10	PxCFG3_9	PxCFG3_8	PxCFG3_7	PxCFG3_6	PxCFG3_5	PxCFG3_4	PxCFG3_3	PxCFG3_2	PxCFG3_1	PxCFG3_0
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
4	{PxCFG3, PxCFG2, PxCFG1, PxCFG0}	0000: GPIO Port x, disable glitch filter function 0001: GPIO Port x, maximum glitch width to be filtered is 1 pclk 0010: GPIO Port x, maximum glitch width to be filtered is 2 pclk 0011: GPIO Port x, maximum glitch width to be filtered is 3 pclk ..... : ..... 1110: GPIO Port x, maximum glitch width to be filtered is 14 pclk 1111: GPIO Port x, maximum glitch width to be filtered is 15 pclk	WR

Note:

1. pclk is the clock period configure to APB bus.
2. This function only used in interrupt input mode.
3. Not every bit for each port is implement this function. please refer *1.1*, only bits marked with "*13*" has this function.

28.3 Program Guide

28.3.1 Port Function Guide

INT	MASK	PAT1	PAT0	Port Description
1	0	0	0	Port is low level triggered interrupt input.
1	0	0	1	Port is high level triggered interrupt input.
1	0	1	0	Port is fall edge triggered interrupt input.
1	0	1	1	Port is rise edge triggered interrupt input.
1	1	0	0	Port is low level triggered interrupt input. Interrupt is masked. Flag is recorded.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	1	0	1	Port is high level triggered interrupt input. Interrupt is masked. Flag is recorded.
1	1	1	0	Port is fall edge triggered interrupt input. Interrupt is masked. Flag is recorded.
1	1	1	1	Port is rise edge triggered interrupt input. Interrupt is masked. Flag is recorded.
0	0	0	0	Port is pin of device 0.
0	0	0	1	Port is pin of device 1.
0	0	1	0	Port is pin of device 2.
0	0	1	1	Port is pin of device 3.
0	1	0	0	Port is GPIO output 0.
0	1	0	1	Port is GPIO output 1.
0	1	1	?	Port is GPIO input.

28.3.2 Configure without 3rd-unexpected state

Function switching between devices and gpio is controlled by combination of 4 registers: INT/MASK/PAT1/PAT0. As the four register is accessed by soft ware in serial sequence, which posing a risk that third states occur during configuration.

For example:

current state is:

INT = 0x0, MASK = 0x0, PAT0 = 0x0, PAT1 = 0x0;

target state is:

INT = 0x1, MASK = 0x1, PAT0 = 0x1, PAT1 = 0x1;

If the first step is to configure INT to 0x1, the 3rd state is:

INT = 0x1, MASK = 0x0, PAT0 = 0x0, PAT1 = 0x0;

This might triggers interrupt if port is low level.

To avoid the unexpected 3-state, the four registers must be updated in the same time.

There is an dummy group GPIOZ, which only used in this purpose. The working mechanism is the information of configure can be temporarily stored in the GPIOZ. When assert GPIOZ.LD, all the four registers will load their content into target group at the same time.

Example:

If the Port B current state is:

INT = 0x0, MASK = 0x0, PAT0 = 0x0, PAT1 = 0x0;

It needs to be configured to state:

INT = 0x1, MASK = 0x1, PAT0 = 0x1, PAT1 = 0x1;

There are two steps to finish:

Step1: put the configuration to temporary buffer: GPIOZ

PzINTS = 0x1;

PzMSKS = 0x1;

PzPAT0S = 0x1;

PzPAT1S = 0x1;

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

alternatively, PzINT, PzMSK, PzPAT0, PzPAT1 are also can be configured straightly.

Step2: configure PzGID2LD to specify which port group to load

PzGID2LD = 0x1; (0x1 stands for GPIO Port B)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29 SMB Controller

29.1 Overview

The SMB bus is a two-wire serial interface, consisting of a serial data line (SDA) and a serial clock (SCL). These wires carry information between the devices connected to the bus. Each device is recognized by a unique address and can operate as either a “transmitter” or “receiver,” depending on the function of the device. Devices can also be considered as masters or slaves when performing data transfers. A master is the device that initializes/terminates a data transfer on the bus and generates clock signals to permit that transfer. During that time, any addressed device is considered as a slave. The SMB controller is software controlled. It behaves as a master or a slave. However, operating as a master and slave simultaneously is not supported.

29.1.1 Features

- Two-wire SMB serial interface – consists of a serial data line (SDA) and a serial clock (SCL)
- Two speeds
  - Standard mode (100 Kb/s)
  - Fast mode (400 Kb/s)
- Device clock is identical with pclk(APB clock)
- Programmable SCL generator
- Master or slave SMB operation
- 7-bit addressing/10-bit addressing whether configure as Master or Slave
- Transmit General Call or START byte
- level transmit and receive FIFOs
- Interrupt operation
- The number of devices that you can connect to the same SMB-bus is limited only by the maximum bus capacitance of 400pF
- APB interface
- 4 independent SMB channels (SMB0, SMB1, SMB2, SMB3)

29.1.2 Pin Description

Table 29-1 SMB Pin Description

Name	Width	IO	Description
SDA	1-bit	IO	SMB serial data
SCL	1-bit	IO	SMB serial clock

Please note that dedicate pull-up resistors must be introduced on board-level. The low-to-high (rise time) transition is highly dependent on RC time constant of the bus.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Totally speaking, for standard-mode SMB-bus system, the pull up resistor depends on following parameters:

- Supply voltage
- Bus capacitance
- Number of connected devices

For fast-mode SMB-bus system, switched pull-up circuit may be essential for strict speed and load requirement. (Please refer NXP Semiconductors, *UM10204: I2C-Bus specification and user manual (Rev.6--4 April 2014)*')

29.2 Registers

29.2.1 Registers Memory Map

A read operation to an address location that contains unused bits results in a 0 value being returned on each of the unused bits.

Registers in SMB controller can be accessed by indicating 24-bit Address Base combined with 8-bit Address Offset.

Table 29-2 Registers Memory Map-Address Base

Name	Addr Base	Description
SMB0	0x10050000	Address base of SMB0
SMB1	0x10051000	Address base of SMB1
SMB2	0x10052000	Address base of SMB2
SMB3	0x10053000	Address base of SMB3

Table 29-3 Registers Memory Map-Address Offset

Name	Addr Offset	Description	Width	RW	Reset
SMB_CON	0x00	SMB control	8bits	RW	0x7D
SMB_TAR	0x04	SMB target address	13bits	RW	0x1055
SMB_SAR	0x08	SMB slave address	10bits	RW	0x055
SMB_DC	0x10	SMB data buffer and command	9bits	RW	0x000
SMB_SHCNT	0x14	Standard speed SMB SCL high count	16bits	RW	0x0190
SMB_SLCNT	0x18	Standard speed SMB SCL low count	16bits	RW	0x01D6
SMB_FHCNT	0x1C	Fast speed SMB SCL high count	16bits	RW	0x003C
SMB_FLCNT	0x20	Fast speed SMB SCL low count	16bits	RW	0x0082
SMB_INTST	0x2C	SMB Interrupt Status	12bits	R	0x000
SMB_INTM	0x30	SMB Interrupt Mask	12bits	R/W	12'h8FF
SMB_RINTST	0x34	SMB Raw Interrupt Status	12bits	R	0x0
SMB_RXTL	0x38	SMB RxFIFO Threshold	6 bits	RW	0x1F
SMB_TXTL	0x3C	SMB TxFIFO Threshold	6 bits	RW	0x20

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SMB_CINT	0x40	Clear Interrupts	1 bit	R	0x0
SMB_CRXUF	0x44	Clear RXUF Interrupt	1 bit	R	0x0
SMB_CRXOF	0x48	Clear RX_OVER Interrupt	1 bit	R	0x0
SMB_CTXOF	0x4C	Clear TX_OVER Interrupt	1 bit	R	0x0
SMB_CRXREQ	0x50	Clear RDREQ Interrupt	1 bit	R	0x0
SMB_CTXABT	0x54	Clear TX_ABRT Interrupt	1 bit	R	0x0
SMB_CRXDN	0x58	Clear RX_DONE Interrupt	1 bit	R	0x0
SMB_CACT	0x5c	Clear ACTIVITY Interrupt	1 bit	R	0x0
SMB_CSTP	0x60	Clear STOP Interrupt	1 bit	R	0x0
SMB_CSTT	0x64	Clear START Interrupt	1 bit	R	0x0
SMB_CGC	0x68	Clear GEN_CALL Interrupt	1 bit	R	0x0
SMB_ENABLE	0x6C	SMB Enable	1 bit	RW	0x0
SMB_ST	0x70	SMB Status register	7 bits	R	0x06
SMB_TXFLR	0x74	TxFIFO Level Register	6 bits	R	0x0
SMB_RXFLR	0X78	RxFIFO Level Register	6 bits	R	0x0
SMB_SDAHD	0x7C	SMB SDA Hold time Register	16 bits	RW	0x55
SMB_ABTSRC	0x80	SMB Transmit Abort Status Register	32 bits	R	0x0000
SMB_DMACR	0x88	DMA Control Register	2 bits	R/W	0x0
SMB_DMATDLR	0x8c	DMA Transmit Data Level	6 bits	R/W	0x0
SMB_DMARDLR	0x90	DMA Receive Data Level	6 bits	R/W	0x0
SMB_SDASU	0x94	SMB SDA Setup Register	8 bits	RW	0x64
SMB_ACKGC	0x98	SMB ACK General Call Register	1 bit	RW	0x1
SMB_ENBST	0x9C	SMB Enable Status Register	3 bits	R	0x0
SMB_FLT	0xA0	SMB Filter Register	8bits	RW	0x1

29.2.2 Registers and Fields Description

29.2.2.1 SMB_CON (SMB Control Register)

This register can be written only when SMB is disabled, which corresponds to SMB_ENABLE[0] being set to 0. Writes at other time have on effect.

	SMB_CON																															BASE + 0x00
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved									SLVDIS	RESTART	MATP	SATP	SPEED		MD
RST																	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1

Bits	Name	Description	RW
15:7	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

6	SLVDIS	This bit controls whether SMB has its slave disabled. 0: slave is enabled 1: slave is disabled Note: Software should ensure that if this bit is written with'0', then bit 0 should also be written with '0'.	RW
5	RESTART	Determines whether RESTART conditions may be sent when acting as a master. Some older slaves do not support handling RESTART conditions. 0: disable 1: enable	RW
4	MATP	This bit is a read-only copy of SMB_TAR.MATP. 0: 7-bit addressing 1: 10-bit addressing	R
3	SATP	When acting as a slave, this bit controls whether the SMB responds to 7-bit or 10-bit addresses. 0: 7-bit addressing 1: 10-bit addressing	RW
2:1	SPEED	These bits control at which speed the SMB operates. 1: standard mode (100 kbps) 2: fast mode (400 kbps) **NOTE:** when these two bits are set to 2'b00 or 2'b11, the speed mode will be automatically set to 2'b10 i.e. fast mode.	RW
0	MD	This bit controls whether the SMB master is enabled. 0: master disabled 1: master enabled Note: Software should ensure that if this bit is written with'1', then bit 6 should also be written with '1'.	RW

29.2.2.2 SMB_TAR (SMB Target Address Register)

	SMB_TAR																BASE + 0x04															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved			MATP	SPECIAL	GC_OR_START	SMBTAR									
RST																	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	1

Bits	Name	Description	R/W
15:13	Reserved	Writing has no effect, read as zero.	R
12	MATP	This bit controls whether the SMB starts its transfers in 7- or 10-bit addressing mode when acting as a master.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: 7-bit addressing 1: 10-bit addressing **NOTE:** this bit is initially set to 0.	
11	SPECIAL	This bit indicates whether software performs a General Call or START BYTE command. 0: ignore the bit of GC_OR_START and use SMBTAR normally 1: perform special SMB command as specified in GC_OR_START bit **NOTE:** this bit is initially set to 0	RW
10	GC_OR_START	If bit 11 (SPECIAL) is set to 1, then this bit indicates whether a General Call or START byte command is to be performed by the SMB. 0: General Call Address – after issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TX_ABRT) of the SMB_INTST register. The SMB remains in General Call mode until the SPECIAL bit value (bit 11) is cleared 1: START BYTE	RW
9:0	SMBTAR	This is the target address for any master transaction. When transmitting a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. If the SMB_TAR and SMB_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself, but only to a slave.	RW

NOTE:

(1).It is not necessary to perform any write to this register if SMB is enabled as an SMB slave only.

(2).The meaning of "START byte" in bit 10 is different with the SMB "start condition".

"START byte" can only be applied when a device does not have such an interface, it must constantly monitor the bus via software. For more detail information, please refer *UM10204: I2C-Bus specification and user manual (Rev.6--4 April 2014),* '3.1.15 START byte' section.

"start condition" means A HIGH to LOW transition on the SDA line while SCL is HIGH defines a START condition. please refer *UM10204: I2C-Bus specification and user manual (Rev.6--4 April 2014),* '3.1.4 START and STOP' section for more information.

29.2.2.3 SMB_SAR (SMB Slave Address Register)

	SMB_SAR																															BASE + 0x08
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved						SMBSAR									
RST																	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
15:10	Reserved	Writing has no effect, read as zero.	R
9:0	SMBSAR	The SMBSAR holds the slave address when the SMB is operating as a slave. For 7-bit addressing, only SMBSAR[6:0] is used. This register can be written only when the SMB interface is disabled. Writes at other times have no effect.	RW

NOTE: It is not necessary to perform any write to this register if SMB is enabled as an SMB master only.

29.2.2.4 SMB_DC (SMB Rx/Tx Data Buffer and Command Register)

This is the register the CPU writes to when filling the TX FIFO and the CPU reads from when retrieving bytes from RX FIFO.

	SMB_DC																	BASE + 0x10														
Bit																		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																		Reserved					RESTART	STOP	CMD	DAT							
RST																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:11	Reserved	Writing has no effect, read as zero.	R
10	RESTART	This bit controls whether a RESTART is issued before the byte is sent or received. 1: If SMB_CON.RESTART is 1, a RESTART is issued before the byte is sent or received, regardless of whether or not the transfer direction is changing from the previous command; if SMB_CON.RESTART is 0, a STOP followed by a START is issued instead. 0: If SMB_CON.RESTART is 1, a RESTERT is issued if the transfer direction is changing from the previous command, if SMB_CON.RESTART is 0, a STOP followed by a START is issued instead.	W
9	STOP	This bit controls whether a STOP is issued after the byte is sent or received. 1: STOP is issued after the byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. 0: STOP is not issued after the byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master continues the current transfer by sending/receiving data bytes according to the value of	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		the CMD bit. If the Tx FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the Tx FIFO.	
8	CMD	This bit controls whether a read or a write is performed. This bit does not control the direction when the SMB acts as a slave. It controls only the direction when it acts as a master. 1: Read 0: Write When programming this bit, you should remember the following: attempting to perform a read operation after a General Call command has been sent results in a *TX_ABRT* interrupt (bit 6 of the *SMB_RAW_INTR_STAT* register), unless bit 11 (*SPECIAL*) in the *SMB_TAR* register has been cleared. If a "1" is written to this bit after receiving a *RD_REQ* interrupt, then a *TX_ABRT* interrupt occurs.	W
7:0	DAT	This register contains the data to be transmitted or received on the SMB bus. If you are writing to this register and want to perform a read, bits 7:0 (*DAT*) are ignored by the SMB module. However, when you read this register, these bits return the value of data received on the SMB interface.	RW

NOTE: [1].this command only transfer 8-bit data combined with 1-bit CMD, extra bits on the bus will be eliminated. i.e. only 8-0 bits on the bus are accepted by SMB controller.
[2].For the detail information about what condition could generate START, STOP and RESTART, please refer "*1.3.6 The condition could generate START, STOP and RESTART*".
[3].The differences between *SMB_ENABLE[5]*(RESTART) and *SMB_DC[10]*(RESTART) is that SMB_ENABLE[5] determine whether or not SMB the capability of generating repeat start, but not transmit RESTART when this bit be setted. If this bit set "1", SMB_DC[10] controls whether a RESTART is issued, if set "0", SMB_DC[10] have no meaning.

29.2.2.5 SMB_SHCNT (SMB Standard Speed SCL High Count Register)

	SMB_SHCNT																BASE + 0x14															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SMBSHCNT															
RST																	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0

Bits	Name	Description	R/W
15:0	SMBSHCNT	This register must be set before any SMB bus transaction can take place to ensure proper I/O timing. The register sets the SCL clock high-period count for standard speed. This register can be written only when the SMB interface is disabled which corresponds to SMB_ENABLE[0] being set to 0. Writes at other	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.	

29.2.2.6 SMB_SLCNT (SMB Standard Speed SCL Low Count Register)

	SMB_SLCNT																														BASE + 0x18
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SMBSLCNT															
RST																	0	0	0	0	0	0	0	1	1	1	0	1	0	1	1	0

Bits	Name	Description	R/W
15:0	SMBSLCNT	This register must be set before any SMB bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for standard speed. This register can be written only when the SMB interface is disabled which corresponds to SMB_ENABLE[0] being set to 0. Writes at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted, results in 8 being set.	RW

29.2.2.7 SMB_FHCNT (SMB Fast Speed SCL High Count Register)

	SMB_FHCNT																														BASE + 0x1C
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SMBFHCNT															
RST																	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0

Bits	Name	Description	R/W
15:0	SMBFHCNT	This register must be set before any SMB bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for fast speed. This register can be written only when the SMB interface is disabled which corresponds to SMB_ENABLE[0] being set to 0. Writes at other times have no effect. Minimum value allowed for the SMBSHCNT registers is 6. If the set value was less than 6, it will be automatically set to 6.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.8 SMB_FLCNT (SMB Fast Speed SCL Low Count Register)

	SMB_FLCNT																BASE + 0x20															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SMBFLCNT															
RST																	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0

Bits	Name	Description	R/W
15:0	SMBFLCNT	This register must be set before any SMB bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for fast speed. This register can be written only when the SMB interface is disabled which corresponds to SMB_ENABLE[0] being set to 0. Writes at other times have no effect. Minimum value that can be programmed in the SMBSLCNT registers is 8. If the set value was less than 8, it will be automatically set to 8	RW

29.2.2.9 SMB_INTST (SMB Interrupt Status Register)

Each bit in this register has a corresponding mask bit in the SMBINTM register. These bits are cleared by reading the matching interrupt clear register. The unmasked raw versions of these bits are available in the *SMB_RAW_INTR_STAT* register

	SMB_INTST																BASE + 0x2C															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved				IGC	ISTT	ISTP	IACT	RXDN	TXABT	RDREQ	TXEMP	TXOF	RXFL	RXOF	RXUF
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:12	Reserved	Writing has no effect, read as zero.	R
11	IGC	Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling SMB or when the CPU reads bit 0 of the *SMB_CGC* register. SMB stores the received data in the Rx buffer.	R
10	ISTT	Indicates whether a START or RESTART condition has occurred on the SMB interface regardless of whether SMB is operating in slave or master mode.	R
9	ISTP	Indicates whether a STOP condition has occurred on the SMB interface	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		regardless of whether SMB is operating in slave or master mode.	
8	IACT	This bit captures SMB activity and stays set until it is cleared. There are four ways to clear it: 1 Disabling the SMB 2 Reading the SMB_CACT register 3 Reading the SMB_CINT register 4 System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the SMB module is idle, this bit remains set until cleared, indicating that there was activity on the bus.	R
7	RXDN	When the SMB is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	R
6	TXABT	This bit indicates if SMB, as an SMB transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an SMB master or an SMB slave, and is referred to as a “transmit abort”. When this bit is set to 1, the SMB_ABTSRC register indicates the reason why the transmit abort takes places. **NOTE:** The SMB flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register SMB_CTXABT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	R
5	RDREQ	This bit is set to 1 when SMB is acting as a slave and another SMB master is attempting to read data from SMB. The SMB holds the SMB bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the SMB_DC register. This bit is set to 0 just after the processor reads the SMB_CRREQ register.	R
4	TXEMP	This bit is set to 1 when the transmit buffer is at or below the threshold value set in the SMB_TXTL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the SMB_ENB bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with SMB_ENB[0]=0.	R
3	TXOF	Set during transmit if the transmit buffer is filled to SMB_TX_BUFFER_DEPTH and the processor attempts to issue another SMB command by writing to the SMB_DC register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when SMB_ENB[0]=0, this interrupt is cleared.	R
2	RXFL	Set when the receive buffer reaches or goes above the SMB_RXTL threshold in the SMB_RXTL register. It is automatically cleared by	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		hardware when buffer level goes below the threshold. If the module is disabled (SMB_ENB[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the SMB_ENB bit 0 is set to 0, regardless of the activity that continues.	
1	RXOF	Set if the receive buffer is completely filled to 64 and an additional byte is received from an external SMB device. The SMB acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (SMB_ENB[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when SMB_ENB[0]=0, this interrupt is cleared.	R
0	RXUF	Set if the processor attempts to read the receive buffer when it is empty by reading from the SMB_DC register. If the module is disabled (SMB_ENB[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when SMB_ENB[0]=0, this interrupt is cleared.	R

NOTE: *29.3.5* summary all the condition could flush TX FIFO.

29.2.2.10 SMB_INTM (SMB Interrupt Mask Register)

These bits mask their corresponding interrupt status bits. This register is active low; a value of 0 masks the interrupt, whereas a value of 1 unmasks the interrupt.

	SMB_INTM																BASE + 0x30															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved				MIGC	MISTT	MISTP	MIACT	MRXDN	MTXABT	MRDREQ	MTXEMP	MTXOF	MRXFL	MRXOF	MRXUF
RST																	0	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1

Bits	Name	Description	R/W
15:12	Reserved	Writing has no effect, read as zero.	R
11	MIGC	These bits mask their corresponding interrupt status bits in the SMBINTST register.	RW
10	MISTT		RW
9	MISTP		RW
8	MIACT		RW
7	MRXDN		RW
6	MTXABT		RW
5	MRDREQ		RW
4	MTXEMP		RW
3	MTXOF		RW
2	MRXFL		RW
1	MRXOF		RW
0	MRXUF		RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.11 SMB_RAW_INTR_STAT

Address: **BASE + 0x34**

Unlike the *SMB_INTST* register, these bits are not masked so they always show the true status of the SMB. See *SMB_INTST* for a detailed description of these bits. All bits are only read.

29.2.2.12 SMB_RXTL (SMB Receive FIFO Threshold Register)

	SMB_RXTL																															BASE + 0x38
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved										RXTL					
RST																	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

Bits	Name	Description	R/W
15:6	Reserved	Writing has no effect, read as zero.	R
5:0	RXTL	Receive FIFO Threshold Level. Controls the level of entries that triggers the RxFIFO full interrupt. A value of n sets the threshold for (n+1) entries. For example, a value of 0 sets the threshold for 1 entry, and a value of 255 sets the threshold for 256 entries.	RW

29.2.2.13 SMB_TXTL (SMB Transmit FIFO Threshold Register)

	SMB_TXTL																															BASE + 0x3C
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved										TXTL					
RST																	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bits	Name	Description	R/W
15:6	Reserved	Writing has no effect, read as zero.	R
5:0	TXTL	Transmit FIFO Threshold Level. Controls the level of entries that trigger the TxFIFO empty interrupt. A value of n sets the threshold for n entries. For example, A value of 0 sets the threshold for 0 entries, and a value of 255 sets the threshold for 255 entries.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.14 SMB_CINT (SMB Clear Combined and Individual Interrupt Register)

	SMB_CINT																															BASE + 0x40
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CINT
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CINT	Read this register to clear the combined interrupt, all individual interrupts, and the SMB_ABTSRC register. This bit does not clear hardware clearable interrupts but software clearable interrupts. Refer to Bit 9 of the SMB_ABTSRC register for an exception to clearing SMB_ABTSRC.	R

29.2.2.15 SMB_CRXUF (SMB Clear RXUF Interrupt Register)

	SMB_CRXUF																															BASE + 0x44
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CRXUF
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CRXUF	Read this register to clear the RXUF interrupt (bit 0) of the SMB_INTST register.	R

29.2.2.16 SMB_CRXOF (SMB Clear RXOF Interrupt Register)

	SMB_CRXOF																															BASE + 0x48
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CRXOF
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	CRXOF	Read this register to clear the RXOF interrupt (bit 1) of the SMBINTST register.	R

29.2.2.17 SMB_CTXOF (SMB Clear TX_OVER Interrupt Register)

	SMB_CTXOF																BASE + 0x4C															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CTXOF
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CTXOF	Read this register to clear the TX_OVER interrupt (bit 3) of the SMBINTST register.	R

29.2.2.18 SMB_CRXREQ (SMB Clear RDREQ Interrupt Register)

	SMB_CRXREQ																BASE + 0x50															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CLRDREQ
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CLRDREQ	Read this register to clear the RDREQ interrupt (bit 5) of the SMBINTST register.	R

29.2.2.19 SMB_CTXABT (SMB Clear TX_ABRT Interrupt Register)

	SMB_CTXABT																BASE + 0x54															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CTXABT
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CTXABT	Read this register to clear the TX_ABRT interrupt (bit 6) of the SMB_INTST register, and the SMB_ABTSRC register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. Refer to Bit 9 of the SMB_ABTSRC register for an exception to clearing SMB_ABTSRC.	R

29.2.2.20 SMB_CRXDN (SMB Clear RX_DONE Interrupt Register)

	SMB_CRXDN																BASE + 0x58															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved															CRXDN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CRXDN	Read this register to clear the RX_DONE interrupt (bit 7) of the SMB_INTST register.	R

29.2.2.21 SMB_CACT (SMB Clear ACTIVITY Interrupt Register)

	SMB_CACT																BASE + 0x5C															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved															CACT
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CACT	Reading this register clears the ACTIVITY interrupt if the SMB is not active anymore. If the SMB module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the SMBINTST register.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.22 SMB_CSTP (SMB Clear STOP Interrupt Register)

	SMB_CSTP																BASE + 0x60															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CSTP
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CSTP	Read this register to clear the STOP interrupt (bit 9) of the SMBINTST register.	R

29.2.2.23 SMB_CSTT (SMB Clear START Interrupt Register)

	SMB_CSTT																BASE + 0x64															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CSTT
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CSTT	Read this register to clear the START interrupt (bit 10) of the SMBINTST register.	R

29.2.2.24 SMB_CGC (SMB Clear GEN_CALL Interrupt Register)

	SMB_CGC																BASE + 0x68															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															CGC
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	CGC	Read this register to clear the GEN_CALL interrupt (bit 11) of SMBINTST register.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.25 SMB_ENABLE (SMB Enable Register)

	SMBE_NABLE																BASE + 0x6C															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved														ABORT	SMBENB
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:2	Reserved	Writing has no effect, read as zero.	R
1	ABORT	When set, the controller initiates the transfer abort. 0: ABORT not initiated or ABORT done 1: ABORT operation in progress The software can abort the SMB transfer in master mode by setting this bit. The software can set this bit only when ENABLE is already set; otherwise, the controller ignores any write to ABORT bit. The software cannot clear the ABORT bit once set. In response to an ABORT, the controller issues a STOP and flushes the Tx FIFO after completing the current transfer, then sets the TX_ABORT interrupt after the abort operation. The ABORT bit is cleared automatically after the abort operation.	R/W
0	SMBENB	Controls whether the SMB is enabled. 0: Disables SMB (TX and RX FIFOs are held in an erased state) 1: Enables SMB Software can disable SMB while it is active. However, it is important that care be taken to ensure that SMB is disabled properly. When SMB is disabled, the following occurs: – The TX FIFO and RX FIFO get flushed. – Status bits in the SMB_INTST register are still active until SMB goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the SMB stops the current transfer at the end of the current byte and does not acknowledge the transfer.	R/W

29.2.2.26 SMB_ST(SMB Status Register)

This is a read-only register used to indicate the current transfer status and FIFO status. The status register may be read at any time. None of the bits in this register request an interrupt.

When the SMB is disabled by writing 0 in bit 0 of the SMB_ENABLE register:

– Bits 1 and 2 are set to 1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

– Bits 3 and 4 are set to 0

When the master or slave state machine goes to idle and SMB_ENABLE[0]=0:

– Bits 5 and 6 are set to 0

	SMB_ST																BASE + 0x70															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved									SLVACT	MSTACT	RFF	RFNE	TFE	TFNF	ACT
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0

Bits	Name	Description	R/W
15:7	Reserved	Writing has no effect, read as zero.	R
6	SLVACT	Slave FSM Activity Status. 0: Slave FSM is in IDLE state so the Slave part of SMB is not Active 1: Slave FSM is not in IDLE state so the Slave part of SMB is Active	R
5	MSTACT	Master FSM Activity Status. 0: Master FSM is in IDLE state so the Master part of SMB is not Active 1: Master FSM is not in IDLE state so the Master part of SMB is Active	R
4	RFF	Receive FIFO Completely Full. When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0: Receive FIFO is not full 1: Receive FIFO is full	R
3	RFNE	Receive FIFO Not Empty. This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0: Receive FIFO is empty 1: Receive FIFO is not empty	R
2	TFE	Transmit FIFO Completely Empty. When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty	R
1	TFNF	Transmit FIFO Not Full. Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0: Transmit FIFO is full 1: Transmit FIFO is not full	R
0	ACT	SMB Activity Status. The OR of SLVACT and MSTACT bits.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.2.2.27 SMB_TXFLR(SMB Transmit FIFO Level Register)

This register contains the number of valid data entries in the transmit FIFO buffer. It is cleared whenever:

- The SMB is disabled
- There is a transmit abort—that is, TX_ABRT bit is set in the SMB_RAW_INTR_STAT register
- The slave bulk transmit mode is aborted

The register increments whenever data is placed into the transmit FIFO and decrements when data is taken from the transmit FIFO.

		BASE+0X74					
Bit		5	4	3	2	1	0
	Reserved	TXFLR					
RST		0	0	0	0	0	0

Bits	Name	Description	R/W
31:6	Reserved	Reserved.	N/A
5:0	TXFLR	Contains the number of valid data entries in the transmit FIFO	R

29.2.2.28 SMB_RXFLR(SMB Receive FIFO Level Register)

This register contains the number of valid data entries in the receive FIFO buffer. It is cleared whenever:

- The SMB is disabled
- Whenever there is a transmit abort caused by any of the events tracked in SMB_ABTSRC

The register increments whenever data is placed into the receive FIFO and decrements when data is taken from the receive FIFO.

		BASE+0X78					
Bit		5	4	3	2	1	0
	Reserved	RXFLR					
RST		0	0	0	0	0	0

Bits	Name	Description	R/W
31:6	Reserved	Reserved.	N/A
5:0	RXFLR	Contains the number of valid data entries in the receive FIFO	R

29.2.2.29 SMB_SDAHD (SMB SDA Hold Time Register)

This register controls the amount of hold time on the SDA signal after a negative edge of SCL in both

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

master and slave mode, in units of APB clock period. The value programmed must be greater than the minimum hold time in each mode for the value to be implemented—one cycle in master mode, seven cycles in slave mode. Writes to this register succeed only when SMB_ENABLE[0]=0.

The programmed SDA hold time cannot exceed at any time the duration of the low part of scl. Therefore the programmed value cannot be larger than N_SCL_LOW-2, where N_SCL_LOW is the duration of the low part of the scl period measured in APB clock cycles.

	SMB_SDAHD																															BASE + 0x7C
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																SDAHD															
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	Reserved	Reserved.	N/A
15:0	SDAHD	SMB Hold Time. Sets the required SDA hold time in units of APB clock period.	RW

29.2.2.30 SMB_ABTSRC (SMB Transmit Abort Source Register)

This register has 32 bits that indicate the source of the TX_ABRT bit. Except for Bit 9, this register is cleared whenever the SMBCTXABT register or the SMBCINT register is read. To clear Bit 9, the source of the SBYTE_NORSTRT must be fixed first; RESTART must be enabled (SMB_CON[5]=1), the SPECIAL bit must be cleared (SMB_TAR[11]), or the GC_OR_START bit must be cleared (SMB_TAR[10]). Once the source of the SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the SBYTE_NORSTRT is not fixed before attempting to clear this bit, Bit 9 clears for one cycle and is then re-asserted.

	SMB_ABTSRC																															BASE + 0x80
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TX_FLUSH_CNT								Reserved							USER_ABRT	SLVRD_INTX	SLV_ARBLOST	SLVFLUSH_TXFIFO	ARB_LOST	ABRT_MASTER_DIS	ABRT_10B_RD_NORSTRT	SBYTE_NORSTRT	ABRT_HS_NORSTRT	SBYTE_ACKDET	ABRT_HS_ACKDET	ABRT_GCALL_READ	ABRT_GCALL_NOCK	ABRT_TXDATA_NOACK	ABRT_10ADDR2_NOACK	ABRT_10ADDR1_NOACK	ABRT_7B_ADDR_NOACK
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:24	TX_FLUSH_CNT	This field preserves the TXFLR value prior to the last TX_ABRT event. It is cleared whenever SMB is disabled.	R
23:17	Reserved	Read as zero.	
16	USER_ABRT	This is a master-mode-only bit. Master has detected the transfer abort	R
15	SLVRD_INTX	1: When the processor side responds to a slave mode request for data to be transmitted to a remote master and user writes a 1 in CMD (bit 8) of SMB_DC register.	R
14	SLV_ARBLOST	1: Slave lost the bus while transmitting data to a remote master. SMB_ABTSRC[12] is set at the same time. **NOTE:** Even though the slave never “owns” the bus, something could go wrong on the bus. This is a fail safe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then SMB no longer own the bus. Reset value: 0x0.	R
13	SLVFLUSH_TXFIFO	1: Slave has received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO. Reset value: 0x0.	R
12	ARB_LOST	1: Master has lost arbitration, or if SMB_ABTSRC[14] is also set, then the slave transmitter has lost arbitration. **NOTE:** SMB can be both master and slave at the same time. Reset value: 0x0.	R
11	ABRT_MASTER_DIS	1: User tries to initiate a Master operation with the Master mode disabled. Reset value: 0x0.	R
10	ABRT_10B_RD_NORSTRT	1: The restart is disabled (RESTART bit (SMB_CON[5]) = 0) and the master sends a read command in 10-bit addressing mode. Reset value: 0x0.	R
9	SBYTE_NORSTRT	To clear Bit 9, the source of the SBYTE_NORSTRT must be fixed first; restart must be enabled (SMB_CON[5]=1), the SPECIAL bit must be cleared (SMB_TAR[11]), or the GC_OR_START bit must be cleared (SMB_TAR[10]). Once the source of the SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets re-asserted. 1: The restart is disabled (RESTART bit (SMB_CON[5]) = 0) and the user is trying to send a START Byte. Reset value: 0x0.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8	ABRT_HS_NORSTRT	1: The restart is disabled (RESTART bit (SMB_CON[5]) = 0) and the user is trying to use the master to transfer data in High Speed mode. Reset value: 0x0.	R
7	SBYTE_ACKDET	1: Master has sent a START Byte and the START Byte was acknowledged (wrong behavior). Reset value: 0x0.	R
6	ABRT_HS_ACKDET	1: Master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior). Reset value: 0x0.	R
5	ABRT_GCALL_READ	1: SMB in master mode sent a General Call but the user programmed the byte following the General Call to be a read from the bus (SMBDC[9] is set to 1). Reset value: 0x0.	R
4	ABRT_GCALL_NOACK	1: SMB in master mode sent a General Call and no slave on the bus acknowledged the General Call. Reset value: 0x0.	R
3	ABRT_TXDATA_NOACK	1: This is a master-mode only bit. Master has received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledgement from the remote slave(s). Reset value: 0x0.	R
2	ABRT_10ADDR2_NOACK	1: Master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave. Reset value: 0x0.	R
1	ABRT_10ADDR1_NOACK	1: Master is in 10-bit address mode and the first 10-bit address byte was not acknowledged by any slave. Reset value: 0x0.	R
0	ABRT_7B_ADDR_NOACK	1: Master is in 7-bit addressing mode and the address sent was not acknowledged by any slave. Reset value: 0x0.	R

29.2.2.31 SMB_DMACR (SMB DMA Control Register)

The register is used to enable the DMA Controller interface operation. There is a separate bit for transmit and receive. This can be programmed regardless of the state of SMB_ENABLE.

	SMB_DMACR																															BASE + 0x88
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																														TDEN	RDEN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
15:2	Reserved	Writing has no effect, read as zero.	R
1	TDEN	**Transmit DMA Enable**. This bit enables/disables the transmit DMA channel. 0: Transmit DMA disabled 1: Transmit DMA enabled Reset value: 0x0.	R/W
0	RDEN	**Receive DMA Enable**. This bit enables/disables the receive DMA channel. 0: Receive DMA disabled 1: Receive DMA enabled Reset value: 0x0.	R/W

29.2.2.32 SMB_DMATDLR (SMB DMA Transmit Data Level Register)

	SMB_DMACR																BASE + 0x8c															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved										TDLR					
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:5	Reserved	Writing has no effect, read as zero.	R
4:0	TDLR	**DMA Transmit Data Level**. This bit field controls the level at which a DMA request is made by the transmit logic. The dma_tx_req signal is generated when the number of valid data entries in the transmit FIFO is equal to or below this field value.	R/W

29.2.2.33 SMB_DMARDLR (SMB DMA Transmit Data Level Register)

	SMB_DMACR																BASE + 0x90															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved										RDLR					
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:5	Reserved	Writing has no effect, read as zero.	R
4:0	RDLR	**DMA Receive Data Level**.	R/W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		This bit field controls the level at which a DMA request is made by the receive logic. The dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or more than this field value + 1.	

29.2.2.34 SMB_SDASU (SMB SDA Setup Register)

This register controls the amount of time delay (in terms of number of APB clock periods) introduced in the rising edge of SCL, relative to SDA changing, when SMB services a read request in a slave-transmitter operation. The relevant SMB requirement is $t_{SU:DAT}$(*NOTE 5*)as detailed in the I2C Bus Specification(Please Google "*UM10204 pdf*").

NOTE: The length of setup time is calculated using [(SMB_SDASU - 1) * (apb_clk_period)], so if the user requires 10 apb clock periods of setup time, they should program a value of 11. The SMB_SDA_SETUP register is only used by the SMB when operating as a slave transmitter

NOTE 5: A Fast-mode SMB-bus device can be used in a Standard-mode SMB-bus system, but the requirement $t_{SU;DAT}$ >= 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{r\ max}$ + $t_{SU;DAT}$ = 1000 +250 =1250 ns (according to the Standard-mode SMB-bus specification) before the SCL line is released. Also the acknowledge timing must meet this set-up time(Please refer NXP Semiconductors, *UM10204: I2C-Bus specification and user manual (Rev.6--4 April 2014), Table 10, Fig 38*)

	SMB_SDASU																BASE + 0x94															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved								SDASU							
RST																	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0

Bits	Name	Description	R/W
15:8	Reserved	Writing has no effect, read as zero.	R
7:0	SDASU	SDA Setup. It is recommended that if the required delay is 1000ns, then for an APB clock frequency of 10 MHz, SMBSDASU should be programmed to a value of 11.	R/W

29.2.2.35 SMB_ACKGC (SMB ACK General Call Register)

The register controls whether SMB responds with an ACK or NACK when it receives an SMB General Call address.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMB_ACKGC																BASE + 0x98															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																															ACKGC
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bits	Name	Description	R/W
15:1	Reserved	Writing has no effect, read as zero.	R
0	ACKGC	ACK General Call. When set to 1, SMB responds with an ACK when it receives a General Call. When set to 0, the SMB does not generate General Call interrupts.	R/W

29.2.2.36 SMB_ENBST (SMB Enable Status)

The register is used to report the SMB hardware status when the SMB_ENABLE[0] is set from 1 to 0; that is, when SMB is disabled.

If SMB_ENABLE[0] has been set to 1, bits 2:1 are forced to 0, and bit 0 is forced to 1.

If SMB_ENABLE[0] has been set to 0, bits 2:1 is only be valid as soon as bit 0 is read as ‘0’.

Note: When SMB_ENABLE[0] has been set to 0, a delay occurs for bit 0 to be read as 0 because disabling the SMB depends on SMB bus activities.

	SMB_ENBST																BASE + 0x9C															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																													SLVRDLST	SLVDISB	SMBEN
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
15:3	Reserved	Writing has no effect, read as zero.	R
2	SLVRDLST	Slave Received Data Lost. This bit indicates if a Slave-Receiver operation has been aborted with at least one data byte received from an SMB transfer due to the setting of SMB_ENABLE[0] from 1 to 0. When read as 1, SMB is deemed to have been actively engaged in an aborted SMB transfer (with matching address) and the data phase of the SMB transfer has been entered, even though a data byte has been responded with a NACK. **NOTE:** If the remote SMB master terminates the transfer with a STOP condition before the SMB has a chance to NACK a transfer, and SMB_ENABLE[0] has been set to 0, then this bit is also set to 1. When read as 0, SMB is deemed to have been disabled without being actively involved in the data phase of a Slave-Receiver	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		transfer. **NOTE:** The CPU can safely read this bit when SMB_ENABLE[0] is read as 0. Reset value: 0x0.	
1	SLVDISB	Slave Disabled While Busy (Transmit, Receive). This bit indicates if a potential or active Slave operation has been aborted due to the setting of the SMB_ENABLE register from 1 to 0. This bit is set when the CPU writes a 0 to the SMB_ENABLE register while: (a) SMB is receiving the address byte of the Slave-Transmitter operation from a remote master; OR, (b) address and data bytes of the Slave-Receiver operation from a remote master. When read as 1, SMB is deemed to have forced a NACK during any part of an SMB transfer, irrespective of whether the SMB address matches the slave address set in SMB (SMBSAR register) OR if the transfer is completed before SMBENB is set to 0 but has not taken effect. **NOTES:** 1 If the remote SMB master terminates the transfer with a STOP condition before the SMB has a chance to NACK a transfer, and SMBENB has been set to 0, then this bit will also be set to 1. When read as 0, SMB is deemed to have been disabled when there is master activity, or when the SMB bus is idle. 2 The CPU can safely read this bit when SMB_ENABLE[0] is read as 0. Reset value: 0x0.	R
0	SMBEN	When read as 1, SMB is deemed to be in an enabled state. When read as 0, SMB is deemed completely inactive.	R

29.2.2.37 SMB_FLT (SMB Filter Register)

This register is used to store the duration, measured in APB clock cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in SS or FS modes. This register must be programmed with a minimum value of 1.

	SMB_FLT																															BASE + 0xA0
Bit																										7	6	5	4	3	2	1	0
																										FLTCNT							
RST																										0	0	0	0	0	0	0	1

Bits	Name	Description	R/W
7:0	FLTCNT	This register sets the duration, measured in SMB device clock cycles,	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		of the longest value in the SCL or SDA lines that are filtered out by the spike suppression logic. This register can be written only when the SMB interface is disabled. Writes at other times have no effect. The minimum valid value is 1; hardware prevents values less than this being written, and if attempted, results in 1 being set.	

29.3 Operating Flow

This section provides information on the following topics:

- "Slave Mode Operation"
- "Master Mode Operation"
- "Disabling SMB"

NOTE: It is important to note that the SMB should only be set to operate as an SMB Master, or SMB Slave, but not both simultaneously. This is achieved by ensuring that bit 6 (SMBSLAVE_DISABLE) and 0 (SMBMASTER_MODE) of the SMBCON register are never set to 0 and 1, respectively.

29.3.1 SMB Behavior

The SMB can be controlled via software to be either:

- An SMB master only, communicating with other SMB slaves.

OR

- An SMB slave only, communicating with one more SMB masters.

The master is responsible for generating the clock and controlling the transfer of data. The slave is responsible for either transmitting or receiving data to/from the master. The device that is receiving data, which can be either a master or a slave, sends the acknowledgement of data. As mentioned previously, the SMB protocol also allows multiple masters to reside on the SMB bus and uses an arbitration procedure to determine bus ownership.

Each slave has a unique address that is determined by the system designer. When a master wants to communicate with a slave, the master transmits a START/RESTART condition that is then followed by the slave's address and a control bit (R/W) to determine if the master wants to transmit data or receive data from the slave. The slave then sends an acknowledgement (ACK) pulse after the address.

If the master (master-transmitter) is writing to the slave (slave-receiver), the receiver gets one byte of data. This transaction continues until the master terminates the transmission with a STOP condition. If the master is reading from a slave (master-receiver), the slave transmits (slave-transmitter) a byte of data to the master, and the master then acknowledges the transaction with the ACK pulse. This transaction continues until the master terminates the transmission by not acknowledging (NACK) the

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

transaction after the last byte is received, and then the master issues a STOP condition or addresses another slave after issuing a RESTART condition.

29.3.2 Master Mode Operation

This section includes the following topics:

- "Initial Configuration"
- "Dynamic SMBTAR or SMB10BITADDR_MASTER Update"
- "Master Transmit and Master Receive"

29.3.2.1 Configuration

To use the SMB as a master perform the following steps:

1. Disable the SMB by writing 0 to the SMBENB register. And wait for the SMBENBST.SMBEN = 0.
2. Write to the SMB_CON register to set the speed mode supported (bits 2:1). Please note that the MATP (bit4) is NOT writable. The addressing mode is controlled by Register SMB_TAR.
3. Set the expected SCL frequency. SMB_CON.SPEED = 2'b01, only SMB_SHCNT and SMBSLCNT are needed to be configured; SMB_CON.SPEED = 2'b10, only SMB_FHCNT and SMB_FLCNT are needed to be configured.

   Supposed:

   T_{scl} : SMB SCL period

   T_{SMB_clk} : SMB device clock period

   $T_{min_scl_l}$: Protocol minimum SCL low time

   $T_{min_scl_h}$: Protocol minimum SCL high time

   Then can get the equation:

   $T_{scl} = T_{SMB_clk} * ((SMB*HCNT + 8) + (SMB*LCNT + 1))$

   And the following conditions should be met:

   $(SMB*HCNT + 8) * T_{SMB_clk} >= T_{min_scl_h}$

   $(SMB*LCNT + 1) * T_{SMB_clk} >= T_{min_scl_l}$

   SMB*HCNT >= 6

   SMB*LCNT >= 8
4. Write to the SMB_TAR register the address of the SMB device to be addressed. It also indicates whether a General Call or a START BYTE command is going to be performed by SMB. The addressing mode that master starts, either 7-bit or 10-bit addressing, is controlled by the SMB10BITADDR_MASTER bit field (bit 12).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5 Enable the SMB by writing a 1 into SMB_ENABLE register. And wait for the SMB_ENBST.SMBEN = 1.

6 Now write the transfer direction and data to be sent to the SMB_DC register (This can be done by DMA). If the SMB_DC register is written before the SMB is enabled, the data and commands are lost as the buffers are kept cleared when SMB is not enabled.

NOTE: For multiple SMB transfers, perform additional writes to the TX FIFO such that the TX FIFO does not become empty during the SMB transaction. If the TX FIFO is completely emptied at any stage and SMB_CON.STPHLD is 0, then further writes to the TX FIFO results in an independent SMB transaction.

29.3.2.2 Dynamic SMBTAR or SMB10BITADDR_MASTER Update

The SMB supports dynamic updating of the SMBTAR (bits 9:0) and SMB10BITADDR_MASTER (bit 12) bit fields of the SMBTAR register. You can dynamically write to the SMBTAR register provided the following conditions are met:

1 SMB is not enabled (SMBENB=0);

OR

2 SMB is enabled (SMBENB=1);

AND

SMB is NOT engaged in any Master (tx, rx) operation (SMBST[5]=0);

AND

SMB is enabled to operate in Master mode (SMBCON[0]=1);

AND

there are NO entries in the TX FIFO (SMBST[2]=0).

29.3.2.3 Master Transmit and Master Receive

The SMB supports switching back and forth between reading and writing dynamically. To transmit data, write the data to be transferred to lower byte of the SMB Rx/Tx Data Buffer and Command Register (SMB_DC). The *CMD* bit (SMB_DC[8]) should be written to 0 for SMB write operations.

Subsequently, a read command may be issued by writing “don’t cares” to the lower byte of the SMBDC register, and a 1 should be written to the *CMD* bit.

29.3.3 Slave Mode Operation

This section includes the following procedures:

- “Initial Configuration”
- “Slave-Transmitter Operation for a Single Byte”
- “Slave-Receiver Operation for a Single Byte”
- “Slave-Transfer Operation For Bulk Transfers”

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.3.3.1 Initial Configuration

To use the SMB as a slave, perform the following steps:

1. Disable the SMB by writing a ‘0’ to bit 0 of the SMBENB register.
2. Write to the SMB_SAR register (bits 9:0) to set the slave address. This is the address to which the SMB responds.
3. Write to the SMB_CON register to specify which type of addressing is supported (7- or 10-bit by setting bit 3). Enable the SMB in slave-only mode by writing a ‘0’ into bit 6 (SMBSLAVE_DISABLE) and a ‘0’ to bit 0 (MASTER_MODE).

   NOTE: Slaves and masters do not have to be programmed with the same type of addressing 7- or 10-bit address. For instance, a slave can be programmed with 7-bit addressing and a master with 10-bit addressing, and vice versa.

4. Enable the SMB by writing a ‘1’ in bit 0 of the SMB_ENB register.

   NOTE: Depending on the reset values chosen, steps 2 and 3 may not be necessary because the reset values can be configured. For instance, if the device is only going to be a master, there would be no need to set the slave address because you can configure SMB to have the slave disabled after reset and to enable the master after reset. The values stored are static and do not need to be reprogrammed if the SMB is disabled.

29.3.3.2 Slave-Transmitter Operation for a Single Byte

When another SMB master device on the bus addresses the SMB and requests data, the SMB acts as a slave-transmitter and the following steps occur:

1. The other SMB master device initiates an SMB transfer with an address that matches the slave address in the SMB_SAR register of the SMB.
2. The SMB acknowledges the sent address and recognizes the direction of the transfer to indicate that it is acting as a slave-transmitter.
3. The SMB asserts the RDREQ interrupt (bit 5 of the SMBINTST register) and holds the SCL line low. It is in a wait state until software responds.
   If the RDREQ interrupt has been masked, due to SMB_INTM[5] register (MRDREQ bit field) being set to 0, then it is recommended that a hardware and/or software timing routine be used to instruct the CPU to perform periodic reads of the SMB_INTST register.
   a. Reads that indicate SMB_INTST[5] (RDREQ bit field) being set to 1 must be treated as the equivalent of the RDREQ interrupt being asserted.
   b. Software must then act to satisfy the SMB transfer.
   c. The timing interval used should be in the order of 10 times the fastest SCL clock period the SMB can handle. For example, for 400 kb/s, the timing interval is 25us.

   NOTE: The value of 10 is recommended here because this is approximately the amount of time required for a single byte of data transferred on the SMB bus.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 If there is any data remaining in the TX FIFO before receiving the read request, then the SMB asserts a TX_ABRT interrupt (bit 6 of the SMBINTST register) to flush the old data from the TX FIFO.

NOTE: Because the SMB's TX FIFO is forced into a flushed/reset state whenever a TX_ABRT event occurs, it is necessary for software to release the SMB from this state by reading the SMB_CTXABT register before attempting to write into the TX FIFO. See register SMB_INTST for more details. If the TX_ABRT interrupt has been masked, due to of SMB_INTM[6] register (MTX_ABRT bit field) being set to 0, then it is recommended that re-using the timing routine (described in the previous step), or a similar one, be used to read the SMBINTST register.

a Reads that indicate bit 6 (TXABT) being set to 1 must be treated as the equivalent of the TX_ABRT interrupt being asserted.
b There is no further action required from software.
c The timing interval used should be similar to that described in the previous step for the SMBINTST[5] register.

5 Software writes to the SMB_DC register with the data to be written (by writing a '0' in bit 8).
6 Software must clear the RDREQ and TX_ABRT interrupts (bits 5 and 6, respectively) of the SMBINTST register before proceeding.
If the RDREQ and/or TX_ABRT interrupts have been masked, then clearing of the SMB_INTST register will have already been performed when either the RDREQ or TXABT bit has been read as 1.
7 The SMB releases the SCL and transmits the byte.
8 The master may hold the SMB bus by issuing a RESTART condition or release the bus by issuing a STOP condition.

29.3.3.3 Slave-Receiver Operation for a Single Byte

When another SMB master device on the bus addresses the SMB and is sending data, the SMB acts as a slave-receiver and the following steps occur:

1 The other SMB master device initiates an SMB transfer with an address that matches the SMB's slave address in the SMBSAR register.
2 The SMB acknowledges the sent address and recognizes the direction of the transfer to indicate that the SMB is acting as a slave-receiver.
3 SMB receives the transmitted byte and places it in the receive buffer.

NOTE: If the RX FIFO is completely filled with data when a byte is pushed, then an overflow occurs and the SMB continues with subsequent SMB transfers. Because a NACK is not generated, software must recognize the overflow when indicated by the SMB (by the RXOF bit in the SMBINTST register) and take appropriate actions to recover from lost data. Hence, there is a real time constraint on software to service the RX FIFO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

before the latter overflow as there is no way to re-apply pressure to the remote transmitting master. You must select a deep enough RX FIFO depth to satisfy the interrupt service interval of their system.

4 SMB asserts the RX_FULL interrupt (SMBINTST[2] register).

If the RX_FULL interrupt has been masked, due to setting SMB_INTM[2] register to 0 or setting SMB_TX_TL to a value larger than 0, then it is recommended that a timing routine be implemented for periodic reads of the "SMBST". Reads of the SMB_ST register, with bit 3 (RFNE) set at 1, must then be treated by software as the equivalent of the RX_FULL interrupt being asserted.

5 Software may read the byte from the SMB_DC register (bits 7:0).

6 The other master device may hold the SMB bus by issuing a RESTART condition or release the bus by issuing a STOP condition.

29.3.3.4 Slave-Transfer Operation For Bulk Transfers

In the standard SMB protocol, all transactions are single byte transactions and the programmer responds to a remote master read request by writing one byte into the slave's TX FIFO.

When a slave (slave-transmitter) is issued with a read request (RDREQ) from the remote master (master-receiver), at a minimum there should be at least one entry placed into the slave-transmitter's TX FIFO.

SMB is designed to handle more data in the TX FIFO so that subsequent read requests can take that data without raising an interrupt to get more data. Ultimately, this eliminates the possibility of significant latencies being incurred between raising the interrupt for data each time had there been a restriction of having only one entry placed in the TX FIFO.

This mode only occurs when SMB is acting as a slave-transmitter. If the remote master acknowledges the data sent by the slave-transmitter and there is no data in the slave's TX FIFO, the SMB holds the SMB SCL line low while it raises the read request interrupt (RDREQ) and waits for data to be written into the TX FIFO before it can be sent to the remote master.

If the RDREQ interrupt is masked, due to bit 5 (MRDREQ) of the SMB_INTST register being set to 0, then it is recommended that a timing routine be used to activate periodic reads of the SMB_INTST register. Reads of SMB_INTST that return bit 5 (RDREQ) set to 1 must be treated as the equivalent of the RDREQ interrupt referred to in this section.

The RDREQ interrupt is raised upon a read request, and like interrupts, must be cleared when exiting the interrupt service handling routine (ISR). The ISR allows you to either write 1 byte or more than 1 byte into the TX FIFO. During the transmission of these bytes to the master, if the master acknowledges the last byte. Then the slave must raise the RDREQ again because the master is requesting for more data.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.3.4 Disabling SMB

The register SMB_ENBST is added to allow software to unambiguously determine when the hardware has completely shutdown in response to the SMB_ENABLE register being set from 1 to 0.

29.3.4.1 Procedure

1 Define a timer interval (tSMB_poll) equal to the 10 times the signaling period for the highest SMB transfer speed used in the system and supported by SMB. For example, if the highest SMB transfer mode is 400 kb/s, then this tSMB_poll is 25us.

2 Define a maximum time-out parameter, MAX_T_POLL_COUNT, such that if any repeated polling operation exceeds this maximum value, an error is reported.

3 Execute a blocking thread/process/function that prevents any further SMB master transactions to be started by software, but allows any pending transfers to be completed.
NOTE: This step can be ignored if SMB is programmed to operate as an SMB slave only.

4 The variable POLL_COUNT is initialized to zero.

5 Set SMBENB to 0.

6 Read the SMB_ENBST register and test the SMB_ENABLE[0]. Increment POLL_COUNT by one. If POLL_COUNT >= MAX_T_POLL_COUNT, exit with the relevant error code.

7 If SMB_ENBST[0] is 1, then sleep for tSMB_poll and proceed to the previous step. Otherwise, exit with a relevant success code.

29.3.5 Summary the condition could flush TX FIFO

1. When IC_ENABLE[0] is set to 0, the TX FIFO is flushed and held in reset.
2. TX_ABRT event occurs. The TX FIFO remains in this flushed state until the register SMB_CTXABT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.
3. When act as a slave, the slave has received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO.

29.3.6 The condition could generate START, STOP and RESTART

A STOP condition is generated only when the user specifically requests it by setting bit 9 (Stop bit) of the command written to SMB_DC register. Naturally, if the last byte send out does not set bit 9 (Stop bit), the component does not generate a STOP if the Tx FIFO becomes empty; in this situation the component holds the SCL line low, stalling the bus until a new entry is available in the Tx FIFO

Figure 29-1 illustrates the behavior of the SMB when the Tx FIFO becomes empty while operating as a master transmitter, as well as showing the generation of a STOP condition.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 29-1 Master Transmitter — Tx FIFO Empties/STOP Generation

Figure 29-2 illustrates the behavior of the SMB when the Tx FIFO becomes empty while operating as a master receiver, as well as showing the generation of a STOP condition.

Figure 29-2 Master Receiver — Tx FIFO Empties/STOP Generation

Figure 29-3 and Figure 29-4 illustrate configurations where the user can control the generation of RESTART conditions on the SMB bus. If bit 10 (Restart) of the SMB_DC register is set and the restart capability is enabled (IC_RESTART_EN=1), a RESTART is generated before the data byte is written to or read from the slave. If the restart capability is not enabled a STOP followed by a START is generated in place of the RESTART. Figure 29-3 illustrates this situation during operation as a master transmitter.

Figure 29-3 Master Transmitter — Restart bit of SMB_DC is set

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 29-4 illustrates the same situation, but during operation as a master receiver.

Figure 29-4 Master Receiver — Restart bit of SMB_DC is set

Figure 29-5 illustrates operation as a master transmitter where the Stop bit of the SMB_DC register is set and the Tx FIFO is not empty

Figure 29-5 Master Transmitter — Stop Bit of SMB_DC Set/Tx FIFO Not Empty

Figure 29-6 illustrates operation as a master transmitter where the first byte loaded into the Tx FIFO is allowed to go empty with the Restart bit set.

Figure 29-6 Master Transmitter — First Byte Loaded Into Tx FIFO Allowed to Empty, Restart Bit Set

Figure 29-7 illustrates operation as a master receiver where the Stop bit of the SMB_DC register is set and the Tx FIFO is not empty

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 29-7 Master Receiver — Stop Bit of IC_DATA_CMD Set/Tx FIFO Not Empty

Figure 29-8 illustrates operation as a master receiver where the first command loaded after the Tx FIFO is allowed to empty and the Restart bit is set

Figure 29-8 Master Receiver — First Command Loaded After Tx FIFO Allowed to Empty/Restart Bit Set

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30 Synchronous Serial Interface

30.1 Overview

The SSI is a full-duplex synchronous serial interface and can connect to a variety of external analog-to-digital (A/D) converters, audio and telecom codecs, and other devices that use serial protocols for transferring data. The SSI supports National's Microwire, Texas Instruments Synchronous Serial Protocol (SSP), and Motorola's Serial Peripheral Interface (SPI) protocol(Standard, Dual and Quad mode).

The SSI operates in master mode (the attached peripheral functions as a slave) and supports serial bit rates from 7.2 KHz to 50 MHz. Serial data formats may range from 2 to 32 bits in length. The SSI provides 128 entries deep x 32 bits wide transmit and receive data FIFOs.

The FIFOs may be loaded or emptied by the Central Processor Unit (CPU) using programmed I/O, or DMA transfers while receiving or transmitting.

Features:

- 3 protocols support: National's Microwire, TI's SSP, and Motorola's SPI
- Standard, Dual, Quad SPI
- Full-duplex or transmit-only or receive-only operation
- Programmable transfer order: MSB first or LSB first
- 128 entries deep x 32 bits wide transmit and receive data FIFOs
- Configurable normal transfer mode or Interval transfer mode
- Programmable clock phase and polarity for Motorola's SSI format
- Two slave select signal (SSI_CE0_ / SSI_CE1_) supporting up to 2 slave devices
- Back-to-back character transmission/reception mode
- Loop back mode for testing (except dual and quad SPI mode)

30.2 Pin Description

Table 30-1 SSI Controller Pins Description

Name	I/O	Description
SSI_CLK	Output	Serial bit-rate clock
SSI_CE0_	Output	First slave select enable
SSI_DT/SSI_IO0	Output/Inout	Transmit data/parallel data in and out(for Dual and Quad SPI mode)
SSI_DR/SSI_IO1	Input/Inout	Receive data / parallel data in and out(for Dual and Quad SPI mode)
SSI_CE1_/ SSI_IO2	Output/ Inout	Second slave select enable/ parallel data in and out (used in Quad SPI)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SSI_GPC/ SSI_IO3	Output/ Inout	General purpose control signal to external chip/ parallel data in and out (used in Quad SPI)

SSI_CLK is the bit-rate clock driven from the SSI to the peripheral. SSI_CLK is toggled only when data is actively being transmitted and received.

SSI_CE0_ or SSI_CE1_ are the frame signal, indicating the begin and the end of a serialized data word.

SSI_DT and SSI_DR are the Transmit and Receive serial data lines(used in the standard SPI, SSP and NM protocol).

SSI_GPC is general-purpose control signal, synchronized with SSI_CLK, can be used for LCD control.

SSI_IO0 - SSI_IO3 are both transmit or receive data lines (used in the Dual or Quad SPI mode)

30.3 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read
   SPEC - special access method, relate to its description
3. Reset Value
   1 - reset to 1
   0 - reset to 0
   ? - value unknown after reset

30.3.1 Registers Mapping

Table 30-2 Registers Memory Map-Address Base

Name	Base	Description
SSI0	0x10043000	Address base of SSI0
SSI1	0x10044000	Address base of SSI1

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The SSI has the following registers: one data, two control, one status, one bit-rate control, and two interval control registers. The table lists these registers.

Table 30-3 SSI0 Serial Port Registers

Name	Description	Reset Value	Address Offset	Access Size
SSI.SSIDR	Date register	0x????????	0x0000	32
SSI.SSICR0	Controller register 0	0x00000000	0x0004	32
SSI.SSICR1	Controller register 1	0x00087830	0x0008	32
SSI.SSISR	Status register	0x00000098	0x000C	32
SSI.SSIITR	Interval mode controller register	0x????0000	0x0010	16
SSI.SSIICR	Interval character-per-frame controller register	0x??????00	0x0014	8
SSI.SSIGR	Bit-rate controller register	0x????0000	0x0018	16
SSI.SSIRCNT	Receive counter register	0x00000000	0x001C	32

30.3.2 SSI Data Register (SSIDR)

	SSIDR																																BASE + 0x0000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D31-D17															GPC/D16	D15-D0															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:17	D31-D17	D31-D17 to be written to/read from Transmit/Receive FIFO.	RW
16	GPC/D16	This bit can be used as normal data bus bit 16 or GPC bit alternatively. When it is used as normal data bus bit, it's readable / writable; when SSI_GPC is used, it is GPC bit for SSI_GPC pin output and it's write-only.	RW
15:0	D15-D0	D15-D0 to be written to/read from Transmit/Receive FIFO. When the transfer frame length is less than 17-bit, received data is automatically right justified in the receive-FIFO and the upper unused bits are filled with '0'. For transmission, the upper unused bits of the data written into SSIDR is ignored by the transmit logic. **NOTE:** "upper unused bits" does not include the SSIDR.GPC bit. National Microwire format includes format 1 and format2, when national Microwire format 2 is selected, Bit 16 of SSIDR is defined as read/write operation judge bit, if it is 0, bit 15~0 represent one read command; if it is 1, bit 15~0 represent one write command and following is the written	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		data. So the maximum length of one command (is defined in MCOM) is 16, the maximum length of one written or read data (is defined in FLEN) can be 17. Transmit-FIFO only contain one read operation command once, or one write operation command and its data once, after transmit-FIFO is empty, next command can be filled in transmit-FIFO.	

30.3.3 SSI Control Register0 (SSICR0)

	SSICR0																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							BLIE	FIE	TFIE	RFIE	Reserved	TENDIAN		RENDIAN		SSIE	TIE	RIE	TEIE	REIE	LOOP	RFINE	RFINC	EACLRUN	FSEL	Reserved	VRCNT	TFMODE	TFLUSH	RFLUSH	DISREV
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Writing has no effect, read as '0'.	R
24	BLIE	This bit enables/disables the bit length error interrupt BLI(In Dual and Quad mode, this interrupt will be generated when the bit length is not proper)	RW
23	FIE	This bit enables/disables the transfer finished interrupt XFI. 0: disable; 1: enable.	RW
22	TFIE	This bit enables/disables the transmit-over interrupt TFI. 0: disable; 1: enable.	RW
21	RFIE	This bit enables/disables the receive-under interrupt RFI. 0: disable; 1: enable.	RW
20	Reserved	Writing has no effect, read as zero.	R
19:18	TENDIAN	Transfer endian 0: Most significant byte in a word first, most significant bit in a byte first. 1: Most significant byte in a word first, least significant bit in a byte first. 2: Least significant byte in a word first, least significant bit in a byte first. 3: Least significant byte in a word first, most significant bit in a byte first.	
17:16	RENDIAN	Receive endian 0: Most significant byte in a word first, most significant bit in a byte first. 1: Most significant byte in a word first, least significant bit in a byte first. 2: Least significant byte in a word first, least significant bit in a byte first. 3: Least significant byte in a word first, most significant bit in a byte first	RW
15	SSIE	This bit is used to enable/disable SSI module. 0: disable; 1: enable. Clearing SSIE will not reset SSI FIFO, SSICR0, SSICR1, SSIGR, SSIITR and SSIICR automatically. Software should ensure the	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td colspan="2">FIFOs/registers are properly configured and be flush/reset manually when necessary before enabling SSI.</td><td></td></tr>
<tr><td>14</td><td>TIE</td><td colspan="2">This bit enables/disables the transmit-FIFO half-empty interrupt TXI and the tdreq signal to PDMA.
0: disable; 1: enable.</td><td>RW</td></tr>
<tr><td>13</td><td>RIE</td><td colspan="2">This bit enables/disables the receive-FIFO half-full interrupt RXI and the rdreq signal to PDMA.
0: disable; 1: enable.</td><td>RW</td></tr>
<tr><td>12</td><td>TEIE</td><td colspan="2">This bit enables/disables the transmit-error interrupt TEI.
0: disable; 1: enable.</td><td>RW</td></tr>
<tr><td>11</td><td>REIE</td><td colspan="2">This bit enables/disables the receive-error interrupt REI.
0: disable; 1: enable.</td><td>RW</td></tr>
<tr><td>10</td><td>LOOP</td><td colspan="2">Used for test purpose. In loop mode, the output of SSI transmit shift register is connected to input of SSI receive shift register internally. The data received should be the same as the data transmitted. And do not output any valid signals on the pins.
0: normal SSI mode; 1: LOOP mode.</td><td>RW</td></tr>
<tr><td>9</td><td>RFINE</td><td>Receive finish control enable.
0: disable; 1: enable.
For SSICR1.FMAT = B’010 (National Microwire format 1 is selected), SSICR0.RFINE must be 0.</td><td>The receive finish condition list below:
<table><tr><th>RFINE</th><th>RFINC</th><th>Receive Finish Condition</th></tr><tr><td>0</td><td>x</td><td>Same as transmit completion condition (transmit-fifo is empty and SSICR1.UNFIN = 0)</td></tr><tr><td>1</td><td>0</td><td>Receive continue</td></tr><tr><td>1</td><td>1</td><td>Receive finish</td></tr></table></td><td>RW</td></tr>
<tr><td>8</td><td>RFINC*</td><td>Receive finish control bit.
0: receive continue
1: receive finished</td><td></td><td>RW</td></tr>
<tr><td>7</td><td>EACLRUN</td><td colspan="2">0: don’t auto clear under flag, software clear under
1: software auto clear under flag when tfifo don’t empty</td><td>RW</td></tr>
<tr><td>6</td><td>FSEL</td><td colspan="2">This bit sets the frame signal to be used for slave select. The unselected frame signal always output invalid level.
0: SSI_CE0_ is selected
1: SSI_CE1_ is selected</td><td>RW</td></tr>
<tr><td>5</td><td>Reserved</td><td colspan="2">Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>4</td><td>VRCNT</td><td colspan="2">Receive counter (RCNT) valid flag.
0: SSIRCNT.RCNT will be ignored.
1: SSIRCNT.RCNT will be used.</td><td>RW</td></tr>
<tr><td>3</td><td>TFMODE</td><td colspan="2">0: new tfifo empty mode(with Grey code convert, recommended)
1: old tfifo empty mode(not recommended)</td><td>RW</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2	TFLUSH	Flush the transmit FIFO when set to 1. Always return 0 after one cycle of pclk.	RW
1	RFLUSH	Flush the receive FIFO when set to 1. Always return 0 after one cycle of pclk	RW
0	DISREV	Receive function enable. 0: enable. 1: disable.	RW

NOTE:

***:** When transmitting finished or for receive-only operation, transmit function can be disabled and this bit is used to control receiving completion, and the SSI will consume less power.

When the finish condition is set, the receiving will complete after present character is completely shifted in, then the SSI will stop the SSI_CLK and negate the SSI_CE0_ / SSI_CE1_ if necessary. To make sure present transfer is completed, user must read and get SSISR.END = 1 (or SSISR.BUSY = 0).

30.3.4 SSI Control Register1 (SSICR1)

	SSICR1																															BASE + 0x0008
8Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRMHL		TFVCK		TCKFI		GPCMD	ITFRM	UNFIN	FMAT			TTRG				MCOM				RTRG				FLEN					GPCHL	PHA	POL
RST	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	0	0	0	0

Bits	Name	Description	RW
31:30	FRMHL	Frame valid level select, FRMHL [1: 0] correspond to SSI_CE1_ and SSI_CE0_ respectively. **FRMHL[1:0]** / **Description** / 00 / SSI_CE0_ is low level valid and SSI_CE1_ is low level valid / Initial value 01 / SSI_CE0_ is high level valid and SSI_CE1_ is low level valid / 10 / SSI_CE0_ is low level valid and SSI_CE1_ is high level valid / 11 / SSI_CE0_ is high level valid and SSI_CE1_ is high level valid /	RW
29:28	TFVCK	Time from frame valid to clock start. When TFVCK = B'00, the time is fixed half or one SSI_CLK cycle according to SSICR1.POL and SSICR1.PHA configuration. For SSICR1.FMAT = B'001, SSICR1.TFVCK is ignored. **TFVCK[1:0]** / **Description** /	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			00	half or one SSI_CLK cycle delay time	Initial value	
			01	1 more SSI_CLK cycle delay is added		
			10	2 more SSI_CLK cycle delay is added		
			11	3 more SSI_CLK cycle delay is added		
27:26	TCKFI	Time from clock stop to frame invalid. When TCKFI = B'00, the time is fixed one SSI_CLK or half SSI_CLK cycle according to SSICR1.POL and SSICR1.PHA configuration. For SSICR1.FMAT = B'001, SSICR1.TFVCK is ignored.				RW
			TCKFI[1:0]	**Description**		
			00	half or one SSI_CLK cycle delay	Initial value	
			01	1 more SSI_CLK cycle delay is added		
			10	2 more SSI_CLK cycle delay is added		
			11	3 more SSI_CLK cycle delay is added		
25	GPCMD	SSI_GPC in normal mode or special mode, 0:normal; 1:special In normal mode, SSI_GPC level is related to SSIDR; In special mode, SSI_GPC can be configured, and its level is GPCHL.				RW
24	ITFRM	Frame during interval, selects if the Frame (SSI_CE0_ /SSI_CE1_) signal is negated or not during interval time at Interval Mode (SSICR1.FMAT = B'000(quad and dual spi?) and SSIITR.IVLTM ≠ H'0000). It's ignored at Normal Mode. 0: SSI_CE0_ /SSI_CE1_ de-asserts during interval time at Interval Mode 1: SSI_CE0_ /SSI_CE1_ keeps asserted during interval time at Interval Mode				RW
23	UNFIN	This bit controls whether the SSI finishes transmission or wait for data filling (underrun happen) after all data in transmit-FIFO are sent out during transfer. This bit must be cleared to 0 when SSICR1.FMAT = B'001. (TI's SSP format) 0: TxFIFO empty means end of transmission 1: Transmission didn't finish when transmit-FIFO is empty, SSI underrun error would occur and SSI waits for data filling; SSI_CLK and SSI_CE0_ /SSI_CE1_ keeps asserted, SSI_CLK stop at the current level **NOTE:** For TxFIFO empty before any transfer after SSI enabled, if SSICR1.UNFIN = 1 or SSICR0.RFINE = 0, SSI will wait till TxFIFO isn't empty then start to transfer and no underrun error will occur; if SSICR1.UNFIN = 0 and SSICR0.RFINE = 1, after transmit-FIFO become empty, SSI will start a receive-only transfer.				RW
22:20	FMAT	These bits set the operating transfer format.				RW
			FMAT	**Description**		
			000	Motorola's SPI format(standard)	Initial value	
			001	TI's SSP format		
			010	National Microwire 1 format		

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			011	National Microwire 2 format		
			100	Dual SPI format		
			101	Quad SPI format		
19:16	TTRG	These bits define the transmit-FIFO half-empty threshold value, which when equal or less characters left in transmit-FIFO, the SSISR.TFHE will be set to '1'. 0000: less than or equal to 1 n: less than or equal to nx8 NOTE: 0000 is not recommended				RW
15:12	MCOM	Defines the length of command. Only used when SSICR1.FMAT = b'010 or b'011 (National Microwire format 1 or 2 is selected). The command length is (MCOM + 1). Initial value is b'0111.				RW
11:8	RTRG	These bits define the receive-FIFO half-full threshold value, which when equal or more characters received in receive-FIFO, the SSISR.RFHF will be set to '1'. 0000: more than or equal to 1 n: more than or equal to nx8				RW
7:3	FLEN	These bits set the bit length of every character to be transmitted/received. The maximum data length can be configured is 32 bits. For data length longer than 17 bits (multiples of the SSICR1.FLEN configured length), the software should ensure properly processing. When SSI_GPC pin is used, the FLEN shouldn't be configured as B'01111 (17-bit data). When TI SSP mode is selected (FMAT = 2'b001), 2-bit data length (FLEN = 0) isn't supported. The bit length of one data = FLEN + 2 NOTE: In Dual SPI/Quad SPI, the bit length should be multiple of 2/4. If it is not proper, the BLI interrupt will be generated.				RW
2	GPCHL	SSI_GPC level. This bit is valid when GPCMD is high 0:low; 1:high				RW
1	PHA	This bit sets the phase of the SSI_CLK from the beginning of a data frame for SPI format (SSICR1.FMAT = B'000 or B'100 or B'101). 0: The leading edge of SSI_CLK is used to sample data from SSI_DR after the SSI_CE0_ /SSI_CE1_ goes valid, it is initial value 1: The leading edge of SSI_CLK is used to drive data onto SSI_DT after the SSI_CE0_ /SSI_CE1_ goes valid				RW
0	POL	SSI_CLK polarity for SPI format during idle state. (SSICR1.FMAT = B'000 or B'100 or B'101). 0: SSI_CLK keeps low level when idle 1: SSI_CLK keeps high level when idle				RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.3.5 SSI Status Register (SSISR)

	SSISR																															BASE + 0x000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved					TOVER	RUNDR	BLE	TFIFO-NUM								RFIFO-NUM								END	BUSY	TFF	RFE	TFHE	RFHF	TUNDR	ROVER
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0

Bits	Name	Description	RW
31:27	Reserved	Writing has no effect, read as zero.	R
26	TOVER	Transmit-FIFO overrun status, and new transmitting data will lose. 0: TFIFO overrun has not occurred 1: TFIFO overrun has occurred. When SSICR0.TFIE is set, it will generate the TFI interrupt. Write ‘0’ to clear this bit, writing ‘1’ has no effect	RW
25	RUNDR	Receive-FIFO underrun status. It will occur in the condition that R-FIFO is empty while reading SSIDR 0: Underrun has not occurred 1: Underrun has occurred; When SSICR0.RFIE is set, it will generate the RFI interrupt. Write ‘0’ to clear this bit, writing ‘1’ has no effect	RW
24	BLE	This bit denotes the bit length is proper or not in Dual and Quad SPI mode(bit length should be multiple of 2 in Dual mode, and multiple of 4 in Quad mode) When SSICR0.BLIE is set, it will the SSI BLI Write ‘0’ to clear this bit, writing ‘1’ has no effect	RW
23:16	TFIFO-NUM	These bits indicate the Characters Number in Transmit-FIFO.	R
15:8	RFIFO-NUM	These bits indicate the Characters Number in Receive-FIFO.	R
7	END	This bit indicates transfer end status. It is the inverse of SSISR.BUSY when transfer is in process, but it’ll keep cleared at interval time before transfer is completed. It’ll be set when transfer finished. When SSICR0.FIE is set, It will generate the XFI interrupt. This interrupt will be auto cleared to '0' after reading SSISR.	R
6	BUSY	This bit indicates SSI’s working status. 0: SSI is idle or at interval time 1: Transmission and/or reception is in process	R
5	TFF	This bit denotes transmit-FIFO is full or not. 0: Transmit-FIFO is not full 1: Transmit-FIFO is full	R
4	RFE	This bit denotes receive-FIFO is empty or not.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Receive-FIFO is not empty 1: Receive-FIFO is empty	
3	TFHE	This bit denotes whether the characters number in transmit-FIFO being less or equal to SSICR1.TTRG. 0: The data in transmit-FIFO is more than the condition set by SSICR1.TTRG 1: The data in transmit-FIFO meets the condition set by SSICR1.TTRG When SSICR0.TIE is set , it will generate the TXI interrupt	R
2	RFHF	This bit denotes whether the characters number in receive-FIFO being more or equal to the number set by SSICR1.RTRG. 0: The data in receive-FIFO is less than the condition set by SSICR1.RTRG 1: The data in receive-FIFO meets the condition set by SSICR1.RTRG When SSICR0.RIE is set, it will generate the RXI interrupt	R
1	TUNDR	Transmit-FIFO underrun status. When underrun happens, SSI set this bit and keeps the current status of SSI_CLK and SSI_CE0_/SSI_CE1_, waiting for transmit-FIFO filling. 0: Transmit-FIFO underrun has not occurred 1: Transmit-FIFO underrun was occurred When SSICR0.TEIE is set, it will generate the TEI interrupt. Write ‘0’ to clear this bit, writing ‘1’ has no effect	RW
0	ROVER	Receive-FIFO overrun status, new received data will lose. 0: Receive-FIFO overrun has not occurred 1: Receive-FIFO overrun was occurred; When SSICR0.REIE is set, it will generate the REI interrupt. Write ‘0’ to clear this bit, writing ‘1’ has no effect	RW

30.3.6 SSI Interval Time Control Register (SSIITR)

	SSIITR																																BASE + 0x0010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																CNTCLK	IVLTM														
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	CNTCLK	Counting clock source select.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Use SSI bit clock (SSI_CLK) as the interval counter clock source 1: Use 32K clock as the interval counter clock source	
14:0	IVLTM	Interval time set, set the cycle number of counting clock source for desired interval time. When SSIITR.IVLTM = 0x0000, normal mode is selected, and SSIITR.CNTCLK and SSIICR are ignored. When SSIITR.IVLTM ≠ 0x0000, interval mode(not supported by Dual and Quad mode) is selected. The interval time is calculated as follows: *Interval time ≈ [CNTCLK clock period] *[Value of IVLTM]* The actual interval time is as follow: When SSIITR.CNTCLK = 0: *Interval time = [CNTCLK clock period] * [Value of IVLTM] + 3 * dev_clk period* When SSIITR.CNTCLK = 1: *Interval time ≥ [CNTCLK clock period] * [Value of IVLTM + 1] + 1 * dev_clk period;* *Interval time ≤ [CNTCLK clock period] * [Value of IVLTM + 2] + 2 * dev_clk period*	RW

30.3.7 SSI Interval Character-per-frame Control Register (SSIICR)

	SSIICR																															BASE + 0x0014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								Reserved					ICC		
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:3	Reserved	Writing has no effect, read as zero.	R
2:0	ICC	Sets the fixed number of characters to be transmitted / received each time during SSI_CLK changing (and SSI_CE0_ / SSI_CE1_ asserting) in interval mode for SSICR1.FMAT = B'000 (Motorola's standard SPI format is selected). SSIICR is ignored for SSICR1.FMAT ≠ B'000. The desired transfer number of characters-per-frame is (SSIICR set value + 1).	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.3.8 SSI Clock Generator Register (SSIGR)

	SSIGR																															BASE + 0x0018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																Reserved								CGV							
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:8	Reserved	Writing has no effect, read as zero.	R
7:0	CGV	Sets the frequency of serial bit clock (SSI_CLK). The serial bit clock (SSI_CLK) is generated by dividing device-clock as follows: F_{SSI_CLK} = [Frequency of device clock] / (2 * (CGV + 1)) Device clock is generated in CPM module. The value in SSIGR can be set from 0 to 255, and initialized to 0x0000 on power-on reset.	RW

30.3.9 SSI Receive Counter Register (SSIRCNT)

	SSIRCNT																															BASE + 0x001C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																RCNT															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:0	RCNT	Number of data to be received. If SSICR1.UNFIN = 0 and SSICR0.RFINE = 1, after transmit-FIFO become empty, SSI will start a receive-only transfer. If SSICR0.VRCNT = 1, a special internal counter will be increased automatically. When the value of the internal counter equals to the RCNT, the receiving operation will be stopped automatically.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.4 Functional Description

Serial data is transferred between the processor and external peripheral through FIFO buffers in the SSI. Data transfers to system memory are handled by either the CPU (using programmed I/O) or by DMA. Operation is full duplex - separate buffers and serial data paths permit simultaneous transfers to and from the external peripheral.

Programmed I/O transmits and receives data directly between the CPU and the transmit/receive FIFO's. The DMA controller transfers data during transmit and receive operations between memory and the FIFO's.

Transmit data is written by the CPU or DMA to the SSI's transmit FIFO. The SSI then takes the data from the FIFO, serializes it, and transmits it via the SSI_DT signal to the peripheral. Data from the peripheral is received via the SSI_DR signal, converted to parallel words and is stored in the Receive FIFO. Read operations automatically target the receive FIFO, while write operations write data to the transmit FIFO. Both the transmit and receive FIFO buffers are 128 entries deep by 17 bits wide. As the received data fills the receive FIFO, a programmable threshold triggers an interrupt to the Interrupt Controller. If enabled, an interrupt service routine responds by identifying the source of the interrupt and then performs one or several read operations from the inbound (receive) FIFO buffer.

30.5 Data Formats

Four signals are used to transfer data between the processor and external peripheral. The SSI supports three formats: Motorola SPI, Texas Instruments SSP, and National Microwire. Although they have the same basic structure the three formats have significant differences, as described below:

1. SSI_CE0_/SSI_CE1_ varies for each protocol as follows:
   - For SPI and Microwire formats, SSI_CE0_/SSI_CE1_ functions as a chip select to enable the external device (target of the transfer), and is held active-low during the data transfer.
   - For SSP format, this signal is pulsed high for one serial bit-clock period at the start of each frame.
2. SSI_CLK varies for each protocol as follows:
   - For Microwire, both transmit and receive data sources switch data on the falling edge of SSI_CLK, and sample incoming data on the rising edge.
   - For SSP, transmit and receive data sources switch data on the rising edge of SSI_CLK, and sample incoming data on the falling edge.
   - For SPI, the user has the choice of which edge of SSI_CLK to use for switching outgoing data, and for sampling incoming data. In addition, the user can move the phase of SSI_CLK, shifting its active state one-half period earlier or later at the start and end of a frame.

While SSP and standard SPI are full-duplex protocols, Dual and Quad SPI and Microwire use a

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

half-duplex master-slave messaging protocol. At the start of a frame, a 1 or 2-byte control message is transmitted from the controller to the peripheral. The peripheral does not send any data. The peripheral interprets the message and, if it is a READ request, responds with requested data, one clock after the last bit of the requesting message.

In Dual and Quad SPI format, if the control message is required be sent by single data line, it should be expanded to 2bit or 4bit format. For example, 01001110 was required be sent by SSI_IO0, so it should be expanded to 0000_0001_0000_0000_0001_0001_0001_0000 for Quad mode, or 0001_0000_0101_0100 for Dual mode(it means that the unused data lines should also have its value)

The serial clock (SSI_CLK) only toggles during an active frame. At other times it is held in an inactive or idle state, as defined by its specified protocol.

30.5.1 Motorola's SPI Format Details

30.5.1.1 General Single Transfer Formats

The figures below show the timing of general single transfer format.

Figure 30-1 Standard SPI Single Character Transfer Format (PHA = 0)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-2 Standard SPI Single Character Transfer Format (PHA = 1)

Figure 30-3 Quad SPI Transfer Format (PHA = 1 and big endian)

Figure 30-4 Dual SPI Transfer Format (PHA = 1 and little endian)

For SSICR1.PHA = 0, when SSICR1.TFVCK = B'00, hardware ensures the first clock edge appears one SSI_CLK period after SSI_CE0_ / SSI_CE1_ goes valid; when SSICR1.TCKFI = B'00, hardware ensures the SSI_CE0_ / SSI_CE1_ negated half SSI_CLK period after last clock change edge; when SSICR1.TFVCK ≠ B'00 or SSICR1.TCKFI ≠ B'00, 1/2/3 more clock cycles are inserted.

For SSICR1.PHA = 1, when SSICR1.TFVCK = B'00, hardware ensures the first clock edge appears half SSI_CLK period after SSI_CE0_ / SSI_CE1_ goes valid; when SSICR1.TCKFI = B'00, hardware ensures the SSI_CE0_ / SSI_CE1_ negated one SSI_CLK period after last clock change edge; when SSICR1.TFVCK ≠ B'00 or SSICR1.TCKFI ≠ B'00, 1/2/3 more clock cycles are inserted.
Data is sampled from SSI_DR at every rising edge (when PHA = 0, POL = 0 or PHA = 1, POL = 1) or at every falling edge (when PHA = 0, POL = 1 or PHA = 1, POL = 0). According to Standard SPI protocol,

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

input data should be stable at every sample clock edge.

Drive data onto output line at every rising edge (when PHA = 0, POL = 1 or PHA = 1, POL = 0) or at every falling edge (when PHA = 0, POL = 0 or PHA = 1, POL = 1).

30.5.1.2 Back-to-Back Transfer Formats

Figure 30-5 Standard SPI Back-to-Back Transfer Format

Figure 30-6 Quad SPI Back-to-Back Transfer Format

For Motorola's SPI format transfers those continuous characters are exchanged during SSI_CE0_ / SSI_CE1_ being valid, the timing is illustrated in the figure (SSICR1.ITFRM = 0).

Back-to-back transfer is performed as transmit-only/full-duplex operation when transmit-FIFO is not empty before the completion of the last character's transfer or performed as receive-only operation.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.1.3 Frame Interval Mode Transfer Format

When in interval mode (SSIITR. IVLTM ≠ '0', not supported by Dual and Quad mode), SSI always wait for an interval time (SSIITR.IVLTM), transfer fixed number of characters (SSIICR), then repeats the operation.

When SSICR0.RFINE = 1, if transmit-FIFO is still empty after the interval time, receive-only transfer will occur.

During interval-wait time, SSI stops SSI_CLK, and when SSICR1.ITFRM = 0 it negates the SSI_CE0_ / SSI_CE1_, when SSICR1.ITFRM = 1 it keeps asserting the SSI_CE0_ / SSI_CE1_.

For transfers finished with transmit-FIFO empty, if the SSI transmit-FIFO is empty before fixed number of characters being loaded to transfer (SSICR1.UNFIN must be 1), then the SSI will set SSISR.UNDR = 1; if enabled, it'll send out a SSI underrun interrupt. At the same time, SSI will hold the SSI_CE0_ / SSI_CE1_ and SSI_CLK signals at current status and wait for the transmit-FIFO filling. The SSI will continue transfer after transmit-FIFO being filled. The SSI always stops after completion of fixed number of characters' transfer (SSICR1.UNFIN must be 0) with transmit-FIFO empty.

For transfers finished by SSICR0.RFINC being valid set, the SSI will stop after finished current character transfer and needn't wait for a whole completion of fixed number of characters' transfer.

Two Interval transfer mode are illustrated in the following figures. In these timing diagram, SSICR1.PHA = 0, SSICR1.POL = 0 and SSIICR = 0.

Figure 30-7 Standard SPI Frame Interval Mode Transfer Format (ITFRM = 0, ITFRM= 0)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-8 Standard SPI Frame Interval Mode Transfer Format (ITFRM = 1, ITFRM = 1)

30.5.2 TI's SSP Format Details

In this format, each transfer begins with SSI_CE0_ pulsed high for one SSI_CLK period. Then both master and slave drive data at SSI_CLK's rising edge and sample data at the falling edge. Data are transferred with MSB first or LSB first. At the end of the transfer, SSI_DT retains the value of the last bit sent through the next idle period.

Figure 30-9 TI's SSP Single Transfer Format

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-10 TI's SSP Back-to-back Transfer Format

30.5.3 National Microwire Format Details

It supports format 1 and format 2.

Format		Drive Edge	Sample Edge
NM1	Master	↓	↑
	Slave	↓	↑
NM2	Master	↓	↓
	Slave	↑	↑

If format 1 is selected, both master and slave drive data at SSI_CLK falling edge and sample data at the rising edge.

If format 2 is selected, master drive and sample data at SSI_CLK falling edge, slave drive and sample data at SSI_CLK rising edge.
SSI_CLK goes high midway through the command's MSB (or LSB) and continues to toggle at the bit rate. One bit clock (format 1) or half bit clock (format 2) period after the last command bit, the external slave must return the serial data requested, with MSB first (or LSB first) on SSI_DR.
SSI_CE0_ / SSI_CE2 de-asserts high half clock (SSI_CLK) period (and 1/2/3 additional clock periods) later.

Format 1 support back-to-back transfer, the start and end of back-to-back transfers are similar to those of a single transfer. However, SSI_CE0_ / SSI_CE2 remains asserted throughout the transfer. The end of a character data on SSI_DR is immediately followed by the start of the next command byte on SSI_DT.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-11 National Microwire Format 1 Single Transfer

Figure 30-12 National Microwire Format 1 Back-to-back Transfer

Figure 30-13 National Microwire Format 2 Read Timing

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 30-14 National Microwire Format 2 Write Timing

30.6 Interrupt Operation

In SSI, there are TXI, RXI, TEI, REI,TFI, RFI, XFI and BLI total 8 interrupts, all these interrupts are combined together to make one SSI interrupt, which can be masked by writing '1' into corresponding mask bit in INTC interrupt mask register (IMR).

Table 30-4 SSI Interrupts

Operation	Condition	Flag Bit	Mask Bit	Interrupt	DMAC Activation
Transmit	T-FIFO is half-empty or less	SSISR.TFHE	SSICR0.TIE	**TXI**	Possible
	Transmit underrun error	SSISR.TUNDR	SSICR0.TEIE	**TEI**	Impossible
	T-FIFO is full while writing SSIDR	SSISR.TOVER	SSICR0.TFIE	**TFI**	Impossible
Receive	R-FIFO is half-full or more	SSISR.RFHF	SSICR0.RIE	**RXI**	Possible
	Receive overrun error	SSISR.ROVER	SSICR0.REIE	**REI**	Impossible
	R-FIFO is empty while reading SSIDR	SSISR.RUNDR	SSICR0.RFIE	**RFI**	Impossible
Work end	All work is finished	SSISR.END	SSICR0.FIE	**XFI**	Impossible
Configure error	Bit length is mismatch in Dual or Quad SPI	SSISR.BLE	SSICR0.BLIE	**BLI**	Impossible

Either SSISR.TFHE or SSISR.RFHF can activate DMA transferring when corresponding individual interrupt mask bit in SSICR0 is cleared (masked) and DMA is enabled and configured.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.7 Software Guid-line

The initial steps are:

- set GPIO
- disable interrupt (or not) by setting SSICR0
- clear SSISR (write 0x0, clear interrupt, FIFO state and working condition)
- flush FIFO (write SSICR0.xFLUSH once for each FIFO)
- select the counting clock by configuring SSIITR
- configure SSIGR (set the frequency of serial bit clock)
- configure SSICR1 (set valid level of CE signals, set head and end delay of the transfer, set state of CE signals in interval time, select protocols, set threshold of FIFO, set bit length, etc.)
- configure SSIICR (set the fixed number of characters to be transmit/received each time during SSI_CLK changing in interval mode)
- configure SSIRCNT (number of data to be received)
- configure SSICR0 (the start trigger is in this register)

NOTE: the frequency of dev_clk should not be 4 times or more higher than pclk

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31 UART Interface

31.1 Overview

This chapter describes the universal asynchronous receiver/transmitter (UART) serial ports. There are five UARTs: All UARTs use the same programming model. Each of the serial ports can operate in interrupt based mode or DMA-based mode.

The Universal asynchronous receiver/transmitter (UART) is compatible with the 16550-industry standard and can be used as slow infrared asynchronous interface that conforms to the Infrared Data Association (IrDA) serial infrared specification 1.1.

31.1.1 Features

- Full-duplex operation
- 5-, 6-, 7- or 8-bit characters with optional no parity or even or odd parity and with 1, 1½, or 2 stop bits
- 64x8 bit transmit FIFO and 64x11bit receive FIFO
- Independently controlled transmit, receive (data ready or timeout), line status interrupts
- Internal diagnostic capability Loopback control and break, parity, overrun and framing-error is provided
- Separate DMA requests for transmit and receive data services in FIFO mode
- Supports modem flow control by software or hardware
- Slow infrared asynchronous interface that conforms to IrDA specification

31.1.2 Pin Description

Table 31-1 UART Pins Description

Name	Type	Description
RxD	Input	Receive data input
TxD	Output	Transmit data output
CTS_	Input	Clear to Send — Modem Transmission enabled
RTS_	Output	Request to Send — UART Transmission request

NOTE: UART4, UART3, UART2, UART1, UART0 support RxD, TxD, RTS_, CTS_.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.2 Register Descriptions

All UART register 32-bit access address is physical address. When ULCR.DLAB is 0, URBR, UTHR and UIER can be accessed; When ULCR.DLAB is 1, UDLLR and UDLHR can be accessed.

31.2.1 Register Mapping

There are 5 UART controllers in chip, name UART0, UART1, UART2, UART3, UART4 respectively.
The base address of UART0 is 0x10030000.
The base address of UART1 is 0x10031000.
The base address of UART2 is 0x10032000.
The base address of UART3 is 0x10033000.
The base address of UART4 is 0x10034000.

Table 31-2 UART Registers Description

Name	Description	RW	Reset Value	Offset Address	Access Size
URBR	UART Receive Buffer Register	R	0x??	0x000	8
UTHR	UART Transmit Hold Register	W	0x??	0x000	8
UDLLR	UART Divisor Latch Low Register	RW	0x00	0x000	8
UDLHR	UART Divisor Latch High Register	RW	0x00	0x004	8
UIER	UART Interrupt Enable Register	RW	0x00	0x004	8
UIIR	UART Interrupt Identification Register	R	0x01	0x008	8
UFCR	UART FIFO Control Register	W	0x00	0x008	8
ULCR	UART Line Control Register	RW	0x00	0x00c	8
UMCR	UART Modem Control Register	RW	0x00	0x010	8
ULSR	UART Line Status Register	R	0x00	0x014	8
UMSR	UART Modem Status Register	R	0x00	0x018	8
USPR	UART Scratchpad Register	RW	0x00	0x01c	8
ISR	Infrared Selection Register	RW	0x00	0x020	8
UMR	UART M Register	RW	0x00	0x024	8
UACR	UART Add Cycle Register	RW	0x00	0x028	16
URCR	UART RXFIFO Counter Register	R	0x00	0x040	8
UTCR	UART TXFIFO Counter Register	R	0x00	0x044	8

31.2.2 UART Receive Buffer Register (URBR)

The read-only URBR is corresponded to one level 11bit buffer in non-FIFO mode and a 32x11bit FIFO that holds the character(s) received by the UART. Bits in URBR are right justified when being configured to use fewer than eight bits, and the rest of most significant data bits are zeroed and the most significant three bits of each buffer are the status for the character in the buffer. If ULSR.DRY is 0, don’t read URBR, otherwise wrong operation may occur.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	URBR																															0x000 (DLAB = 0)
Bit																										7	6	5	4	3	2	1	0
																										URBR							
RST																										?	?	?	?	?	?	?	?

Bits	Name	Description	RW
7:0	URBR	8-bit UART receive read data.	R

31.2.3 UART Transmit Hold Register (UTHR)

The write-only UTHR is corresponded to one leve 8 bit buffer in non-FIFO mode and a 32x8bit FIFO in FIFO mode that holds the data byte(s) to be transmitted next.

	UTHR																															0x000 (DLAB = 0)
Bit																										7	6	5	4	3	2	1	0
																										UTHR							
RST																										?	?	?	?	?	?	?	?

Bits	Name	Description	RW
7:0	UTHR	8-bit UART transmit write hold data.	W

31.2.4 UART Divisor Latch Low/High Register (UDLLR / UDLHR)

UART Divisor Latch registers, UDLLR/UDLHR together compose the divisor for the programmable baud rate generator that can take the UART device clock and divide it by 1 to (2^{16} – 1).

The UART device source clock is EXCLK or EXCLK/2 that is determined by CPCCR.ECS. UDLHR/UDLLR stores the high/low 8-bit of the divisor respectively. Load these divisor latches during initialization to ensure that the baud rate generator operates properly. If both Divisor Latch registers are 0, the 16X clock stops.

If you don't set UMR and UACR, UART will work at normal mode with the specified frequency. The relationship between baud rate and the value of Divisor is shown by the formula when UMR and UACR are not set:

Baud Rate = (UART device clock) / (16 * **Divisor**)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	UDLLR																															0x000 (DLAB = 1)
Bit																													7	6	5	4	3	2	1	0
	Divisor Latch Low 8-bit																																			
RST																													0	0	0	0	0	0	0	0

	UDLHR																															0x004 (DLAB = 1)
Bit																													7	6	5	4	3	2	1	0
	Divisor Latch High 8-bit																																			
RST																													0	0	0	0	0	0	0	0

31.2.5 UART Interrupt Enable Register (UIER)

The UART Interrupt Enable Register (UIER) contains the interrupt enable bits for the five types of interrupts (receive data ready, timeout, line status, and transmit data request, and modem status) that set a value in UIIR.

	UIER																															0x004 (DLAB = 0)
Bit																													7	6	5	4	3	2	1	0
																													Reserved			RTOIE	MSIE	RLSIE	TDRIE	RDRIE
RST																													0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:5	Reserved	Writing has no effect, read as zero.	R
4	RTOIE	**Receive Timeout Interrupt Enable.** 0: Disable the receive timeout interrupt 1: Enable the receive timeout interrupt Timeout means the URDR (FIFO mode) is not empty but no character has received for a period of time T: T (bits) = 4 X Word length + 12.	RW
3	MSIE	**Modem Status Interrupt Enable.** 0: Disable the modem status interrupt 1: Enable the modem status interrupt	RW
2	RLSIE	**Receive Line Status Interrupt Enable.** 0: Disable receive line status interrupt 1: Enable receive line status interrupt	RW
1	TDRIE	**Transmit Data Request Interrupt Enable.**	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Disable the transmit data request interrupt 1: Enable the transmit data request interrupt	
0	RDRIE	**Receive Data Ready Interrupt Enable.** 0: Disable the receive data ready interrupt 1: Enable the receive data ready interrupt	RW

31.2.6 UART Interrupt Identification Register (UIIR)

The read-only UART Interrupt Identification Register (UIIR) records the prioritized pending interrupt source information. Its initial value after power-on reset is 0x01.

	UIIR																															0x008
Bit																									7	6	5	4	3	2	1	0
																									FFMSEL		Reserved		INID			INPEND
RST																									0	0	0	0	0	0	0	1

Bits	Name	Description	RW
7:6	FFMSEL	**FIFO Mode Select.** 0b00: Non-FIFO mode 0b01: Reserved 0b10: Reserved 0b11: FIFO mode	R
5:4	Reserved	Writing has no effect, read as zero.	R
3:1	INID	**Interrupt Identifier.** These bits identify the current highest priority pending interrupt. INID — Description 0b000 — Modem Status 0b001 — Transmit Data Request 0b010 — Receive Data Ready 0b011 — Receive Line Status 0b100 — Reserved 0b101 — Reserved 0b110 — Receive Time Out 0b111 — Reserved See Table 31-3 for details.	R
0	INPEND	**Interrupt Pending.** 0: interrupt is pending 1: No interrupt pending	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 31-3 UART Interrupt Identification Register Description

UIIR.INID	Interrupt Set/Clear Cause			
	Priority	**Type**	**Source**	**Clear Condition**
0b0001	—	None	No pending interrupt	—
0b0110	1st Highest	Receive Line Status	Overrun, Parity, Frame Error, Break Interrupt, and FIFO Error (DMA mode only)	Reading ULSR or empty all the error characters in DMA mode
0b0100	2nd Highest	Receive Data Ready	FIFO mode: Trigger threshold was reached Non-FIFO mode: URBR full	FIFO mode: Reading URBR till below trigger threshold. Non-FIFO mode: Empty URBR
0b1100	2nd Highest	Receive Timeout	FIFO mode only: URBR not empty but no data read in for a period of time	Reset receive buffer by setting UFCR.RFRT to 1 or Reading URBR
0b0010	3rd Highest	Transmit Data Request	FIFO mode: Empty location in UTHR equal to half or more than half Non-FIFO mode: UTHR empty	FIFO mode: Data number in UTHR more than half Non-FIFO mode: Writing UTHR
0b0000	4th Highest	Modem Status	Modem CTS_ pin status change	Reading UMSR

31.2.7 UART FIFO Control Register (UFCR)

The write-only register UFCR contains the control bits for receive and transmit FIFO.

	UFCR																													0x008
Bit																									7	6	5	4	3	2	1	0
																									RDTR		Reserved	UME	DME	TFRT	RFRT	FME
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:6	RDTR	**Receive Buffer Data Number Trigger.** These bits are used to select the trigger level for the receive data ready interrupt in FIFO mode. 0b00: 1 0b01: 16 0b10: 32 0b11: 60	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5	Reserved	Writing has no effect, read as zero.	R
4	UME	**UART Module Enable.** 0: Disable UART 1: Enable UART	W
3	DME	**DMA Mode Enable.** 0: Disable DMA mode 1: Enable DMA mode	W
2	TFRT	**Transmit Holding Register Reset.** 0: Not reset 1: Reset transmit FIFO	W
1	RFRT	**Receive Buffer Reset.** 0: Not reset 1: Reset receive FIFO	W
0	FME	**FIFO Mode Enable.** Set this bit before the trigger levels. 0: non-FIFO mode 1: FIFO mode	W

31.2.8 UART Line Control Register (ULCR)

The ULCR defines the format for UART data transmission.

	ULCR																															0x00C
Bit																										7	6	5	4	3	2	1	0
																										DLAB	SBK	STPAR	PARM	PARE	SBLS	WLS	
RST																										0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7	DLAB	**Divisor Latch Access Bit.** 0: Enable to access URBR, UTHR or UIER 1: Enable to access UDLLR or UDLHR	W
6	SBK	**Set Break.** Causes a break condition (at least one 0x00 data) to be transmitted to the receiving UART. Acts only on the TXD pin and has no effect on the transmit logic. 0: No effect on TXD output 1: Forces TXD output to 0	W
5	STPAR	**Sticky Parity.** Setting this bit forces parity location to be opposite of PARM bit when PARE is 1 (it is ignored when PARE is 0).	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Disable Sticky parity 1: Enable Sticky parity (opposite o f PARM bit)	
4	PARM	**Parity Odd/Even Mode Select.** If PARE = 0, PARM is ignored. 0: Odd parity 1: Even parity	W
3	PARE	**Parity Enable.** Enables a parity bit to be generated on transmission or checked on reception. 0: No parity 1: Parity	W
2	SBLS	**Stop Bit Length Select.** Specifies the number of stop bits transmitted and received in each character. When receiving, the receiver checks only the first stop bit. 0: 1 stop bit 1: 2 stop bits, except for 5-bit character then 1-1/2 bits	W
1:0	WLS	**Word Length Select.** 0b00: 5-bit character 0b01: 6-bit character 0b10: 7-bit character 0b11: 8-bit character	W

31.2.9 UART Line Status Register (ULSR)

The read-only ULSR indicates status information during the data transfer. Receive error information in ULSR[4:1] remains set until software reads ULSR and it must be read before the error character is read.

	ULSR																															0x014
Bit																										7	6	5	4	3	2	1	0
																										FIFOE	TEMP	TDRQ	BI	FMER	PARER	OVER	DRY
RST																										0	1	1	0	0	0	0	0

Bits	Name	Description	RW
7	FIFOE	**FIFO Error Status. (FIFO mode only)** FIFOE is set when there is at least one kind of receive error (parity, frame, overrun, break) for any of the characters in receive buffer. FIFOE is reset when all error characters are read out of the buffer. During DMA transfer, the error interrupt generates when FIFOE is 1, and	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		no receive DMA request generates even when data in receive buffer reaches the trigger threshold until all the error characters are read out. In non-DMA mode, FIFOE set does not generate error interrupt. 0: No error data in receive buffer or non-FIFO mode 1: One or more error character in receive buffer	
6	TEMP	**Transmit Holding Register Empty.** Set when both UTHR and shift register are empty. It is cleared when either the UTHR or the shift register contains a data character. 0: There is data in the transmit shifter and UTHR 1: All the data in the transmit shifter and UTHR has been shifted out	R
5	TDRQ	**Transmit Data Request.** Set when UTHR has half or more empty location (FIFO mode) or empty (non-FIFO mode). When both UIER.TDRIE and TDRQ are 1, transmit data request interrupt generates or during DMA transfer, DMA request to the DMA controller generates when UIER.TDRIE is 0 and TDRQ is 1. 0: There is one (non-FIFO mode) or more than half data (FIFO mode) in UTHR 1: None data (non-FIFO mode) or half or less than half data (FIFO mode) in UTHR	R
4	BI	**Break Interrupt.** BI is set when the received data input is held low for longer than a full-word transmission time (the total time of start bit + data bits + parity bit + stop bits). BI is cleared when the processor reads the ULSR. In FIFO mode, only one character equal to 0x00 is loaded into the FIFO regardless of the length of the break condition. BI shows the break condition for the character at the front of the FIFO, not the most recently received character. 0: No break signal has been received 1: Break signal received	R
3	FMER	**Framing Error.** Set when the bit following the last data bit or parity bit is detected to be 0. If the ULCR had been set for two or one and half stop bits, the other stop bits are not checked except the first one. In FIFO mode, FMER shows a framing error for the character at the front of the receive buffer, not for the most recently received character. Cleared when the processor reads the ULSR. 0: No framing error 1: Invalid stop bit has been detected	R
2	PARER	**Parity Error.**	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Indicates that the received data character does not have the correct even or odd parity, as selected by the even parity select bit. PARER is set upon detection if a parity error and is cleared when the processor reads the ULSR. In FIFO mode, PARER shows a parity error for the character at the front of the FIFO, not the most recently received character. 0: No parity error 1: Parity error has occurred	
1	OVER	**Overrun Error.** Set when both receive buffer and shifter are full and new data is received which will be lost. Cleared when the processor reads the ULSR. 0: No data has been lost 1: Receive data has been lost	R
0	DRY	**Data Ready.** Set when a complete incoming character has been received into the Receive Buffer registers. DRY is cleared when the receive buffer is read (non-FIFO mode) or when the buffer is empty or when the buffer is reset by setting UFCR.RFRT to 1. 0: No data has been received 1: Data is available in URBR	R

31.2.10 UART Modem Control Register (UMCR)

The UMCR uses the modem control pins RTS_ and CTS_ to control the interface with a modem or data set. UMCR also controls the loopback mode. Loopback mode must be enabled before the UART is enabled.

	UMCR																															0x010
Bit																										7	6	5	4	3	2	1	0
																										MDCE	FCM	Reserved	LOOP	Reserved		RTS	Reserved
RST																										0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7	MDCE	**Modem Control Enable.** 0: Modem function is disabled 1: Modem function is enabled	W
6	FCM	**Flow Control Mode.** 0: Flow control by software 1: Flow control by hardware	
5	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4	LOOP	**Loop Back.** This bit is used for diagnostic testing of the UART. When LOOP is 1, TXD output pin is set to a logic 1 state, RXD is disconnected from the pin, and the output of the transmitter shifter register is looped back into the receiver shift register input internally, similar to CTS_ and RTS_ pins and the RTS bit of the UMCR is connected to CTS bit of UMSR respectively. Loopback mode must be selected before the UART is enabled. 0: Normal operation mode 1: Loopback-mode UART operation	W
3:2	Reserved	Writing has no effect, read as zero.	R
1	RTS	**Request To Send.** This bit can control the RTS_ output state. 0: RTS_ force to high 1: RTS_ force to low	W
0	Reserved	Writing has no effect, read as zero.	R

31.2.11 UART Modem Status Register (UMSR)

The read-only UMSR provides the current state of the control lines from the modem to the processor. They are cleared when the processor reads UMSR.

	UMSR																															0x018
Bit																										7	6	5	4	3	2	1	0
																										Reserved			CTS	Reserved			CCTS
RST																										0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:5	Reserved	Writing has no effect, read as zero.	R
4	CTS	**Status of Clear To Send.** When MDCE bit is 1, this bit is the complement of CTS_ input. If Loop bit of UMCR is 1, this bit is equivalent to RTS bit of UMCR. 0: CTS_ pin is 1 1: CTS_ pin is 0	R
3:1	Reserved	Writing has no effect, read as zero.	R
0	CCTS	**Change status of CTS_.** When MDCE bit is 1, this bit indicates the state change on CTS_ pin. 0: No state change on CTS_ pin since last read of UMSR 1: A change occurs on the state of CTS_ pin	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.2.12 UART Scratchpad Register

This Scratchpad register is used as a scratch register for the programmer and has no effect on the UART.

	USPR																															0x01C
Bit																									7	6	5	4	3	2	1	0
																									Scratch Data							
RST																									?	?	?	?	?	?	?	?

31.2.13 Infrared Selection Register (ISR)

The ISR is used to configure the slow-infrared (SIR) interface that is provided in each UART to support two-way wireless communication using infrared transmission that conforms to the IrDA serial infrared specification 1.1. The maximum frequency is up to 115.2kbps.

	ISR																															0x020
Bit																									7	6	5	4	3	2	1	0
																									Reserved			RDPL	TDPL	XMODE	RCVEIR	XMITIR
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:5	Reserved	Writing has no effect, read as zero.	R
4	RDPL	**Receive Data Polarity.** 0: Slow-infrared (SIR) interface decoder takes positive pulses as zeros 1: SIR decoder takes negative pulses as zeros	W
3	TDPL	**Transmit Data Polarity.** 0: SIR encoder generates a positive pulse for a data bit of zero 1: SIR encoder generates a negative pulse for a data bit of zero	W
2	XMODE	**Transmit Pulse Width Mode.** Set when the transmit encoder needs to generate 1.6us pulses (that are 3/16 of a bit-time at 115.2 kbps). Cleared when the transmit encoder needs to generate 3/16 of a bit-time wide according to current baud rate. 0: Transmit pulse width is 3/16 of a bit-time wide 1: Transmit pulse width is 1.6 us	W
1	RCVEIR	**Receiver SIR Enable.** This bit is used to select the signal from the RXD pin is processed by the IrDA decoder before it is fed to the UART (RCVEIR = 1) or bypass IrDA	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		decoder and is fed directly to the UART (RCVEIR = 0). 0: Receiver is in UART mode 1: Receiver is in SIR mode	
0	XMITIR	**Transmitter SIR Enable.** This bit is used to select TXD output pin is processed by the IrDA encoder before it is fed to the device pin (XMITIR = 1) or bypass IrDA encoder and is fed directly to the device pin (XMITIR = 0). **NOTE:** disable infrared LED before XMITIR is set, otherwise a false start bit may occur. 0: Transmitter is in UART mode 1: Transmitter is in SIR mode	W

31.2.14 UART M Register (UMR)

	UMR																															0x024
Bit																									7	6	5	4	3	2	1	0
	Reserved																											M				
RST																									0	0	0	0	0	0	0	0

M is the value of UMR register.

It will take UART at least M cycles to transmit or receive one bit.

It will take UART at most M+1 cycles to transmit or receive one bit.

31.2.15 UART Add Cycle Register (UACR)

	UACR																															0x028
Bit																					11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																				AC											
RST																					0	0	0	0	0	0	0	0	0	0	0	0

If nth bit of the register is 1, it will take UART M+1 cycles to transmit or receive the bit of date.

If the register is 12'h0, UART will receive or transmit a bit by M cycle(s).

If the register is 12'hfff, UART will receive or transmit a bit by M+1 cycle(s).

For the detail to see For any frequency clock to use the Uart.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.2.16 UART RXFIFO Counter Register (URCR)

	URCR																															0x040
Bit																									7	6	5	4	3	2	1	0
																									Reserved	RCNT						
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7	Reserved	Writing has no effect, read as zero.	R
6:0	RCNT	RXFIFO Counter. Indicates there are n data in RXFIFO when this field is n.	R

31.2.17 UART TXFIFO Counter Register (UTCR)

	UTCR																															0x044
Bit																									7	6	5	4	3	2	1	0
																									Reserved	TCNT						
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7	Reserved	Writing has no effect, read as zero.	R
6:0	TCNT	TXFIFO Counter. Indicates there are n data in TXFIFO when this field is n.	R

31.3 Operation

The following sections describe the UART operations that include flow of configuration, data transmission, data reception, and Infrared mode.

31.3.1 UART Configuration

Before UART starts to transfer data or changing transfer format, configuration must be done to define the transfer format. The sample flow is as the following:

In FIFO mode, set FME bit of UFCR to 1, reset receive and transmit FIFO, then initialize the UART as

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

described below:

1 Clear UFCR.UME to 0.
2 Set value in UDLL/UDHR to generate the baud rate clock.
3 Set data format in ULCR.
4 If it is in FIFO MODE, set FME bit and other FIFO control in UFCR, reset receive and transmit FIFO, otherwise skip item 4.
5 Set each interrupt enable bit in UIER in interrupt-based transfer or set UFCR.DME in DMA-based transfer (DMA transfer is FIFO mode only), then set UFCR.UME.

31.3.2 Data Transmission

After configuration, UART is ready for data transfer. For data transmission, refer to the following procedure:

1 Read ULSR.TDRQ (interrupt disable) or wait for transmit data request interrupt (interrupt enable), if TDRQ = 1 or transmit data request interrupt generates, that means there is enough empty location in UTHR for new data.
2 If ULSR.TDRQ is 1 or get the transmit data request interrupt, write transmit data to UTHR to start transmission.
3 Do item 1 and item 2 if there are more data waiting for transmit.
4 After all necessary data are written to UTHR, wait ULSR.TEMP = 1, that means all data completely transmitted.
5 If it is necessary to send break, set ULCR.SBK and at least wait for 1-bit interval time to send a valid break, then clear ULCR.SBK.
6 Clear UME bit to finish UART transmission.

31.3.3 Data Reception

After configuration, UART is ready for data transfer. For data reception, refer to the following sample procedure:

1 Read ULSR.DRY (interrupt disable) or wait for receive data request interrupt (interrupt enable), if ULSR.DRY =1 or receive data request interrupt generates, that means URBR has one data (non-FIFO mode) or data in URBR reaches the trigger value. (FIFO mode)
2 If ULSR.DRY = 1 or receive data request interrupt generates, then read ULSR.FIFOE or see if there is error interrupt, if FIFOE = 1, it means received data has receive error, then go to error handler, other wise go to item 3.
3 Read one received data in URBR (non-FIFO mode) or data equal to trigger value in URBR. (FIFO mode)
4 Check whether all data received: check whether ULSR.DRY = 0, in FIFO mode and interrupt is enabled, timeout interrupt may generate, when timeout interrupt generates, read URBR till ULSR.DRY = 0.
5 Clear UFCR.UME to end data reception when all data are received and ULSR.DRY = 0.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.3.4 Receive Error Handling

A sample error handling flow is as the following:

1. If ULSR.FIFOE = 1, it means there is receive error in received data, then check what error it is.
2. If ULSR.OVER = 1, go to OVER error handling.
3. If ULSR.BI = 1, go to Break handling.
4. If ULSR.FMER = 1, go to Frame error handling.
5. If PARER = 1, go to PARER error handling.

31.3.5 Modem Transfer

When UMCR.MDCE = 1, modem control is enabled. Transfer flow can be stopped and restarted by software through RTS_ and CTS_ pin. When UART transmitter detects low level on CTS_ pin, it stops transmission and TxD pin goes to mark state after finishing transmitting the current character until it detects CTS_ pin goes back to high level. RTS_ pin is output to receiving UART and its state can be controlled by setting UMCR.RTS bit, that is, setting UMCR.RTS to 1, RTS_ pin is low level output that means UART is ready to receive data, on the contrary, it means UART currently can't receive more data.

31.3.6 DMA Transfer

UART can operate in DMA-based (UFCR.DME = 1, FIFO mode only), that is, dma request initiated by UART takes the place of interrupt request and transmission/reception is carried out using DMA instead of CPU. Be sure that software guarantee to disable transmit and receive interrupt except timeout and error interrupts.

During DMA transfer, if an interrupt occurs, software must first read the ULSR to see if an error interrupt exists, then check the UIIR for the source of the interrupt and if DMA channel is already halt because of the error indicator from UART, then disable DMA channel and read out all the error data from receive FIFO. Software re-set and re-enable DMA and data transfer by DMA will re-start.

31.3.7 Slow IrDA Asynchronous Interface

Each UART supports slow infra-red (SIR) transmission and reception by setting ISR.XMITIR and ISR.RCVEIR to 1 (make sure the two bits are not set to 1 at the same time because SIR can't operate full-duplex). According to the IrDA 1.1, data rate is limited at a maximum value of 115.2Kbps.

In SIR transmit mode, the transmit pulse comes out at a rate of 3/16 (when the transmit data bit is zero); in SIR receive mode, the receiver must detect the 3/16 pulsed period to recognize a zero value (an active high or low pulse is demodulation to 0, and no pulse is demodulation to 1).

Compared to normal UART, there are some limitations to SIR, that is, each character is fixed to 8-bit data width, no parity and 1 stop bit and modem function is ignored. The IrDA 1.1 specifies a minimum

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

10ms latency after an optical node ceases transmitting before its receiver recovers its receiving function and software must guarantee this delay.

In the IrDA 1.1 specification, communication must start up at the rate of 9600bps, but then allows the link to negotiate higher (or lower) data rates if supported by both ends. However, the communication rate will not automatically change. Change, if necessary, is performed by software.

31.3.8 For any frequency clock to use the UART

NOTE: if you don't set M register and UACR the UART work at normal mode with the specified frequencies. To use other frequency you should to set M register and UACR to right value.

1 The Improving

Following changes are made:

a One bit is composed by M CLK_{BR} cycles, which can be 4~1024.

b Some extra CLK_{BR} cycles can be inserted in some bits in one frame, so that like M has fraction.

For instance:

$CLK_{BR} = CLK_{DEV} / N$ N = 1, 2, …

$CLK_{BR} = CLK_{DEV} = 4MHz$

Band rate = 460800

In accurate

$M_a = 8.681$

We take

M = 8, with 8 extra cycles in every frame

A 12-bit register is used to indicate where to insert the extra cycles.

The first line is the time expected

The second line is the time actual

The third line is the time error

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

For transmission, in theory, the biggest error is half of CLK_{BR} cycle, which is 0.125us here.

2 To set UMR register

$CLK_{BR} = CLK_{DEV} / N$

$M_a = CLK_{BR}$/band rate

M is modem of M_a.

Write M to Mregister.

Considering the power and the robust quality, for M form 6 to 32 is you better select by set the UDLR.

The max error

$$\frac{0.5/CLK_{BR}}{M_a/CLK_{BR}} = 0.5/M_a < 0.5/M$$

M	4	8	16	32	64
error/W_{bit}	12.5%	6.25%	3.125%	1.56%	0.78%

3 To set UACR value

For each bit of it means:

0: means not to add additional cycle to the bit that UART is prepare to transmit or receive, in another word, you will to use M cycles to transmit or receive the bit

1: means to add additional cycle to the bit that UART is prepare to transmit or receive, in another word, you will to use M+1 cycles to transmit or receive the bit

To set UACR value you must ensure that the max error of each bit should be less than $0.5P_{BR}$.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

For example：M_a-M =0.15; M+1-M_a=0.85;

Write 8 to UMR,

Write 0x408 to UACR

cycle/bit	：	M，	M，	M，	M+1，	M，	M，	M，	M，	M，	M，	M+1，	M
UACR	：	0	0	0	1	0	0	0	0	0	0	1	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32 MMC/SD CE-ATA Controller

32.1 Overview

The Multi Media Card (MMC) is a universal low cost mass storage and communication media that is designed to cover a wide area of applications such as electronic toys, organizers, PDAs, smart phones, and so on.

The Secure Digital (SD) card is an evolution of MMC. It is specifically designed to meet the security, capacity, performance, and environmental requirements inherent in newly emerging audio and video consumer electronic devices. The physical form factor, pin assignment, and data transfer protocol are forward compatible with the Multi Media Card with some additions. An SD card can be categorized as SD memory or SD I/O card, commonly known as SDIO. A memory card invokes a copyright protection mechanism that complies with the security of the SDMI standard and is faster and capable of higher memory capacity. The SDIO card provides high-speed data I/O with low-power consumption for mobile electronic devices.

For CE-ATA detail protocol , please referred to WWW.CE-ATA.ORG.

32.2 Features

- Fully compatible with the MMC System Specification version 4.2
- Support SD Specification 3.0
- Support SD I/O Specification 1.0 with 1 command channel and 4 data channels
- Maximum data rate is 50MBps
- Support MMC data width 1bit ,4bit and 8bit
- Built-in programmable frequency divider for MMC/SD bus
- Built-in Special Descriptor DMA
- Multi-SD function support including multiple I/O and combined I/O and memory
- IRQ supported enable card to interrupt MMC/SD controller
- Single or multi block access to the card including erase operation
- Stream access to the MMC card
- Supports SDIO read wait, interrupt detection during 1-bit or 4-bit access
- Supports CE-ATA digital protocol commands
- Support Command Completion Signal and interrupt to CPU
- Command Completion Signal disable feature
- The maximum block length is 4096bytes

32.3 Pins Description

- MSC_CLK, output, host to card clock signal.
- MSC_CMD, inout, bidirectional command/response signal.
- MSC_DAT[7:0], inout, bidirectional data bus.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.4 Block Diagram

Figure 32-1 MMC/SD Controller Block Diagram

32.5 MMC/SD Controller Signal I/O Description

MSC and the card communication over the CMD and DATA line is base on command and data bit streams which are initiated by a start bit and terminated by a stop bit.

Command: a command is a token, which starts an operation. A command is sent from MSC either to a single card (addressed command) or to all connected cards (broadcast command). A command is transferred serially on the CMD line. Each command token is preceded by a start bit ('0') and succeeded by an end bit ('1'). The total length is 48 bits and protected by CRC bits.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 32-1 Command Token Format

Bit position	**47**	**46**	**[45 : 40]**	**[39 : 8]**	**[7 : 1]**	**0**
Width (bits)	1	1	6	32	7	1
Value	0	1	X	X	x	1
Description	Start bit	Transmission bit	Command index	argument	CRC7	End bit

Response: a response is a token which is sent from an addressed card, or (synchronously) from all connected cards, to MSC as an answer to a previously received command. A response is transferred serially on the CMD line. Response tokens have varies coding schemes depending on their content.

Data: data can be transferred from the card to MSC or vice versa. Data is transferred via the data line. Data transfers to/from the SD Memory Card are done in blocks. Data blocks always succeeded by CRC bits. Single and multiple block operations are defined. Note that the Multiple Block operation mode is better for faster write operation. A multiple block transmission is terminated when a stop command follows on the CMD line. Data transfer can be configured by the MSC to use single or multiple data lines.

Table 32-2 MMC/SD Data Token Format

Description	**Start bit**	**Data**	**CRC16**	**End bit**
Stream Data	0	X	no CRC	1
Block Data	0	X	X	1

32.6 Register Description

Following chapter will descript the functions of all software accessible registers.

Conventions:

1. Register Address = Base + Address offset
2. Register read/write attribute
   R - Read only
   W - Write only
   RW - read and write
   RCW - read and write, but clear to 0 by read
   RSW - read and write, but set to 1 by read
   RWC - read and write, clear to 0 by write 1, write 0 has no effect
   RWS - read and write, set to 1 by write 1, write 0 has no effect
   RC - read only, and clear to 0 by read
   RS - read only, and set to 1 by read

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SPEC - special access method, relate to its description

3. Reset Value

1 - reset to 1

0 - reset to 0

? - value unknown after reset

The MMC-SD-CE_ATA controller is controlled by a set of registers that the application configures before every operation. The following table lists all the MSC registers.

Table 32-3 Registers Memory Map Base Address

Name	Base	Description
MSC0	0x13450000	Address base of MSC0
MSC1	0x13460000	Address base of MSC1

Table 32-4 MMC/SD Controller Registers Map

Name	RW	Reset Value	Address Offset	Access Size
MSC_CTRL	W	0x0000	0x0000	16
MSC_STAT	R	0x??000040	0x0004	32
MSC_CLKRT	RW	0x0000	0x0008	16
MSC_CMDAT	RW	0x00005000	0x000C	32
MSC_RESTO	RW	0x0100	0x0010	16
MSC_RDTO	RW	0x00FFFFFF	0x0014	32
MSC_BLKLEN	RW	0x0000	0x0018	16
MSC_NOB	RW	0x0000	0x001C	16
MSC_SNOB	R	0x????	0x0020	16
MSC_IMASK	RW	0xFFFFFFFF	0x0024	32
MSC_IFLG	RW	0x2000	0x0028	32
MSC_CMD	RW	0x00	0x002C	8
MSC_ARG	RW	0x00000000	0x0030	32
MSC_RES	R	0x????	0x0034	16
MSC_RXFIFO	R	0x????????	0x0038	32
MSC_TXFIFO	W	0x????????	0x003C	32
MSC_LPM	RW	0x00000000	0x0040	32
MSC_DMAC	RW	0x00000000	0x0044	32
MSC_DMANDA	RW	0x00000000	0x0048	32
MSC_DMADA	R	0x00000000	0x004C	32
MSC_DMALEN	R	0x00000000	0x0050	32
MSC_DMACMD	R	0x00000000	0x0054	32
MSC_CTRL2	RW	0x00800000	0x0058	32
MSC_RTCNT	R	0x00000000	0x005C	32

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.6.1 MSC Control Register (MSC_CTRL)

	MSC_CTRL																													BASE + 0x0000	
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	SEND_CCSD	SEND_AD_CCSD	Reserved						EXIT_MULTIPLE	EXIT_TRANSFER	START_READ WAIT	STOP_READ WAIT	RESET	START_OP	CLOCK_CTRL	
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	SEND_CCSD	0: clear bit 1: Send Command Completion Signal Disable (CCSD) to CE_ATA device when set, host sends CCSD to CE_ATA device. Software set the bit only if current command is expecting CCS and interrupts are enabled in CE_ATA devices. Once the CCSD pattern is sent to device, host automatically clears the SEND_CCSD bit.	W
14	SEND_AS_CCSD	0: clear bit 1: send internally generated stop after sending CCSD to CE_ATA device When set, host automatically sends internally-generated STOP command(CMD12) to CE_ATA device. After sending CMD12, MSC_STAT. AUTO_CMD12_DONE is set and generates interrupt to CPU. After sending the CCSD, controller automatically clears the SEND_AS_CCSD bit.	W
13:8	Reserved	Writing has no effect.	R
7	EXIT_MULTIPLE	If CMD12 or CMD52 (I/O abort) is to be sent to terminate multiple block read/write in advance, set this bit to 1. 0: No effect 1: Exit from multiple block read/write	W
6	EXIT_TRANSFER	**Only used for SDIO suspend/resume and MMC stream read.** For SDIO, after suspend is accepted, set this bit with 1. For MMC, after the data of the expected number are received, set this bit with 1. 0: No effect 1: Exit from multiple block read/write after suspend is accepted, or exit from stream read	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5	START_READ WAIT	Only used for SDIO Read Wait. Start the Read Wait cycle. 0: No effect 1: Start Read Wait	W
4	STOP_READ WAIT	Only used for SDIO Read Wait. Stop the Read Wait cycle. 0: No effect 1: Start Read Wait	W
3	RESET	Reset the MSC controller. 0: No effect 1: Reset the MSC controller	W
2	START_OP	This bit is used to start the new operation. When starting the clock, this bit can be 1. When stopping the clock, this bit can only be 0. 0: Do nothing 1: Start the new operation	W
1:0	CLOCK_CTRL	These bits are used to start or stop clock. 00: Do nothing 01: Stop MMC/SD clock 10: Start MMC/SD clock 11: Reserved	W

32.6.2 MSC Control 2 Register (MSC_CTRL2)

	MSC_CTRL2																															BASE + 0x0058
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved			PIP					Reserved_1	Reserved																		STPRM	Reserved_0	SMS		
RST	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:29	Reserved	Writing has no effect, read as zero.	R
28:24	PIP	Pin Level Interrupt Polarity 1: Interrupt can be trigged when pin level is high 0: Interrupt can be trigged when pin level is low	RW
23	Reserved_1	Inner usage, keep to 1.	RW
22:5	Reserved	Writing has no effect, read as zero.	R
4	STPRM	Stop Read Operation Mode Selection 0: Host can stop read operation during data transfer 1: Host only can stop read operation during block gap	RW
3	Reserved_0	Inner usage, keep to 0.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2:0	SMS	Speed Mode Selection 000: Default speed 001: High speed 010: SDR12 011: SDR25 100: SDR50 Others: reserved	RW

32.6.3 MSC Status Register (MSC_STAT)

	MSC_STAT																															BASE + 0x0004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AUTO_CMD12_DONE	Reserved		PIN_LEVEL					Reserved			BCE	BDE	BAE	BAR	DMAEND	IS_RESETTING	SDIO_INT_ACTIVE	PRG_DONE	DATA_TRAN_DONE	END_CMD_RES	DATA_FIFO_AFULL	IS_READ WAIT	CLK_EN	DATA_FIFO_FULL	DATA_FIFO_EMPTY	CRC_RES_ERR	CRC_READ_ERROR	CRC_WRITE_ERROR		TIME_OUT_RES	TIME_OUTREAD
RST	0	0	0	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bits	Name	Description	RW
31	AUTO_CMD12_DONE	Indicates that the stop command (CMD12) that is internally generated by controller has finished.	R
30:29	Reserved	Read as zero.	R
28:24	PIN_LEVEL	MSC Interface Pin Level PIN_LEV[4:0] indicates the pin level of MSC_CMD, MSC_DAT[3:0] separately.	R
23:21	Reserved	Read as zero.	R
20	BCE	Boot CRC error. 0: No boot CRC error occurs. 1: Boot CRC is not correct.	R
19	BDE	Boot data end. 0: No boot data or boot data is not finished. 1: Boot data is received completely.	R
18	BAE	Boot acknowledge is error. 0: No boot acknowledge is received or boot acknowledge is correct (decided by BAR) 1: The received boot acknowledge is not correct.	R
17	BAR	Boot acknowledge received. 0: No boot acknowledge received.	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Boot acknowledge received.	
16	DMAEND	Indicates that the DMA has finished the current transfer.	R
15	IS_RESETTING	MSC is resetting after power up or MSC_CTRL[RESET] is written with 1. 0: Reset has been finished 1: Reset has not been finished	R
14	SDIO_INT_ACTIVE	Indicates whether an interrupt is detected at the SD I/O card. A separate acknowledge command to the card is required to clear this interrupt. 0: No interrupt detected 1: The interrupt from SDIO is detected	R
13	PRG_DONE	Indicates whether card has finished programming. 0: not finished 1: finished	R
12	DATA_TRAN_DONE	Indicates whether data transmission to card has completed. 0: not completed 1: completed	R
11	END_CMD_RES	Indicates whether command and response/no-response sequence have been completed 0: not completed 1: completed	R
10	DATA_FIFO_AFULL	Indicates whether the data FIFO is almost full. 0: The number of words in FIFO is less then 127. 1: The number of words in FIFO is equal to or great then 127	R
9	IS_READ WAIT	Indicates whether SDIO card has entered Read Wait State. 0: Card has not entered Read Wait 1: Card has entered Read Wait	R
8	CLK_EN	Clock enabled. 0: Clock is off 1: Clock is on	R
7	DATA_FIFO_FULL	Indicates whether the data FIFO is full. 0: Data FIFO is not full 1: Data FIFO is full	R
6	DATA_FIFO_EMPTY	Indicates whether data FIFO is empty. 0: Data FIFO is not empty 1: Data FIFO is empty	R
5	CRC_RES_ERR	Response CRC error. 0: No error on the response CRC 1: CRC error occurred on the response	R

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

4	CRC_READ_ERROR	Read CRC error. 0: No error on received data 1: CRC error occurred on received data	R
3:2	CRC_WRITE_ERROR	Write CRC error. 00: No error on transmission of data 01: Card observed erroneous transmission of data 10: No CRC status is sent back 11: Reserved	R
1	TIME_OUT_RES	Response Time Out. 0: Card response has not timed out 1: Card response has time out	R
0	TIME_OUT_READ	Read time out. 0: Card read data has not timed out 1: Card read data has timed out	R

32.6.4 MSC Clock Rate Register (MSC_CLKRT)

The MSC_CLKRT register specifies the frequency division of the MMC/SD bus clock. The software is responsible for setting this register.

	MSC_CLKRT																BASE + 0x0008															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																													CLK_RATE		
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:3	Reserved	Writing has no effect, read as zero.	R
2:0	CLK_RATE	Clock rate. 000: DEV_CLK 001: 1/2 of DEV_CLK 010: 1/4 of DEV_CLK 011: 1/8 of DEV_CLK 100: 1/16 of DEV_CLK 101: 1/32 of DEV_CLK 110: 1/64 of DEV_CLK 111: 1/128 of DEV_CLK This field must be set to 0 when the controller works during normal writing or reading.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.6.5 MSC Command and Data Control Register (MSC_CMDAT)

	MSC_CMDAT																															BASE + 0x000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCS_EXPECTED	READ_CEATA	Reserved		DIS_BOOT	Reserved_0	EXP_BOOT_ACK	BOOT_MODE	Reserved					Reserved_0	SDIO_PDRT	AUTO_CMD12	RTRG		TTRG		IO_ABORT	BUS_WIDTH		Reserved	INIT	BUSY	STREAM_BLOCK	WRITE_READ	DATA_EN	RESPONSE_FORMAT		
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	CCS_EXPECTED	0: interrupts are not enabled in CE-ATA device, or commands does not expect CCS from device 1: interrupts are enabled in CE_ATA device, or RW_BLK command expects command completion signal from device If the command expects Command Completion Signal (CCS) from the device, the software should set the control bit. It is auto cleared 0 by hardware.	RW
30	READ_CEATA	0: host is not performing read access (RW_BLK or RW_REG) towards CE_ATA device 1: host id performing read access (RW_BLK or RW_REG) towards CE_ATA device Software should set the bit to indicate that CE_ATA device is being accessed for read transfer. The bit is used to disable read data timeout indication while performing CE_ATA read transfers. It is auto cleared 0 by hardware.	RW
29:28	Reserved	Writing has no effect, read as zero.	R
27	DIS_BOOT	0: Do nothing 1: Stop boot operation DIS_BOOT should not be used with ENA_BOOT at the same time.	RW
26	Reserved_0	Inner usage, keep to 0.	RW
25	EXP_BOOT_ACK	Whether boot acknowledge pattern is expected or not 0: Boot acknowledge pattern is not expected 1: Boot acknowledge pattern is expected	RW
24	BOOT_MODE	Boot mode operation selection. 0: Ignore.	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Alternative boot operation	
23:19	Reserved	Writing has no effect, read as zero.	R
18	Reserved_0	Inner usage, keep to 0.	RW
17	SDIO_PRDT	Determine whether SDIO interrupt is 2 cycle or extend more cycle when data block last is transferred. 0: more cycle (like single block) 1: exact 2 cycle	RW
16	AUTO_CMD12	This field controls use of auto command functions. 0: Auto CMD12 is disabled 1: MSC controller will issue CMD12 automatically after the last data block transfer is finished. when stop command has finished, it is auto cleared 0 by hardware.	RW
15:14	RTRG	Receive FIFO Trigger Value Select. These bits set the receive FIFO half-empty threshold value, when the number of transmit FIFO >= threshold value , RXFIFO_RD_REQ will be set to 1. 00: more than or equal to 16 01: more than or equal to 32 10: more than or equal to 64 11: more than or equal to 96	RW
13:12	TTRG	Transmit FIFO Trigger Value Select. These bits set the transmit FIFO half-empty threshold value, when the number of transmit FIFO < threshold value , TXFIFO_WR_REQ will be set to 1. 00: less than 16 01: less than 32 10: less than 64 11: less than 96	RW
11	IO_ABORT	Specifies the current command is used to abort data transfer. 0: Nothing 1: The current command is used to abort transfer It is auto cleared 0 by hardware.	WR
10:9	BUS_WIDTH	Specifies the width of the data bus. 00: 1-bit 01: Reserved 10: 4-bit 11: 8bit	WR
8	Reserved	Writing has no effect. Read as zero.	R
7	INIT	80 initialization clocks. 0: Issues command directly	W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Issues 80 clocks before command	
6	BUSY	Specifies whether a busy signal is expected after the current command. This bit is for no data command/response transactions only. 0: Not expect a busy signal 1: Expects a busy signal. If the response is R1b, then set it	WR
5	STREAM_BLOCK	Stream mode. 0: Data transfer of the current command sequence is not in stream mode 1: Data transfer of the current command sequence is in stream mode	WR
4	WRITE_READ	Data Transfer Direction Selection. 0: Read (from card to host) 1: Write (from host to card)	RW
3	DATA_EN	Specifies whether the current command includes data transfer or not. It is also used to reset RX_FIFO and TX_FIFO. 0: Current command without data transfer 1: Current command with data transfer	RW
2:0	RESPONSE_FORMAT	Response Type Selection. 000: No response 001: Format R1 and R1b 010: Format R2 011: Format R3 100: Format R4 101: Format R5 110: Format R6 111: Format R7	RW

32.6.6 MSC Response Time Out Register (MSC_RESTO)

	MSC_RESTO																BASE + 0x0010															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	RES_TO															
RST																	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
15:0	RES_TO	Specifies the maximum number of MSC_CLK clock cycles between the end bit of the command and the response from the SD card. The default value is 128.	RW

32.6.7 MSC Read Time Out Register (MSC_RDTO)

	MSC_RDTO																															BASE + 0x0014
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	READ_TO																															
RST	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31:0	READ_TO	Specifies the maximum number of clocks between the command and when the MMC/SD host controller turns on the time-out error for the received data. The unit is MSC_CLK.	RW

32.6.8 MSC Block Size Register (MSC_BLKLEN)

	MSC_BLKLEN																															BASE + 0x0018
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	BLK_LEN															
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:0	BLK_LEN	Specifies the number of bytes in a block, and is normally set to 0x200 for MMC/SD data transactions. The value Specified in the cards CSD.	RW

32.6.9 MSC Block Count Register (MSC_NOB)

	MSC_NOB																															BASE + 0x001C
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	NOB															
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
15:0	NOB	Specifies the number of blocks in a data transfer. One block is a possibility.	RW

32.6.10 MSC Successfully-transferred Blocks Count Register (MSC_SNOB)

In block mode, the MSC_SNOB register records the number of successfully transferred blocks. If the last block has CRC error, this register also summaries it. It is used to query blocks for multiple block transfer.

	MSC_SNOB																BASE + 0x0020															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	MSC_SNOB															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
15:0	MSC_SNOB	Specify the number of successfully transferred blocks for a multiple block transfer.	R

32.6.11 MSC Interrupt Mask Register (MSC_IMASK)

	MSC_IMASK																BASE + 0x0024															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_DATA_DONE	Reserved_1		PIN_LEVEL					WR_ALL_DONE	Reserved		BCE	BDE	BAE	BAR	DMAEND	AUTO_CMD12_DONE	DATA_FIFO_FULL	DATA_FIFO_EMP	CRC_RES_ERR	CRC_READ_ERR	CRC_WRITE_ERR	TIME_OUT_RES	TIME_OUT_READ	SDIO	TXFIFO_WR_REQ	RXFIFO_RD_REQ	Reserved		END_CMD_RES	PRG_DONE	DATA_TRAN_DONE
RST	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bits	Name	Description	RW
31	DMA_DATA_DONE	Mask the interrupt DMA_DATA_DONE 0: Not masked 1: Masked	RW
30:29	Reserved_1	Inner usage, keep to 1.	RW
28:24	PIN_LEVEL	Mask the interrupt of PIN_LEVEL separately 0: Not masked 1: Masked	RW
23	WR_ALL_DONE	Mask the interrupt WR_ALL_DONE 0: Not masked	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Masked	
22:21	Reserved	Writing has no effect, read as zero.	R
20	BCE	Boot CRC error. 0: Not masked 1: Masked	RW
19	BDE	Boot data end. 0: Not masked 1: Masked	RW
18	BAE	Mask the interrupt of BAE. 0: Not masked. 1: Masked.	RW
17	BAR	Mask the interrupt of BAR. 0: Not masked. 1: Masked.	RW
16	DMAEND	Mask the interrupt of DMA end. 0: Not masked. 1: Masked.	RW
15	AUTO_CMD12_DONE	Mask the interrupt AUTO_CMD12_DONE. 0: Not masked 1: Masked	RW
14	DATA_FIFO_FULL	0: Not masked 1: Masked	RW
13	DATA_FIFO_EMP	0: Not masked 1: Masked	RW
12	CRC_RES_ERR	0: Not masked 1: Masked	RW
11	CRC_READ_ERR	0: Not masked 1: Masked	RW
10	CRC_WRITE_ERR	0: Not masked 1: Masked	RW
9	TIME_OUT_RES	0: Not masked 1: Masked	RW
8	TIME_OUT_READ	0: Not masked 1: Masked	RW
7	SDIO	Mask the interrupt from the SD I/O card. 0: Not masked 1: Masked	WR
6	TXFIFO_WR_REQ	Mask the Transmit FIFO write request interrupt. 0: Not masked 1: Masked	WR
5	RXFIFO_RD_REQ	Mask the Receive FIFO read request interrupt. 0: Not masked	WR

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Masked	
4:3	Reserved	Writing has no effect, read as zero.	R
2	END_CMD_RES	Mask the End command response interrupt. 0: Not masked 1: Masked	WR
1	PRG_DONE	Mask the Programming done interrupt. 0: Not masked 1: Masked	WR
0	DATA_TRAN_DONE	Mask the Data transfer done interrupt. 0: Not masked 1: Masked	WR

32.6.12 MSC Interrupt Flag Register (MSC_IFLG)

The MSC_IFLG register shows the currently requested interrupt. The FIFO request interrupts, TXFIFO_WR_REQ, and RXFIFO_RD_REQ are masked off with the DMA function is used. The software is responsible for monitoring these bit in program I/O mode.

	MSC_IFLG																															BASE + 0x0028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_DATA_DONE	Reserved_0		PIN_LEVEL					WR_ALL_DONE	Reserved		BCE	BDE	BAE	BAR	DMAEND	AUTO_CMD12_DONE	DATA_FIFO_FULL	DATA_FIFO_EMP	CRC_RES_ERR	CRC_READ_ERR	CRC_WRITE_ERR	TIME_OUT_RES	TIME_OUT_READ	SDIO	TXFIFO_WR_REQ	RXFIFO_RD_REQ	Reserved		END_CMD_RES	PRG_DONE	DATA_TRAN_DONE
RST	0	0	0	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	DMA_DATA_DONE	This field will be set to 1 when DATA_TRAN_DONE and DMAEND are valid. 0: the interrupt is not detected 1: the interrupt is detected	RW
30:29	Reserved_0	Inner usage, keep to 0.	RW
28:24	PIN_LEVEL	Indicates PIN_LEVEL interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
23	WR_ALL_DONE	This field will be set to 1 when DATA_TRAN_DONE, DMAEND and PRG_DONE are valid. 0: the interrupt is not detected	RW

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: the interrupt is detected	
22:21	Reserved_0	Inner usage, keep to 0	R
20	BCE	Boot CRC error. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
19	BDE	Boot data end. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
18	BAE	Indicates the BAE interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
17	BAR	Indicates the BAR interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
16	DMAEND	Indicates the DMA end interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
15	AUTO_CMD12_DONE	Indicates AUTO_CMD12_DONE interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RWC
14	DATA_FIFO_FULL	Indicate data FIFO is full interrupt. 0: the interrupt is not detected 1: the interrupt is detected	R
13	DATA_FIFO_EMP	Indicate data FIFO is empty interrupt. 0: the interrupt is not detected 1: the interrupt is detected	R
12	CRC_RES_ERR	Indicate response CRC error interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear	RWC
11	CRC_READ_ERR	Indicate CRC read error interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear	RWC
10	CRC_WRITE_ERR	Indicate CRC write error interrupt. 0: the interrupt is not detected	RWC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: the interrupt is detected Write 1 to clear	
9	TIME_OUT_RES	Indicate response time out interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear	RWC
8	TIME_OUT_READ	Indicate read time out interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear	RWC
7	SDIO	Indicates whether the interrupt from SDIO is detected. 0: The interrupt from SDIO is not detected 1: The interrupt from SDIO is detected	R
6	TXFIFO_WR_REQ	Transmit FIFO write request. Set if data FIFO becomes half empty. (the number of words is < 8) 0: No Request for data Write to MSC_TXFIFO 1: Request for data write to MSC_TXFIFO	R
5	RXFIFO_RD_REQ	Receive FIFO read request. Set if data FIFO becomes half full (the number of words is >= 8) or the entries in data FIFO are the last read data. 0: No Request for data read from MSC_RXFIFO 1: Request for data read from MSC_RXFIFO	R
4:3	Reserved	Writing has no effect, read as zero.	R
2	END_CMD_RES	Indicates whether the command/response sequence has been finished. 0: The command/response sequence has not been finished 1: The command/response sequence has been finished Write 1 to clear.	RWC
1	PRG_DONE	Indicates whether card has finished programming. 0: Card has not finished programming and is busy 1: Card has finished programming and is no longer busy Write 1 to clear.	RWC
0	DATA_TRAN_DONE	Indicates whether data transfer is done. Note that for stream read/write, only when CMD12 (STOP_TRANS) has been sent, is this bit set. 0: Data transfer is not complete 1: Data transfer has completed or an error has occurred Write 1 to clear.	RWC

32.6.13 MSC Command Index Register (MSC_CMD)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MSC_CMD																															BASE + 0x002C
Bit																									7	6	5	4	3	2	1	0
																									Reserved		CMD_INDEX					
RST																									0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:6	Reserved	Writing has no effect, read as zero.	R
5:0	CMD_INDEX	Specifies the command index to be executed.	RW

32.6.14 MSC Command Argument Register (MSC_ARG)

	MSC_ARG																															BASE + 0x0030
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARG																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	ARG	Specifies the argument for the current command.	RW

32.6.15 MSC Response FIFO Register (MSC_RES)

The read-only MMC/SD Response FIFO register (RES_FIFO) holds the response sent back to the MMC/SD controller after every command. The size of this FIFO is 8 x 16-bit. The RES_FIFO does not contain the 7-bit CRC for the response. The Status for CRC checking and response time-out status is in the status register MSC_STAT.

The first half-word read from the response FIFO is the most significant half-word of the received response.

	MSC_RES																															BASE + 0x0034
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	DATA															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
15:0	DATA	Contains the response to the command that is sent by the MMC/SD controller.	R

32.6.16 MSC Receive FIFO Port Register (MSC_RXFIFO)

The MSC_RXFIFO is used to read the data from a card. It is read-only to the software, and is read on 32-bit boundary. The size of this FIFO is 128 x 32-bit.

	MSC_RXFIFO																															BASE + 0x0038
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	DATA	One word of read data.	R

32.6.17 MSC Transmit FIFO Port Register (MSC_TXFIFO)

The MSC_TXFIFO is used to write the data to a card. It is write-only to the software, and is written on 32-bit boundary. The size of this FIFO is 128 x 32-bit.

	MSC_TXFIFO																															BASE + 0x003C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	DATA	One word of write data.	W

32.6.18 MSC Low Power Mode Register (MSC_LPM)

The MSC_LPM is used to control whether MSC controller enters Low-Power Mode.

	MSC_LPM																															BASE + 0x0040
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRV_SEL		SMP_SEL	Reserved																												LPM
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:30	DRV_SEL	Drive clock selection 0: CMD and data are driven by MSC_CLK falling edge 1: CMD and data are driven by clock rising edge which is 1ns delayed by MSC_CLK 2: CMD and data are driven by clock rising edge which is 1/4 phase delayed by MSC_CLK 3: Reserved When this field is not zero, MSC_CLKRT.CLK_RATE must be set to 0.	RW
29	SMP_SEL	Sample clock selection 0: CMD and data are sampled by MSC_CLK rising edge 1: CMD and data are sampled by clock rising edge which is 1/4 or 1/2 phase delayed by MSC_CLK. When this field is 1, MSC_CLKRT.CLK_RATE must be set to 0.	
28:1	Reserved	Writing has no effect, read as zero.	R
0	LPM	Low Power Mode Enable 0: Disable low power mode 1: Enable low power mode Clock will stop when card in idle (should be normally set to only MMC and SD cards. For SDIO cards, if interrupts must be detected, clock should not be stopped) When software sets the bit, MSC clock can auto be stopped. **NOTE:** when set the bit, the clock start and stop can be not use.	RW

32.6.19 MSC DMA Control Register (MSC_DMAC)

The MSC_DMAC register is used to control the DMA transfer.

	MSC_DMAC																															BASE + 0x0044
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																								MODE_SEL	AOFST		ALIGNEN	INCR		DMASEL	DMAEN
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:8	Reserved	Writing has no effect. Read as zero.	R
7	MODE_SEL	Transfer mode selection. 0: Do not specify the transfer length. This is the standard transfer.	

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Specify the transfer length. If this mode is selected, the transfer performance can be improved.	
6:5	AOFST	Address Offset. This field is effective only when ALIGNEN is 1b. 00b:The lowest 2 bit of the data transfer start address is 00b 01b: The lowest 2 bit of the data transfer start address is 01b 10b: The lowest 2 bit of the data transfer start address is 10b 11b: The lowest 2 bit of the data transfer start address is 11b	RW
4	ALIGNEN	Align Enable 0: Only word boundary data transfer is supported 1: Byte boundary data transfer is supported	RW
3:2	INCR	Burst type selection. The field only effects to special DMA in MSC controller. 2'b00: The burst type of DMA is INCR16. 2'b01: The burst type of DMA is INCR32. 2'b10: The burst type of DMA is INCR64.	RW
1	DMASEL	One of supported DMA modes is selected. 0: Special DMA in MSC controller is used 1: Common DMA in DMAC controller is used	RW
0	DMAEN	DMA Function Enable 0: Disable DMA function 1:Enable DMA function	RW

32.6.20 MSC DMA Descriptor Address Register (MSC_DMANDA)

	MSC_DMANDA																															BASE + 0x0048
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NDA																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	NDA	SDMA Next Descriptor Physical Address The SDMA will change the register when a descriptor is read in. It should be 4-word aligned.	RW

32.6.21 MSC DMA Data Address Register (MSC_DMADA)

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MSC_DMADA																																BASE + 0x004C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	DA	SDMA Data Physical Address The data address of the current descriptor will be copied to this field. The SDMA will increment it during the data transfer automatically.	R

32.6.22 MSC DMA Data Length Register (MSC_DMALEN)

	MSC_DMALEN																																BASE + 0x0050
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LEN																																
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	LEN	SDMA Data Length to be Transferred Its unit is byte. The data length of the current descriptor will be copied to this field. The SDMA will decrement it during the data transfer automatically.	R

32.6.23 MSC DMA Command Register (MSC_DMACMD)

	MSC_DMACMD																																BASE + 0x0054
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IDI								ID								Reserved					OFFSET		ALIGN_EN							ENDI	LINK
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:24	IDI	Identification of the Interrupt DMA. This field will copy the ID when a DMA transfer interrupt occurs. If the interrupt of DMA is disabled, the field will not change.	R
23:16	ID	Identification of Current DMA Transfer.	R
15:11	Reserved	It has no use in the current version.	R
10:9	OFFSET	Address Offset. This field is effective when ALIGN_EN is set to 1b. 00: data address[1:0] is 00b 01: data address[1:0] is 01b 10: data address[1:0] is 10b 11: data address[1:0] is 11b	R
8	ALIGN_EN	Address Align Enable. This bit should be set to 1b if the data address is not word-aligned.	R
7:2	Reserved	It has no use in the current version.	R
1	ENDI	Interrupt Enable for Current DMA Transfer End 0: Disable interrupt 1: Enable interrupt	R
0	LINK	Control the end of DMA Descriptor. 0: DMA will go to idle state after the current transmission is finished. The MSC_STAT.DMAEND will be set to 1b when the current read/write is finished. If the interrupt mask bit MSC_IMASK.DMAEND is 0b, the interrupt flag MSC_IFLG. DMAEND will be set to 1b also. 1: DMA will continue to fetch another descriptor.	R

32.6.24 MSC RTFIFO Data Counter Register(MSC_RTCNT)

	MSC_RTCNT																																BASE + 0x005C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTCNT																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:0	RTCNT	This field indicates how many data in word units are stored in RTFIFO.	R

32.7 MMC/SD Functional Description

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

All communication between system and cards is controlled by the MSC. The MSC sends commands of two type: broadcast and addressed (point-to-point) commands.

Broadcast commands are intended for all cards, command like “Go_Idle_State”, “Send_Op_Cond”, “All_Send_CID” and “Set_Relative_Addr” are using way of broadcasting. During Broadcast mode, all cards are in open-drain mode, to avoid bus contention.

After Broadcast commands “Set_Relative_Addr” issue, cards are enter standby mode, and Addressed command will be used from now on, in this mode, CMD/DAT will return to push-pull mode, to have maximum driving for maximum operation frequency.

The MMC and the SD are similar product. Besides the 4x bandwidth and the built-in encryption, they are being programmed similarly.

The MMC/SD controller (MSC) is the interface between the software and the MMC/SD bus. It is responsible for the timing and protocol between the software and the MMC/SD bus. It consists of control and status registers, a 16-bit response FIFO that is 8 entries deep, and one 32-bit receive/transmit data FIFOs that are 16 entries deep. The registers and FIFOs are accessible by the software.

MSC also enable minimal data latency by buffering data and generating and checking CRCs.

32.7.1 MSC Reset

The MMC/SD controller (MSC) can be reset by a hardware reset or software reset. All registers and FIFO controls are set to their default values after any reset.

32.7.2 Voltage Validation

All cards shall be able to establish communication with the host using any operation voltage in the maximal allowed voltage range specified in this standard. However, the support minimum and maximum values for Vdd are defined in Operation Conditions register (OCR) and many not cover the whole range. Cards that store the CID and CSD data in the payload memory would be able to communicate these information only under data transfer Vdd conditions. That means if host and card have non compatible Vdd ranges, the card will not be able to complete the identification cycle, nor to send CSD data.

Therefore, a special command Send_Op_Cont (CMD1 for MMC), SD_Send_Op_Cont (CMD41 for SD Memory) and IO_Send_Op_Cont (CMD5 for SDIO) are designed to provide a mechanism to identify and reject cards which do not match the Vdd range desired by the host. This is accomplished by the host sending the required Vdd voltage window as the operand of this command. Cards which can not perform data transfer in the specified range must discard themselves from further bus operations and go into Inactive State. By omitting the voltage range in the command, the host can query each card

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

and determine the common voltage range before sending out-of-range cards into the Inactive State. This query should be used if the host is able to select a common voltage range or if a notification to the application of non usable cards in the stack is desired.

32.7.3 Card Registry

Card registry on MCC and SD card are different.

For SD card, Identification process start at clock rate Fod, while CMD line output drives are push-pull drivers instead of open-drain. After the bus is activated the host will request the cards to send their valid operation conditions. The response to ACMD41 is the operation condition register of the card. The same command shall be send to all of the new cards in the system. Incompatible cards are sent into Inactive State. The host then issue the command All_Send_CID (CMD2) to each card and get its unique card identification (CID) number. Card that is unidentified, that is, which is in Ready State, send its CID number as the response. After the CID was sent by the card it goes into Identification State. Thereafter, the host issues Send_Relative_Addr (CMD3) asks the card to publish a new relative card address (RCA), which is shorter that CID and which will be used to address the card in the future data transfer mode. Once the RCA is received the card state changes to the Stand-by State. At this point, if the host wants that the card will have another RCA number, it may ask the card to publish a new number by sending another Send_Relative_Addr command to the card. The last published RCA is the actual RCA of the card. The host repeats the identification process, that is, the cycles with CMD2 and CMD3 for each card in the system.

In MMC, the host starts the card identification process in open-drain mode with the identification clock rate Fod. The open drain driver stages on the CMD line allow parallel card operation during card identification. After the bus is active the host will request the cards to send their valid operation conditions (CMD1). The response to CMD1 is the ‘wired or’ operation on the condition restrictions of all cards in the system. Incompatible cards are sent into Inactive State. The host then issues the broadcast command All_Send_CID (CMD2), asking all cards for their unique card identification (CID) number. All unidentified cards, that is, those which are in Ready State, simultaneously start sending their CID numbers serially, while bit-wise monitoring their outgoing bit-stream. Those cards, whose outgoing CID bits do not match the corresponding bits on the command line in any one of the bit periods stop sending their CID immediately and must wait for the next identification cycle. Since CID is unique for each card, only one card can be successfully send its full CID to the host. This card then goes into Identification State. Thereafter, the host issues Set_Relative_Addr (CMD3) to assign to this card a relative card address (RCA). Once the RCA is received the card state changes to the Stand-by State, and the card does not react to further identification cycles, and its output switches from open-drain to push-pull. The host repeat the process, which is CM2 and CMD3, until the host receive time-out condition to recognize completion of the identification process.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.7.4 Card Access

32.7.4.1 Block Access, Block Write and Block Read

During block write (CMD24-27) one or more blocks of data are transferred from the host to the card with a CRC appended to the end of each block by the host. A card supporting block write shall always be able to accept a block of data defined by WRITE_BL_LEN. If the CRC fails, the card shall indicate the failure on the DAT line; the transferred data will be discarded and not written, and all further transmitted blocks (in multiple block write mode) will be ignored.

Programming of the CID and CSD registers does not require a previous block length setting. The transferred data is also CRC protected. If a part of the CSD or CID register is stored in ROM, then this unchangeable part must match the corresponding part of the receive buffer. If this match fails, then the card will report an error and not change any register contents. Some cards may require long and unpredictable times to write a block of data. After receiving a block of data and completing the CRC check, the card will begin writing and hold the DAT line low if its write buffer is full and unable to accept new data from a new WRITE_BLOCK command. The host may poll the status of the card with a SEND_STATUS command (CMD13) at any time, and the card will respond with its status. The status bit READY_FOR_DATA indicates whether the card can accept new data or whether the write process is still in progress). The host may deselect the card by issuing CMD7 (to select a different card) which will displace the card into the Disconnect State and release the DAT line without interrupting the write operation. When reselecting the card, it will reactivate busy indication by pulling DAT to low if programming is still in progress and the write buffer is unavailable.

Block read is similar to stream read, except the basic unit of data transfer is a block whose maximizes is defined in the CSD (READ_BL_LEN). If READ_BL_PARTIAL is set, smaller blocks whose starting and ending address are entirely contained within one physical block (as defined by READ_BL_LEN) may also be transmitted. Unlike stream read, a CRC is appended to the end of each block ensuring data transfer integrity. CMD17 (READ_SINGLE_BLOCK) initiates a block read and after completing the transfer, the card returns to the Transfer state. CMD18 (READ_MULTIPLE_BLOCK) starts a transfer of several consecutive blocks. Blocks will be continuously transferred until a stop command is issued. If the host uses partial blocks whose accumulated length is not block aligned and block misalignment is not allowed, the card shall detect a block misalignment at the beginning of the first mis-aligned block, set the ADDRESS_ERROR error bit in the status register, abort transmission and wait in the Data State for a stop command.

32.7.4.2 Stream Access, Stream Write and Stream Read (MMC Only)

Stream write (CMD20) starts the data transfer from the host to the card beginning from the starting address until the host issues a stop command. Since the amount of data to be transferred is not determined in advance, CRC can not be used. If the end of the memory range is reached while sending data and no stop command has been sent by the host, all further transferred data is discarded.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

There is a stream oriented data transfer controlled by READ_DAT_UNTIL_STOP (CMD11). This command instructs the card to send its payload, starting at a specified address, until the host sends a STOP_TRANSMISSION command (CMD12). The stop command has execution delay due to the serial command transmission. The data transfer stops after the end bit of the stop command. If the end of the memory range is reached while sending data and no stop command has been sent yet by the host, the contents of the further transferred payload is undefined.

32.7.4.3 Erase, Group Erase and Sector Erase (MMC Only)

It is desirable to erase many sectors simultaneously in order to enhance the data throughput. Identification of these sectors is accomplished with the TAG_* commands. Either an arbitrary set of sectors within a single erase group, or an arbitrary selection of erase groups may be erase at one time, but not both together. That is, the unit of measure for determining an erase is either a sector or an erase group. If a set of sectors must be erased, all selected sectors must lie within the same erase group. To facilitate selection, a first command with the starting address is followed by a second command with the final address, and all sectors (or groups) within this range will be selected for erase.

32.7.4.4 Wide Bus Selection/Deselection

Wide Bus (4 bit bus width) operation mode may be selected / deselected using ACMD6. The default bus width after power up or GO_IDLE (CMD0) is 1 bit bus width. ACMD6 command is valid in ‘trans state’ only. That means the bus width may be changed only after a card was selected (CMD7).

32.7.5 Protection Management

Three write protect methods are supported in the host for Cards, Card internal write protect (Card’s responsibility), Mechanical write protect switch (Host responsibility only) and Password protection card lock operation.

32.7.5.1 Card Internal Write Protection

Card data may be protected against either erase or write. The entire card may be permanently write protected by the manufacturer or content provider by setting the permanent or temporary write protect bits in the CSD. For cards which support write protection of groups of sectors by setting the WP_GRP_SIZE sectors as specified in the CSD), and the write protection may be changed by the application. The SET_WRITE_PROT command sets the write protection of the addressed write-protect group, and the CLR_WRITE_PROT command clears the write protection of the addressed write-protect group.

The SEND_WRITE_PROT command is similar to a single block read command. The card shall send a data block containing 32 write protection bits (representing 32 write protect groups starting at the specified address) followed by 16 CRC bits. The address field in the write protect commands is a group address in byte units. The card will ignore all LSB’s below the group size.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.7.5.2 Mechanical write protect switch

A mechanical sliding tablet on the side of the card will be used by the user to indicate that a given card is write protected or not. If the sliding tablet is positioned in such a way that the window is open that means the card is write protected. If the window is close the card is not write protected.

A proper, matched, switch on the socket side will indicated to the host that the card is write protected or not. It is the responsibility of the host to protect the card. The position of the write protect switch is un-known to the internal circuitry of the card.

32.7.5.3 Password Protect

The password protection feature enables the host to lock a card while providing a password, which later will be used for unlocking the card. The password and its size is kept in an 128-bit PWD and 8-bit PWD_LEN registers, respectively. These registers are non-volatile so that a power cycle will not erase them.

Locked cards respond to (and execute) all commands in the basic command class (class 0) and “lock card” command class. Thus the host is allowed to reset, initialize, select, query for status, etc., but not to access data on the card. If the password was previously set (the value of PWD_LEN is not 0) will be locked automatically after power on. Similar to the existing CSD and CID register write commands the lock/unlock command is available in “trans-state” only. This means that it does not include an address argument and the card must be selected before using it. The card lock/unlock command has the structure and bus transaction type of a regular single block write command. The transferred data block includes all the required information of the command (password setting mode, PWD itself, card lock/unlock etc.). The following table describes the structure of the command data block.

Table 32-5 Command Data Block Structure

Byte #	Bit 7-4	Bit 3	Bit 2	Bit 1	Bit 0
0	Reserved	ERASE	LOCK_UNLOCK	CLR_PWD	SET_PWD
1	PWDS_LEN				
2	Password Data				
…					
PWDS_LEN + 1					

ERASE – 1 Defines Forced Erase Operation (all other bits shall be 0) and only the command byte is sent.

LOCK/UNLOCK – 1=Locks the card. 0=Unlock the card (note that it is valid to set this bit together with SET_PWD but it is not allowed to set it together with CLR_PWD).

CLR_PWD – 1=Clears PWD.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SET_PWD – 1=Set new password to PWD.
PWD_LEN – Defines the following password length (in bytes).
PWD – The password (new or currently used depending on the command).

The data block size shall be defined by the host before it send the card lock/unlock command. This will allow different password sizes.

The following paragraphs define the various lock/unlock command sequences:

Lock command sequences:

1 Setting the Password.
   a Select a card (CMD7), if not previously selected already.
   b Define the block length (CMD16), given by the 8bit card lock/unlock mode, the 8 bits password size (in bytes), and the number of bytes of the new password. In case that a password replacement is done, then the block size shall consider that both passwords, the old and the new one, are sent with the command.
   c Send Card Lock/Unlock command with the appropriate data block size on the data line including 16-bit CRC. The data block shall indicate the mode (SET_PWD), the length (PWD_LEN) and the password itself. In case that a password replacement is done, then the length value (PWD_LEN) shall include both passwords, the old and the new one, and the PWD field shall include the old password (currently used) followed by the new password.
   d In case that the sent old password is not correct (not equal in size and content) then LOCK_UNLOCK_FAILED error bit will be set in the status register and the old password does not change. In case that PWD matches the sent old password then the given new password and its size will be saved in the PWD and PWD_LEN fields, respectively.

   NOTE:
   the password length register (PWD_LEN) indicates if a password is currently set. When it equals 0 there is no password set. If the value of PWD_LEN is not equal to zero the card will lock itself after power up. It is possible to lock the card immediately in the current power session by setting the LOCK/UNLOCK bit (while setting the password) or sending additional command for card lock.

2 Reset the password.
   a Select a card (CMD7), if not previously selected already.
   b Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password.
   c Send the card lock/unlock command with the appropriate data block size on the data line including 16-bit CRC. The data block shall indicate the mode CLR_PWD, the length (PWD_LEN) and the password (PWD) itself (LOCK/UNLOCK bit is don't care). If the PWD and PWD_LEN is set to 0. If the password is not correct then the LOCK_UNLOCK_FAILED error bit will be set in the status register.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3 Locking a card.

a Select a card (CMD7), if not previously selected already.

b Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of currently used password.

c Send the card lock/unlock command with the appropriate data block size on the data line including 16-bit CRC. The data block shall indicate the mode LOCK, the length (PWD_LEN) and the password (PWD) itself.

If the PWD content equals to the sent password then the card will be locked and the card-locked status bit will be set in the status register. If the password is not correct then LOCK_UNLOCK_FAILED error bit will be set in the status register.

NOTE:

it is possible to set the password and to lock the card in the same sequence. In such case the host shall perform all the required steps for setting the password (as described above) including the bit LOCK set while the new password command is sent. If the password was previously set (PWD_LEN is not 0), then the card will be locked automatically after power on reset. An attempt to lock a locked card or to lock a card that does not have a password will fail and the LOCK_UNLOCK_FAILED error bit will be set in the status register.

Unlock command sequences:

1 Unlocking the card.

a Select a card (CMD7), if not previously selected already.

b Define the block length (CMD16), given by the 8-bit card lock/unlock mode, the 8-bit password size (in bytes), and the number of bytes of the currently used password.

c Send the card lock/unlock command with the appropriate data block size on the data line including 16-bit CRC. The data block shall indicate the mode UNLOCK, the length (PWD_LEN) and the password (PWD) itself.

If the PWD content equals to the sent password then the card will be unlocked and the card-locked status bit will be cleared in the status register. If the password is not correct then the LOCK_UNLOCK_FAILED error bit will be set in the status register.

NOTE:

the unlocking is done only for the current power session. As long as the PWD is not cleared the card will be locked automatically on the next power up. The only way to unlock the card is by clearing the password. An attempt to unlock an unlocked card will fail and LOCK_UNLOCK_FAILED error bit will be set in the status register.

2 Forcing Erase.

In case that the user forgot the password (the PWD content) it is possible to erase all the card data content along with the PWD content. This operation is called Forced Erase.

a Select a card (CMD7), if not previously selected already.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

b Define the block length (CMD16) to 1 byte (8bit card lock/unlock command). Send the card lock/unlock command with the appropriate data block of one byte on the data line including 16-bit CRC. The data block shall indicate the mode ERASE (the ERASE bit shall be the only bit set).

If the ERASE bit is not the only bit in the data field then the LCOK_UNLOCK_FAILED error bit will be set in the status register and the erase request is rejected. If the command was accepted then ALL THE CARD CONTENT WILL BE ERASED including the PWD and PWD_LEN register content and the locked card will get unlocked.

An attempt to force erase on an unlocked card will fail and LOCK_UNLOCK_FAILED error bit will be set in the status register.

32.7.6 Card Status

The response format R1 contains a 32-bit field named card status. This field is intended to transmit the card's status information (which may be stored in a local status register) to the host. If not specified otherwise, the status entries are always related to the previous issued command.

Table below defines the different entries of the status. The type and clear condition fields in the table are abbreviate as follows:

Type:

E: Error bit.

S: Status bit..

R: Detected and set for the actual command response.

X: Detected and set during command execution. The host must poll the card by issuing the status command in order to read these bits.

Clear Condition:

A: According to the card current state.

B: Always related to the previous command. Reception of a valid command will clear it (with a delay of one command).

C: Clear by read.

Table 32-6 Card Status Description

Bits	Identifier	Type	Description	Clear Condition
31	OUT_OF_RAGE	E R	The command's argument was out of the allowed range for this card. 0: No Error 1: Error	C
30	ADDRESS_ERROR	E R X	A misaligned address which did not match the block length was used in	C

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			the command. 0: No Error 1: Error	
29	BLOCK_LEN_ERROR	E R	The transferred block length is not allowed for this, or the number of transferred bytes does not match the block length. 0: No Error 1: Error	C
28	ERASE_SEQ_ERROR	E R	An error in the sequence of erase commands occurred. 0: No Error 1: Error	C
27	ERASE_PARAM	E X	An invalid selection of sectors or groups for erase occurred. 0: No Error 1: Error	C
26	WP_VIOLATION	E R X	Attempt to program a write protected block. 0: No Protected 1: Protected	C
25	CARD_IS_LOCKED	S X	When set, signals that the card is locked by the host. 0: Card unlocked 1: Card locked	A
24	LOCK_UNLOCK_FAILED	E R X	Set when a sequence or password error has been detected in lock/unlock card command or if there was an attempt to access a locked card. 0: No Error 1: Error	C
23	COM_CRC_ERROR	E R	The CRC check of the previous command failed. 0: No Error 1: Error	B
22	ILLEGAL_COMMAND	E R	Command not legal for the card state. 0: No Error 1: Error	B
21	CARD_ECC_FAILED	E X	Card internal ECC was applied but failed to correct the data. 0: normal	C

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			1: failure	
20	CC_ERROR	E R X	Internal card controller error. 0: No Error 1: Error	C
19	ERROR	E R X	A general or an unknown error occurred during the operation. 0: No Error 1: Error	C
18	UNDERRUN	E X	The card could not sustain data transfer in stream read mode. 0: No Error 1: Error	C
17	OVERRUN	E X	The card could not sustain data programming in stream write mode. 0: No Error 1: Error	C
16	CID/CSD_OVERWRITE	E R X	Can be either one of the following errors. 0: No Error 1: Error	C
15	WP_ERASE_SKIP	S X	Only partial address space was erased due to existing write protected blocks. 0: No Protected 1 : Protected	C
14	CARD_ECC_DISABLED	S X	The command has been executed without using the internal ECC. 0: enabled 1: disabled	A
13	ERASE_RESET	S R	An erase sequence was cleared before executing because an out of erase sequence command was received. 0: normal 1: set	C
12:9	CURRENT_STATE	S X	The state of the card when receiving the command. If the command execution causes a state change, it will be visible to the host in the response to the next command. The four bits are interpreted as binary coded number between 0 and 15. 0: idle	B

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			1: ready 2: identify 3: standby 4: transition 5: data 6: receive 7: program 8 : disable (9 – 15) : received	
8	READY_FOR_DATA	S X	Corresponds to buffer empty signaling on the bus. 0: No Ready 1: Ready	A
7:6	Reserved	-	-	-
5	APP_CMD	S R	The card will expect ACMD, or indication that the command has been interpreted as ACMD. 0: Disable 1: Enable	C
4:0	Reserved	-	-	-

32.7.7 SD Status

The SD status contains status bits that are related to the SD card proprietary features and may be used for future application specific usage. The size of the SD status is one data block of 512bit. The content of this register is transmitted to the Host over the DAT bus along with 16-bit CRC. The SD status is sent to the host over the DAT bus if ACMD13 is sent (CMD55 followed with CMD13). ACMD13 can be sent to a card only in tran_state (card is selected). SD status structure is described in below.

The same abbreviation for *type* and *clear condition* were used as for the Card Status above.

Table 32-7 SD Status Structure

Bits	Identifier	Type	Description	Clear Condition
511:510	DAT_BUS_WIDTH	S R	Shows the currently defined data bus width that was defined by SET_BUS_WIDTH command. 00: 1 (default) 01: Reserved 10: 4 bit width 11: Reserved	A
509	SECURED_MODE	S R	Card is in Secured Mode of	A

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

			operation. 0: Not in the Mode 10: In the mode	
508:496	Reserved.			
495:480	SD_CARD_TYPE	S R	All 0, is SD Memory cards.	A
479:448	SIZE_OF_PROTECTED_AREA	S R	Size of protected area.	A
447:312	Reserved.			
311:0	Reserved for manufacturer.			

32.7.8 SDIO

I/O access differs from memory in that the registers can be written and read individually and directly without a FAT file structure or the concept of blocks (although block access is supported). These registers allow access to the IO data, control of the IO function, and report on status or transfer I/O data to and from the host.

Each SDIO card may have from 1 to 7 functions plus one memory function built into it. A function is a self contained I/O device. I/O functions may be identical or completely different from each other. All I/O functions are organized as a collection of registers, and there is a maximum of 131,072 registers possible for each I/O function.

32.7.8.1 SDIO Interrupts

In order to allow the SDIO card to interrupt the host, and interrupt function is added to a pin on the SD interface. Pin number 8 which is used as DAT[1] when operating in the 4 bit SD mode is used to signal the card's interrupt to the host. The use of interrupt is optional for each card or function within a card. The SDIO interrupt is "level sensitive", that is, the interrupt line must be held active (low) until it is either recognized and acted upon by the host or de-asserted due to the end of the Interrupt Period. Once the host has serviced the interrupt, it is cleared via an IO write to the appropriate bit in the CCCR. The interrupt output of all SDIO cards is active low.

As Pin 8 of the card is shared between the IRQ and DAT[1] use in the 4 bit SD mode, and interrupt shall only be sent by the card and recognized by the host during a specific time. The time that a low on Pin 8 will be recognized as an interrupt is defined as the Interrupt Period.

The host here will only sample the level of Pin 8 (DAT[1]/IRQ) into the interrupt detector during the Interrupt Period. At all other times, the host will ignore the level on Pin 8. Note that the Interrupt Period is applicable for both memory and IO operations. The definition of the Interrupt Period is different for operations with single block and multiple block data transfer.

32.7.8.2 SDIO Suspend/Resume

Within a multi-function SDIO or a Combo (Mix IO and Memory) card, there are multiple devices (I/O

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

and memory) that must share access to the SD bus. In order to allow the sharing of access to the host among multiple devices, SDIO and combo cards can implement the optional concept of suspend/resume. In a card supports suspend/resume, the host may temporarily halt a data transfer operation to one function or memory (suspend) in order to free the bus for a higher priority transfer to a different function of memory. Once this higher-priority transfer is complete, the original transfer is re-started where it left off (resume). The host controller here is supported by all IO functions except zero, and the memory of a combo card, and can suspend multiple transactions and resume them in any order desired. IO function zero does not support suspend/resume.

The procedure used to perform the Suspend/Resume operation on the SD bus is:

- The host determines which function currently used the DAT[] line(s).
- The host requests the lower priority or slower transaction to suspend.
- The host checks for the transaction suspension to complete.
- The host begins the higher priority transaction.
- The host waits for the completion of the higher priority transaction.
- The host restores the suspended transaction.

32.7.8.3 SDIO Read Wait

The optional Read Wait (RW) operation is defined only for the SD 1-bit and 4-bit modes. The read wait operation allows a host to signal a card that it is doing a read multiple (CMD53) operation to temporarily stall the data transfer while allowing the host to send commands to any function within the SDIO device. To determine if a card supports the Read Wait protocol, the host must test capability bits in CCCR. The timing for Read Wait is base on the Interrupt Period.

32.7.9 Clock Control

The software should guarantee that the card identification process starts in open-drain mode with the clock rate fod (0 ~ 400khz). In addition, the software should also make the card into interrupt mode with fod (only for MMC). The commands that require fod are CMD0, CMD1, CMD2, CMD3, CMD5, CMD40 and ACMD41. In data transfer mode, the MSC controller can operate card with clock rate fpp (0 ~ 25Mhz).

32.7.10 Application Specified Command Handling

The MMC/SD system is designed to provide a standard interface for a variety applications types. In this environment it is anticipate that there will be a need for specific customers/applications features. To enable a common way of implementing these features, two types of generic commands are defined in the standard: Application Specific Command, ACMD, and General Command, GEN_CMD.

GEN_CMD, this command, when received by the card, will cause the card to interpret the following command as an application specific command, ACMD. The ACMD has the same structure as of regular MMC standard commands and it may have the same CMD number. The card will recognize it

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

as ACMD by the fact that it appears after APP_CMD.

The only effect of the APP_CMD is that if the command index of the, immediately, following command has an ACMD overloading, the none standard version will used. If, as an example, a card has a definition for ACMD13 but not for ACMD7 then, if received immediately after APP_CMD command, Command 13 will be interpreted as the non standard ACMD13 but, command 7 as the standard CMD7.

In order to use one of the manufacturer specific ACMD's the host will:

1. Send APP_CMD. The response will have the APP_CMD bit (new status bit) set signaling to the host that ACMD is now expected.
2. Send the required ACMD. The response will have the APP_CMD bit set, indicating that the accepted command was interpreted as ACMD. If a non-ACMD is sent then it will be respected by the card as normal MMC command and the APP_CMD bit in the Card Status stays clear.

If a non valid command is sent (neither ACMD nor CMD) then it will be handled as a standard MMC illegal command error.

The bus transaction of the GEN_CMD is the same as the single block read or write commands (CMD24 or CMD17). The difference is that the argument denotes the direction of the data transfer (rather than the address) and the data block is not a memory payload data but has a vendor specific format and meaning.

The card shall be selected before sending CMD56. The data block size is the BLOCK_LEN that was defined with CMD16. The response to CMD56 will be R1b (card status + busy indication).

32.8 MMC/SD Controller Operation

32.8.1 Data FIFOs

The controller FIFOs for the response tokens, received data, and transmitted data are MSC_RES, MSC_RXFIFO, and MSC_TXFIFO, respectively. These FIFOs are accessible by the software and are described in the following paragraphs.

32.8.1.1 Response FIFO (MSC_RES)

The response FIFO, MSC_RES, contains the response received from an MMC/SD card after a command is sent from the controller. MSC_RES is a read-only, 16-bit, and 8-entry deep FIFO.

The FIFO will hold all possible response lengths. Responses that are only one byte long are located on the LSB of the 16-bit entry in the FIFO. The first half-word read from the response FIFO is the most significant half-word of the received response. For example, if the response format is R1, then the response read from RES_FIFO is bit [47:32], bit[31:16], bit[15:0] and in the third half-word only the low

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8-bit is effective response [15:8] and the high 8-bit is ignored. If the response format is R2, then the response read from MSC_RES is bit [135:8] and needs reading 8 times.

The FIFO does not contain the response CRC. The status of the CRC check is in the status register, MSC_STAT.

32.8.1.2 Receive/Transmit Data FIFO (MSC_RXFIFO/MSC_TXFIFO)

The receive data FIFO and transmit data FIFO share one 16-entry x 32-bit FIFO, because at one time data are only received or are only transmitted. If it is used to receive data, it is called MSC_RXFIFO and read-only. If it is used to transmit data, it is called MSC_TXFIFO and write-only.

Data FIFO and its controls are cleared to a starting state after a system reset or at the beginning of the operations which include data transfer. (MSC_CMDAT[DATA_EN] == 1)

If at any time MSC_RXFIFO becomes full and the data transmission is not complete, the controller turns the MSC_CLK off to prevent any overflows. When the clock is off, data transmission from the card stops until the clock is turned back on. After MSC_RXFIFO is not full, the controller turns the clock on to continue data transmission. The full status of the FIFO is registered in the MSC_STAT [DATA_FIFO_FULL] bit.

If at any time MSC_TXFIFO becomes empty and the data transmission is not complete, the controller turns the MSC_CLK off to prevent any under run. When the clock is off, data transmission to the card stops until the clock is turned back on. When MSC_TXFIFO is no longer empty, the controller automatically restarts the clock. The empty status of the FIFO is registered in the MSC_STAT [DATA_FIFO_EMPTY] bit.

The FIFO is readable on word (32-bit) boundaries. The max read/written number is 16 words. The controller can correctly process big-endian and little-endian data.

Because at the beginning of the operation which include data transfer (MSC_CMDAT [DATA_EN] == 1), Data FIFO and its controls are cleared, software should guarantee data in FIFO have been read/written before beginning a new command.

32.8.2 DMA and Program I/O

Software may communicate to the MSC controller via the DMA or program I/O. The SDMA and CDMA are supported at the same time. SDMA is the special DMA in MSC controller, while CDMA is the common DMA in DMAC controller.

32.8.2.1 Structure of SDMA Descriptor

SDMA is a descriptor DMA. And a descriptor can be linked to another one by the NDA until the LINK in

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

the 4th word is set to 1b.

A SDMA descriptor includes 4 words.

Word 1: The physical address of the next SDMA descriptor.

Bits	Name	Description
31:0	NDA	Next descriptor physical address. It should be 4-word aligned.

Word 2: The system memory address where the data will be read from or stored to.

Bits	Name	Description
31:0	DA	Data Address.

Word 3: The length of data to be transferred.

Bits	Name	Description
31:0	LEN	Unit is byte.

Word 4: The command for the current SDMA.

Bits	Name	Description
31:24	Reserved	It has no use in the current version.
23:16	ID	Identification of Current DMA Transfer.
15:2	Reserved	It has no use in the current version.
1	ENDI	Interrupt Enable for Current DMA Transfer End 0: Disable interrupt 1: Enable interrupt
0	LINK	Control the end of DMA Descriptor. 0: DMA will continue to fetch another descriptor. 1: DMA will go to idle state after the current transfer. The MSC_STAT.DMAEND will be set to 1b when the current read/write is finished. If the interrupt mask bit MSC_IMASK.DMAEND is 0b, the interrupt flag MSC_IFLG. DMAEND will be set to 1b also.

32.8.2.2 Operation of SDMA

Step 1: Prepare the descriptor is system memory

Step 2: If the address is not word boundary, it is suggested to configure MSC_DMAC.ALIGNEN and MSC_DMAC.AOFST

Step 3: Write the descriptor address to MSC_DMANDA

Step 4: Configure MSC controller to read or write operation, e,g, set MSC_NOB, MSC_BLEN and

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

MSC_CMDAT.WRITE_READ etc.

Step 5: If DMAEND interrupt is wanted, clear MSC_IMASK.DMAEND, otherwise, set this bit

Step 6: Select SDMA by setting MSC_DMAC.DMASEL to 0b

Step 7: Start DMA transfer by setting MSC_DMAC.DMAEN

Step 8: Waiting the DMAEND interrupt (if interrupt is used) or status (if interrupt is not used)

Step 9: Disable the SDMA by configuring MSC_DMAC.DMAEN to 0b

32.8.2.3 Operation of CDMA

To access MSC_RXFIFO/MSC_TXFIFO with the DMA, the software must program the DMA to read or write the FIFO with source port width 32-bit, destination port width 32-bit, transfer data size 32-byte, transfer mode single. For example, to write 64 bytes of data to the MSC_TXFIFO, the software must program the DMA as follows:

```
DMA_DCTRn = 2              // Write 2 32-bytes (64 bytes)
DMA_DCCRn[SWDH] = 0        // source port width is 32-bit
DMA_DCCRn[DWDH] = 0        // destination port width is 32-bit
DMA_DCCRn[DS] = 4          // transfer data size is 32-byte
DMA_DCCRn[TM] = 4          // transfer mode is single
DMA_DCCRn[RDIL] = 0        // request detection interval length is 0
```

The number of 32-bytes should be calculated from the number of transferred bytes as follows:

The number of words = (The number of bytes + 31) / 32

If the number of transferred bytes is not the multiple of 4, the controller can correctly process endian.

The DMA trigger level is 8 words, that is to say, the DMA read trigger is when data words in MSC_RXFIFO is >= 8 and the DMA write trigger is when data words in MSC_TXFIFO is < 8. Software can also configure DMA registers based on requirements, but the above 32-byte transfer data size is most efficient.

32.8.2.4 Operation of Program I/O

With program I/O, the software waits for the MSC_IFLG [RXFIFO_RD_REQ] or MSC_IFLG [TXFIFO_WR_REQ] interrupts before reading or writing the respective FIFO.

NOTES:

1. The MSC_CMDAT [DMA_EN] bit must be set to a 1 to enable communication with the DMA and it must be set to a 0 to enable program I/O.
2. DMA can be enabled only after MSC_CMDAT is written, because MSC_CMDAT [DATA_EN] is used to reset TX/RXFIFO.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.8.3 Start and Stop clock

The software stops the clock as follows:

1 Write MSC_CTRL with 0x01 to stop the MMC/SD bus clock.
2 Wait until MSC_STAT[CLK_EN] becomes zero.

To start the clock the software writes MSC_CTRL with 0x02.

32.8.4 Software Reset

Reset includes the MSC reset and the card reset.

The MSC reset is through MSC_CTRL [RESET] bit.

The card reset is to make the card into idle state. CMD0 (GO_IDLE_STATE) sets the MMC and SD memory cards into idle state. CMD52 (IO_RW_DIRECT, with argument 0x88000C08) reset the SD I/O card. The MMC/SD card are initialized with a default relative card address (RCA = 0x0001 for MMC and RCA = 0x0000 for SD) and with a default driver stage register setting (lowest speed, highest driving current capability).

The following registers must be set before the clock is started:

Step 1. Stop the clock.
Step 2. Set MSC_CTRL register to 0x08 to reset MSC.
Step 3. Wait while MSC_STAT [IS_RESETTING] is 1.
Step 4. Set MSC_CMD with CMD0.
Step 5. Update the MSC_CMDAT register as follows:
  a Write 0x0000 to MSC_CMDAT [RESPONSE_FORMAT].
  b Clear the MSC_CMDAT [DATA_EN] bit.
  c Clear the MSC_CMDAT [BUSY] bit.
  d Clear the MSC_CMDAT [INIT] bit.
Step 6. Start the clock.
Step 7. Start the operation. (write MSC_CTRL with 0x04)
Step 8. Wait for the END_CMD_RES interrupt.
Step 9. Set MSC_CMD with CMD52.
Step 10. Set MSC_ARG with 0x88000C08.
Step 11. Update the MSC_CMDAT register as follows:
  a Write 0x005 to MSC_CMDAT [RESPONSE_FORMAT].
  b Clear the MSC_CMDAT [DATA_EN] bit.
  c Clear the MSC_CMDAT [BUSY] bit.
  d Clear the MSC_CMDAT [INIT] bit.
Step 12. Start the operation.
Step 13. Wait for the END_CMD_RES interrupt.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.8.5 Voltage Validation and Card Registry

At most 10 MMC and 1 SD (either SDMEM or SDIO) can be inserted MMC/SD bus at the same time, and their voltage validation and card registry steps are different, so the software should be programmed as follows:

Step 1. Check whether SDIO card is inserted.
Step 2. Check whether SDMEM card is inserted.
Step 3. Check whether MMC cards are inserted.

32.8.5.1 Check SDIO

The commands are sent as follows:

Step 1. (Optional) Send CMD52 (IO_RW_DIRECT) with argument 0x88000C08 to reset SDIO card.
Step 2. Send CMD5 (IO_SEND_OP_CMD) to validate voltage.
Step 3. If the response is correct and the number of IO functions > 0, then continue, else go to check SDMEM.
Step 4. If C-bit in the response is ready (the initialization has finished), go to 6.
Step 5. Send CMD5 (IO_SEND_OP_CMD) to validate voltage, then go to 4.
Step 6. If memory-present-bit in the response is true, then it is a combo card (SDIO + Memory), else it is only a SDIO card.
Step 7. If it is a combo card, go to check SDMEM to initialize the memory part.
Step 8. Send CMD3 (SET_RELATIVE_ADDR) to let the card publish a RCA. The RCA is returned from the response.
Step 9. If do not accept the new RCA, go to 8, else record the new RCA.
Step 10. Go to check MMC, because we can assure that there is no SDMEM card.

32.8.5.2 Check SDMEM

If there is no SDIO card or there is a combo card, continue to check SDMEM.

The commands are sent as follows:

Step 1. (Optional) Send CMD0 (GO_IDLE_STATE) to reset MMC and SDMEM card. This command has no response.
Step 2. Send CMD55. Here the default RCA 0x0000 is used for CMD55.
Step 3. If the response is correct (CMD55 has response), then continue, else go to check MMC.
Step 4. Send ACMD41 (SD_SEND_OP_CMD) to validate voltage (the general OCR value is 0x00FF8000).
Step 5. If the initialization has finished, go to 7. (The response is the OCR register and it includes a status information bit (bit [31]). This status bit is set if the card power up procedure has been finished. As long as the card is busy, the corresponding bit[31] is set to LOW.)
Step 6. Send CMD55 and ACMD41 to validate voltage, and then go to 5.
Step 7. Send CMD2 (ALL_SEND_CID) to get the card CID.
Step 8. Send CMD3 (SET_RELATIVE_ADDR) to let card publish a RCA. The RCA is returned

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

from the response.

Step 9. If do not accept the new RCA, go to 8, else record the new RCA.

Step 10. Go to check MMC.

32.8.5.3 Check MMC

Because there may be several MMC card, so some steps (5 ~ 8) should be repeated several times.

The commands are sent as follows:

Step 1. Send CMD1 (SEND_OP_CMD) to validate voltage (the general OCR value is 0x00FF88000).

Step 2. If the response is correct, then continue, else go to 9.

Step 3. If the initialization has finished, go to 5. (The response is the OCR register and it includes a status information bit (bit [31]). This status bit is set if the card power up procedure has been finished. As long as the card is busy, the corresponding bit[31] is set to LOW.)

Step 4. Send CMD1 (SEND_OP_CMD) to validate voltage, and then go to 3.

Step 5. Send CMD2 (ALL_SEND_CID) to get the card CID.

Step 6. If the response timeout occurs, go to 9.

Step 7. Send CMD3 (SET_RELATIVE_ADDR) to assign the card a RCA.

Step 8. If there are other MMC cards, then go to 5.

Step 9. Finish.

32.8.6 Single Data Block Write

In a single block write command, the following registers must be set before the operation is started:

Step 1. Set MSC_NOB register to 0x0001.

Step 2. Set MSC_BLKLEN to the number of bytes per block.

Step 3. Update the MSC_CMDAT register as follows:

a Write 0x001 to MSC_CMDAT [RESPONSE_FORMAT].

b Write 0x2 to MSC_CMDAT [BUS_WIDTH] if the card is SD, else clear it.

c Set the MSC_CMDAT [DATA_EN] bit.

d Set the MSC_CMDAT [WRITE_READ] bit.

e Clear the MSC_CMDAT [STREAM_BLOCK] bit.

f Clear the MSC_CMDAT [BUSY] bit.

g Clear the MSC_CMDAT [INIT] bit.

Step 4. Start the operation.

Step 5. Write MSC_IMASK with some value to unmask the expected interrupts.

Then the software must perform the following steps:

Step 1. Wait for the MSC_IFLG [END_CMD_RES] interrupt.

Step 2. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.

At the same time write data to the MSC_TXFIFO and continue until all of the data have been written to the FIFO.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Step 3. Wait for MSC_IFLG [PROG_DONE] interrupt. This interrupt indicates that the card has finished programming. Certainly software may start another command sequence on a different card.

Step 4. Read the MSC_STAT register to verify the status of the transaction (i.e. CRC error status).

To address a different card, the software sends a select command to that card by sending a basic no data command and response transaction. To address the same card, the software must wait for MSC_IFLG [PROG_DONE] interrupt. This ensures that the card is not in the busy state.

In addition, CMD26 (PROGRAM_CID), CMD27 (PROGRAM_CSD), CMD42 (LOCK/UNLOCK), CMD56 (GEN_CMD: write) and CMD53 (single block write) operations are similar to single block write.

32.8.7 Single Block Read

In a single block read command, the following registers must be set before the operation is started:

Step 1. Set MSC_NOB register to 0x0001.

Step 2. Set MSC_BLKLEN register to the number of bytes per block.

Step 3. Update the following bits in the MSC_CMDAT register:

a Write 0x001 to MSC_CMDAT [RESPONSE_FORMAT].

b Write 0x2 to MSC_CMDAT [BUS_WIDTH] if the card is SD, else clear it.

c Set the MSC_CMDAT [DATA_EN] bit.

d Clear the MSC_CMDAT [WRITE_READ] bit.

e Clear the MSC_CMDAT [STREAM_BLOCK] bit.

f Clear the MSC_CMDAT [BUSY] bit.

g Clear the MSC_CMDAT [INIT] bit.

Step 4. Start the operation.

Step 5. Write MSC_IMASK with some value to unmask the expected interrupts.

Then the software must perform the following steps:

Step 1. Wait for the MSC_IFLG [END_CMD_RES] interrupt.

Step 2. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.

At the same time read data from the MSC_RXFIFO as data becomes available in the FIFO, and continue reading until all data is read from the FIFO.

Step 3. Read the MSC_STAT register to verify the status of the transaction (i.e. CRC error status).

In addition, CMD30 (SEND_WRITE_PROT), ACMD13 (SD_STATUS), CMD56 (GEN_CMD-read), ACMD51 (SEND_SCR) and CMD53 (single block read) are similar to single block read.

32.8.8 Multiple Block Write

The multiple block write mode is similar to the single block write mode, except that multiple blocks of data are transferred. Each block is the same length. All the registers are set as they are for the single

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

block write, except that the MSC_NOB register is set to the number of blocks to be written.

The multiple block write mode also requires a stop transmission command, CMD12, after the data is transferred to the card. After the MSC_IFLG [DATA_TRAN_DONE] interrupt occurs, the software must program the controller register to send a stop data transmission command.

If multiple block write with pre-defined block count (refer to MMC spec v-3.3) is used, CMD12 should not be sent.

For SDIO card, CMD53 (multiple block write) is also similar, but when IO abort (CMD52) is sent, MSC_CMDAT [IO_ABORT] should be 1.

Table 32-8 How to stop multiple block write

Operation	Stop condition	Software processing
Open-ended or SDIO infinite	After write MSC_NOB blocks	1 Wait for DATA_TARN_DONE interrupt. 2 Send CMD12 or CMD52. (IO abort) 3 Wait for END_CMD_RES and PRG_DONE interrupt.
Open-ended or SDIO infinite	Stop writing in advance (not write MSC_NOB blocks)	1 Set MSC_CTRL [EXIT_MULTIPLE]. 2 Wait for DATA_TRAN_DONE interrupt. 3 Send CMD12 or CMD52. (IO abort) 4 Wait for END_CMD_RES and PRG_DONE interrupt.
Predefined block or SDIO finite	After writing MSC_NOB blocks	1 Wait for DATA_TRAN_DONE interrupt.
Predefined block or SDIO finite	Stop writing in advance (not write MSC_NOB blocks)	1 Set MSC_CTRL [EXIT_MULTIPLE]. 2 Wait for DATA_TRAN_DONE interrupt. 3 Send CMD12 or CMD52. (IO abort) 4 Wait for END_CMD_RES and PRG_DONE interrupt.

32.8.9 Multiple Block Read

The multiple blocks read mode is similar to the single block read mode, except that multiple blocks of data are transferred. Each block is the same length. All the registers are set as they are for the single block read, except that the MSC_NOB register is set to the number of blocks to be read.

The multiple blocks read mode requires a stop transmission command, CMD12, after the data from the card is received. After the MSC_IFLG [DATA_TRAN_DONE] interrupt has occurred, the software must program the controller registers to send a stop data transmission command.

If multiple block read with pre-defined block count (refer to MMC spec v-3.3) is used, CMD12 should not be sent.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

For SDIO card, CMD53 (multiple block read) is also similar, but when IO abort (CMD52) is sent, MSC_CMDAT [IO_ABORT] should be 1.

Table 32-9 How to stop multiple block read

Operation	Stop condition	Software processing
Open-ended or SDIO infinite	After reading MSC_NOB blocks	1 Wait for DATA_TRAN_DONE interrupt. 2 Send CMD12 or CMD52. (IO abort) 3 Wait for END_CMD_RES interrupt.
Open-ended or SDIO infinite	Stop reading in advance (not write MSC_NOB blocks)	1 Set MSC_CTRL [EXIT_MULTIPLE]. 2 Wait for DATA_TRAN_DONE interrupt. 3 Send CMD12 or CMD52. (IO abort) 4 Wait for END_CMD_RES interrupt.
Predefined block or SDIO finite	After reading MSC_NOB blocks	1 Wait for DATA_TRAN_DONE interrupt.
Predefined block or SDIO finite	Stop reading in advance (not write MSC_NOB blocks)	1 Set MSC_CTRL [EXIT_MULTIPLE]. 2 Wait for DATA_TRAN_DONE interrupt. 3 Send CMD12 or CMD52. (IO abort) 4 Wait for END_CMD_RES interrupt.

32.8.10 Stream Write (MMC)

In a stream write command, the following registers must be set before the operation is started:

1. Update MSC_CMDAT register as follows:
   - a Write 0x001 to the MSC_CMDAT [RESPONSE_FORMAT].
   - b Clear the MSC_CMDAT [BUS_WIDTH] because only MMC support stream write.
   - c Set the MSC_CMDAT [DATA_EN] bit.
   - d Set the MSC_CMDAT [WRITE_READ] bit.
   - e Set the MSC_CMDAT [STREAM_BLOCK] bit.
   - f Clear the MSC_CMDAT [BUSY] bit.
   - g Clear the MSC_CMDAT [INIT] bit.
2. Start the operation.
3. Write MSC_IMASK with some value to unmask the expected interrupts.

Then the software must perform the following steps:

1. Wait for the MSC_IFLG [END_CMD_RES] interrupt.
2. Write data to the MSC_TXFIFO and continue until all of the data is written to the Data FIFO.
3. Stop clock. Wait until MSC_STAT[CLK_EN] becomes 0. The clock must be stopped.
4. Set the command registers for a stop transaction command (CMD12) and other registers.
5. Start the clock and start the operation.
6. Wait for the MSC_IFLG [END_CMD_ERS] interrupt.
7. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

8 Wait for the MSC_IFLG [PRG_DONE] interrupt. This interrupt indicates that the card has finished programming. Certainly software may start another command sequence on a different card.
9 Read the MSC_STAT register to verify the status of the transaction.

To address a different card, the software must send a select command to that card by sending a basic no data command and response transaction. To address the same card, the software must wait for MSC_IFLG [PRG_DONE] interrupt. This ensures that the card is not in the busy state.

If partial blocks are allowed (if CSD parameter WRITE_BL_PARTIAL is set) the data stream can start and stop at any address within the card address space, otherwise it shall start and stop only at block boundaries. If WRITE_BL_PARTIAL is not set, 16 more stuff bytes need to be written after the useful written data, otherwise only write the useful written data.

32.8.11 Stream Read (MMC)

In a stream read command, the following registers must be set before the operation is turned on:

1 Update the MSC_CMDAT register as follows:
   a Write 0x01 to the MSC_CMDAT [RESPONSE_FORMAT].
   b Clear the MSC_CMDAT [BUS_WIDTH] because only MMC support stream read.
   c Clear the MSC_CMDAT [WRITE_READ] bit.
   d Set the MSC_CMDAT [STREAM_BLOCK] bit.
   e Clear the MSC_CMDAT [BUSY] bit.
   f Clear the MSC_CMDAT [INIT] bit.
2 Start the operation.
3 Write MSC_IMASK with some value to unmask the expected interrupts.

Then the software must perform the following steps:

1 Wait for the MSC_IFLG [END_CMD_RES] interrupt.
2 Read data from the MSC_RXFIFO and continue until all of the expected data has been read from the FIFO.
3 Write MSC_CTRL [EXIT_TRANSER] with 1. If MSC_STAT[DATA_FIFO_FULL] is 1, then read MSC_RXFIFO to make it not full. Because if data FIFO is full, MSC_CLK is stopped. Here, the data FIFO contains useless data.
4 Set the command registers for a stop transaction command (CMD12) and send it. There is no need to stop the clock.
5 Wait for the MSC_IFLG [END_CMD_RES].
6 Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.
7 Read the MSC_STAT register to verify the status of the transaction.

32.8.12 Erase, Select/Deselect and Stop

For CMD7 (SELECT/DESELECT_CARD), CMD12 (STOP_TRANSMISSION) and CMD38 (ERASE),

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

the following registers must be set before the operation is started:

1. Update the MSC_CMDAT register as follows:
   a. Write 0x01 to the MSC_CMDAT [RESPONSE_FORMAT].
   b. Clear the MSC_CMDAT [DATA_EN] bit.
   c. Clear the MSC_CMDAT [WRITE_READ] bit.
   d. Clear the MSC_CMDAT [STREAM_BLOCK] bit.
   e. Set the MSC_CMDAT [BUSY] bit.
   f. Clear the MSC_CMDAT [INIT] bit.
2. Start the operation.
3. Write MSC_IMASK with some value to unmask the expected interrupts.

Then the software must perform the following steps:

1. Wait for the MSC_IFLG [END_CMD_RES] interrupt.
2. Wait for the MSC_IFLG [PRG_DONE] interrupt. If CMD12 is sent to terminate data read operation, then there is no need to wait for MSC_IFLG [PRG_DONE] interrupt. This interrupt indicates that the card has finished programming. Certainly software may start another command sequence on a different card.

32.8.13 SDIO Suspend/Resume

The actual suspend/resume steps are as follows:

1. During data transfer, send CMD52 to require suspend. BR and RAW flag should be 1.
2. If BS flag in the response is 0, then suspend has been accepted and goto 4.
3. Send CMD52 to query card status. R flag should be 1. Go to 2.
4. Write MSC_CTRL [EXIT_TRANSFER] with 1.
5. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.
6. Read MSC_NOB, MSC_SNOB and etc, save them into variables.
7. Set registers for high priority transfer and start it.
8. Wait until high priority transfer is finished.
9. Restore registers from variables, but MSC_NOB should be (MSC_NOB – MSC_SNOB).
10. Send CMD52 to require resume. FSx should be resumed function number.

32.8.14 SDIO Read Wait

The actual Read Wait steps are as follows:

1. During multiple block read, read MSC_SNOB. If MSC_SNOB is nearby or equal to MSC_NOB, no need to use Read Wait.
2. Write MSC_CTRL [START_READ WAIT] with 1.
3. Wait until MSC_STAT [IS_READ WAIT] becomes 1.
4. Send CMD52 to query card status.
5. Write MSC_CTRL [STOP_READ WAIT] with 1.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

32.8.15 Operation and Interrupt

The software can use polling-status method to operate the MMC/SD card, but this is not the proposed method, because its performance is very low. The proposed method is to use interrupt. Generally there are fixed necessary steps to finish each command. The steps are as follows:

1. (Optional) Stop clock. Poll CLK_EN.
2. Fill the registers (MSC_CMD, MSC_CMDAT, MSC_ARG, MSC_CLKRT, and etc).
3. (Optional) Start clock.
4. Start the operation. Wait for the MSC_IFLG [END_CMD_RES] interrupt.
5. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.
6. Send STOP_TRANS (CMD12) or I/O abort (CMD52). Wait for the MSC_IFLG [END_CMD_ERS] interrupt.
7. Wait for the MSC_IFLG [DATA_TRAN_DONE] interrupt.
8. Wait for the MSC_IFLG [PRG_DONE] interrupt.

Table 32-10 The mapping between Commands and Steps

Index	Abbreviation	1	2	3	4	5	6	7	8	Comments
CMD0	GO_IDLE_STATE	Y	Y	Y	Y					
CMD1	SEND_OP_COND	Y	Y	Y	Y					
CMD2	ALL_SEND_CID	Y	Y	Y	Y					
CMD3	SET_RELATIVE_ADDR	Y	Y	Y	Y					
CMD4	SET_DSR	Y	Y	Y	Y					
CMD7	SELECT/DSELECT_CARD	Y	Y	Y	Y				Y	
CMD9	SEND_CID	Y	Y	Y	Y					
CMD10	SEND_CSD	Y	Y	Y	Y					
CMD11	READ_DAT_UNTIL_STOP	Y	Y	Y	Y		Y	Y		
CMD12	STOP_TRANSMISSION	Y	Y	Y	Y				Y	
CMD13	SEND_STATUS	Y	Y	Y	Y					
CMD15	GO_INACTIVE_STATE	Y	Y	Y	Y					
CMD16	SET_BLOCKLEN	Y	Y	Y	Y					
CMD17	READ_SINGLE_BLOCK	Y	Y	Y	Y	Y				
CMD18	READ_MULTIPLE_BLOCK	Y	Y	Y	Y	Y	Y			Open-ended
CMD18	READ_MULTIPLE_BLOCK	Y	Y	Y	Y	Y				Predefine blocks
CMD20	WRITE_DAT_UNTIL_STOP	Y	Y	Y	Y		Y	Y	Y	
CMD23	SET_BLOCK_COUNT	Y	Y	Y	Y					
CMD24	WRITE_SINGLE_BLOCK	Y	Y	Y	Y	Y			Y	
CMD25	WRITE_MULTIPLE_BLOCK	Y	Y	Y	Y	Y	Y		Y	Open-ended
CMD25	WRITE_MULTIPLE_BLOCK	Y	Y	Y	Y	Y			Y	Predefine blocks
CMD26	PROGRAM_CID	Y	Y	Y	Y	Y			Y	
CMD27	PROGRAM_CSD	Y	Y	Y	Y	Y			Y	
CMD28	SET_WRITE_PROT	Y	Y	Y	Y				Y	
CMD29	CLR_WRITE_PROT	Y	Y	Y	Y				Y	
CMD30	SEND_WRITE_PROT	Y	Y	Y	Y	Y				

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

CMD32	ERASE_WR_BLOCK_START	Y	Y	Y	Y					
CMD33	ERASE_WR_BLOCK_END	Y	Y	Y	Y					
CMD35	ERASE_GROUP_START	Y	Y	Y	Y					
CMD36	ERASE_GROUP_END	Y	Y	Y	Y					
CMD38	ERASE	Y	Y	Y	Y				Y	
CMD39	FAST_IO	Y	Y	Y	Y					
CMD40	GO_IRQ_STATE	Y	Y	Y	Y					
CMD42	LOCK/UNLOCK	Y	Y	Y	Y	Y			Y	
CMD55	APP_CMD	Y	Y	Y	Y					
CMD56	GEN_CMD	Y	Y	Y	Y	Y				Read
CMD56	GEN_CMD	Y	Y	Y	Y	Y			Y	Write
ACMD6	SET_BUS_WIDTH	Y	Y	Y	Y					
ACMD13	SD_STATUS	Y	Y	Y	Y	Y				
ACMD22	SEND_NUM_WR_BLOCKS	Y	Y	Y	Y					
ACMD23	SET_WR_BLOCK_COUNT	Y	Y	Y	Y					
ACMD41	SD_SEND_OP_COND	Y	Y	Y	Y					
ACMD42	SET_CLR_CARD_DETECT	Y	Y	Y	Y					
ACMD51	SEND_SCR	Y	Y	Y	Y	Y				

NOTE: For stream read/write, STOP_CMD is sent after finishing data transfer. For write, STOP_CMD is with the last six bytes. For read, STOP_CMD is sent after receiving data and card sends some data which MSC ignores.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33 OTG Controller

33.1 Overview

This chapter describes the USB On-The-Go (OTG) implemented in the processor.

The Universal Serial Bus (USB) supports serial data exchanges between a host computer and a variety of simultaneously accessible portable peripherals. Many of these portable devices would benefit a lot from being able to communicate to each other over the USB interface. And OTG make this possible. An OTG device can plays the role of both host and device.

Familiarity with the *Universal Serial Bus Specification*, Revision 1.1 and OTG supplement are necessary to fully understand the material contained in this section.

Features:

- Complies with the USB 2.0 standard for high-speed (480 Mbps) functions and with the *On-The-Go* supplement to the USB 2.0 specification
- Operates either as the function controller of a high- /full-speed USB peripheral or as the host/peripheral in point-to-point or multi-point communications with other USB functions
- Supports Session Request Protocol (SRP) and Host Negotiation Protocol (HNP)
- UTMI+ Level 3 Transceiver Interface
- Soft connect/disconnect
- 16 Endpoints:
- Supports control, interrupt, ISO and bulk transfer

33.2 Pin Description

Table 33-1 OTG Pins Description

Name	Type	Description
DP	Inout	Data Positive
DM	Inout	Data Minus
ID	Inout	Identification
DRVVBUS_PE10	Out	Charge pump enable

33.3 Control and Status Overview

Your application controls the DWC_otg core by reading from and writing to the Control and Status Registers (CSRs) through the AHB Slave interface. These registers are 32 bits wide, the addresses are 32- bit block aligned and the base address is 0x13500000.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Only the Core Global, Data FIFO Access, and Host Port registers can be accessed in both Host and Device modes. When the DWC_otg core is operating in one mode, either Device or Host, the application must not access registers from the other mode. If an illegal access occurs, a Mode Mismatch interrupt is generated and reflected in the Core Interrupt register (GINTSTS.ModeMis).
When the core switches from one mode to another, the registers in the new mode must be reprogrammed as they would be after a power-on reset.

33.4 CSR Memory Map

The CSR address map is fixed and does not depend on the core's configuration (for example, how many endpoints are implemented). Host and Device mode registers occupy different addresses. All registers are implemented in the AHB Clock domain.
Following figure shows the CSR address map.

Figure 33-1 OTG CSR Memory Map

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.4.1 Register Maps

The tables in this section provide high-level summaries of each register and register group.

Register Name	Name of register types and register names ordered by offset address
Acronym	Shorthand names for registers that are mapped to the offset address. These are used extensively in the programming examples in the Programming Guide. The first letter is a prefix for the register type: G: Core Global H: Host mode D: Device mode
Offset Address	Address, in hexadecimal (h), of the first byte of each register.

Note: FIFO size and FIFO depth are used interchangeably.

33.4.2 Global CSR Map

These registers are available in both Host and Device modes.

Table 33-2 Core Global CSR Map (000h-3FFh)

Acronym	Offset Address	Register Name
GOTGCTL	000h	"Control and Status Register (GOTGCTL)"
GOTGINT	004h	"Interrupt Register (GOTGINT)"
GAHBCFG	008h	"AHB Configuration Register (GAHBCFG)"
GUSBCFG	00Ch	"USB Configuration Register (GUSBCFG)"
GRSTCTL	010h	"Reset Register (GRSTCTL)"
GINTSTS	014h	"Interrupt Register (GINTSTS)"
GINTMSK	018h	"Interrupt Mask Register (GINTMSK)"
GRXSTSR	01Ch	"Receive Status Debug Read/Status Read and Pop Registers (GRXSTSR/GRXSTSP)
GRXSTSP	020h	
GRXFSIZ	024h	"Receive FIFO Size Register (GRXFSIZ)"
GNPTXFSIZ	028h	"Non-Periodic Transmit FIFO Size Register (GNPTXFSIZ)"
GNPTXSTS	02Ch	"Non-Periodic Transmit FIFO/Queue Status Register (GNPTXSTS)"
	030h - 038h	Reserved
GUID	03Ch	"User ID Register (GUID)"
GSNPSID	040h	"Synopsys ID Register (GSNPSID)"
GHWCFG1	044h	"User HW Config1 Register (GHWCFG1)"
GHWCFG2	048h	"User HW Config2 Register (GHWCFG2)"
GHWCFG3	04Ch	"User HW Config3 Register (GHWCFG3)"
GHWCFG4	050h	"User HW Config4 Register (GHWCFG4)"
GLPMCFG	054h	"Core LPM Configuration Register (GLPMCFG)"

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	0x58h	Reserved
GDFIFOCFG	05Ch	"DFIFO Software Config Register (GDFIFOCFG)"
GADPCTL	0x60h	"ADP Timer, Control and Status Register (GADPCTL)"
HPTXFSIZ	100h	"Host Periodic Transmit FIFO Size Register (HPTXFSIZ)"
DPTXFSIZn	104h-13Ch Dedicated FIFO	"Device Periodic Transmit FIFO-n Size Register (DPTXFSIZn)"
DIEPTXFn	104h-13Ch Dedicated FIFO	"Device IN Endpoint Transmit FIFO Size Register: (DIEPTXFn)"
	140h-3FFh	Reserved

33.4.3 Host Mode CSR Map

These registers must be programmed every time the core changes to Host mode.

Table 33-3 Host Mode CSR Map (400h-7FFh)

Acronym	Offset Address	Register Name
HCFG	400h	Host Configuration Register (HCFG)
HFIR	404h	Host Frame Interval Register (HFIR)
HFNUM	408h	Host Frame Number/Frame Time Remaining Register (HFNUM)
	40Ch	Reserved
HPTXSTS	410h	Host Periodic Transmit FIFO/Queue Status Register (HPTXSTS)
HAINT	414h	Host All Channels Interrupt Register (HAINT)
HAINTMSK	418h	Host All Channels Interrupt Mask Register (HAINTMSK)
HFLBAddr	41Ch	Host Frame List Base Address Register (HFLBAddr)
HPRT	440h	Host Port Control and Status Register (HPRT)
	444h-4FCh	Reserved
HCCHARn	500h - 6E0	Host Channel-n Characteristics Register (HCCHARn)
HCSPLTn	504h - 6E4	Host Channel-n Split Control Register (HCSPLTn)
HCINTn	508h - 6E8	Host Channel-n Interrupt Register (HCINTn)
HCINTMSKn	50Ch - 6EC	Host Channel-n Interrupt Mask Register (HCINTMSKn)
HCTSIZn	510h - 6F0	Host Channel-n Transfer Size Register (HCTSIZn)
	518h - 6F8	Reserved
HCDMABn	51Ch - 6FCh	Host Channel-n DMA Buffer Address Register (HCDMABn)
	6F0-7FFh	Reserved

33.4.4 Device Mode CSR Map

These registers must be programmed every time the core changes to Device mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-4 Device Mode CSR Map (800h-BFFh)

Acronym	Offset Address	Register Name
	800h-ACh	Device Logical IN Endpoint-Specific Registers
DCFG	800h	Device Configuration Register (DCFG)
DCTL	804h	Device Control Register (DCTL)
DSTS	808h	Device Status Register (DSTS)
	80Ch	Reserved
DIEPMSK	810h	Device IN Endpoint Common Interrupt Mask Register (DIEPMSK)
DOEPMSK	814h	Device OUT Endpoint Common Interrupt Mask Register (DOEPMSK)
DAINT	818h	Device All Endpoints Interrupt Register (DAINT)
DAINTMSK	81Ch	Device All Endpoints Interrupt Mask Register (DAINTMSK)
DTKNQR1	820h	Device IN Token Sequence Learning Queue Read Register 1 (DTKNQR1)
DTKNQR2	824h	Device IN Token Sequence Learning Queue Read Register 2 (DTKNQR2)
DTKNQR3	830h	Device IN Token Sequence Learning Queue Read Register 3 (DTKNQR3)
DTKNQR4	834h	Device IN Token Sequence Learning Queue Read Register 4 (DTKNQR4)
DVBUSDIS	828h	Device VBUS Discharge Time Register (DVBUSDIS)
DVBUSPULSE	82Ch	Device VBUS Pulsing Time Register (DVBUSPULSE)
DTHRCTL	830h	Device Threshold Control Register (DTHRCTL)
DIEPEMPMSK	834h	Device IN Endpoint FIFO Empty Interrupt Mask Register: (DIEPEMPMSK)
DEACHINT	838h	Device Each Endpoint Interrupt Register (DEACHINT)
DEACHINTMSK	83Ch	Device Each Endpoint Interrupt Register Mask (DEACHINTMSK)
DIEPEACHMSKn	840h	Device Each In Endpoint-n Interrupt Register (DIEPEACHMSKn)
DOEPEACHMSKn	880h	Device Each Out Endpoint-n Interrupt Register (DOEPEACHMSKn)
DIEPCTL0	900h	Device Control IN Endpoint 0 Control Register (DIEPCTL0)
	904h	Reserved
DIEPCTLn	920h-AE0h	Device Endpoint-n Control Register (DIEPCTLn/DOEPCTLn)
DIEPINTn		Device Endpoint-n Interrupt Register (DIEPINTn/DOEPINTn)
	90Ch	Reserved

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

DIEPTSIZ0	910h	Device Endpoint 0 Transfer Size Register (DIEPTSIZ0/DOEPTSIZ0)
DIEPTSIZn	910h	Device Endpoint-n Transfer Size Register (DIEPTSIZn/DOEPTSIZn)
DIEPDMAn	914h	Device Endpoint-n DMA Address Register (DIEPDMAn/DOEPDMAn)
DTXFSTSn	918h	Device IN Endpoint Transmit FIFO Status Register (DTXFSTSn)
DIEPDMAB0/DIEPDMABn	91Ch	Device Endpoint-n DMA Buffer Address Register (DIEPDMABn/DOEPDMABn)
DOEPCTL0	B00h	Device Control OUT Endpoint 0 Control Register (DOEPCTL0)
	B04h	Reserved
DOEPCTLn	B20h-CE0h	Device Endpoint-n Control Register (DIEPCTLn/DOEPCTLn)
DOEPINTn	B08h	Device Endpoint-n Interrupt Register (DIEPINTn/DOEPINTn)
	B0Ch	Reserved
DOEPTSIZ0	B10h	Device Endpoint 0 Transfer Size Register (DIEPTSIZ0/DOEPTSIZ0)
DOEPTSIZn	B30h-DB0h	Device Endpoint-n Transfer Size Register (DIEPTSIZn/DOEPTSIZn)
DOEPDMAn	B14h-CF4h	Device Endpoint-n DMA Address Register (DIEPDMAn/DOEPDMAn)
DOEPDMAB0/ DOEPDMABn	B1Ch-CFCh	Device Endpoint-n DMA Buffer Address Register (DIEPDMABn/DOEPDMABn)

33.4.5 Data FIFO (DFIFO) Access Register Map

These registers, available in both Host and Device modes, are used to read or write the FIFO space for a specific endpoint or a channel, in a given direction. If a host channel is of type IN, the FIFO can only be read on the channel. Similarly, if a host channel is of type OUT, the FIFO can only be written on the channel.

Table 33-5 Data FIFO (DFIFO) Access Register Map

FIFO Access Register Section	Address Range	Access
Device IN Endpoint 0/Host OUT Channel 0: DFIFO Write Access Device OUT Endpoint 0/Host IN Channel 0: DFIFO Read Access	1000h-1FFCh	WO/RO
Device IN Endpoint 1/Host OUT Channel 1: DFIFO Write Access Device OUT Endpoint 1/Host IN Channel 1: DFIFO Read Access	2000h-2FFCh	WO/RO
...	...	...

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Device IN Endpoint 14/Host OUT Channel 14: DFIFO Write Access Device OUT Endpoint 14/Host IN Channel 14: DFIFO Read Access	F000h-FFFCh	WO/RO
Device IN Endpoint 15/Host OUT Channel 15: DFIFO Write Access Device OUT Endpoint 15/Host IN Channel 15: DFIFO Read Access	10000h-10FFCh	WO/RO

33.4.6 Interrupt Hierarchy

Following figure displays the DWC_otg interrupt hierarchy.

Figure 33-2 Interrupt Hierarchy

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.4.6.1 The Core Interrupt Handler

Following figure illustrates how the core interrupt handler works.

Figure 33-3 Core Interrupt Handler

33.5 Register Descriptions

This section describes Core Global, Device Mode, Host Mode CSRs. Detailed register programming examples are provided in the Programming Guide.

33.5.1 Application Access to the CSRs

The Access column of each register description that follows specifies how the application and the core can access the register fields of the CSRs. The following conventions are used:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Read Only (RO)	Register field can only be read by the application. Writes to read-only fields have no effect.
Write Only (WO)	Register field can only be written by the application.
Read and Write (R_W)	Register field can be read and written by the application. The application can set this field by writing 1'b1 and can clear it by writing 1'b0.
Read, Write, and Self Clear (R_W_SC)	Register field can be read and written by the application (Read and Write), and is cleared to 1'b0 by the core (Self Clear). The conditions under which the core clears this field are explained in detail in the field's description.
Read, Write, Self Set, and Self Clear (R_W_SS_SC)	Register field can be read and written by the application (Read and Write), set to 1'b1 by the core on certain USB events (Self Set), and cleared to 1'b0 by the core (Self Clear). The conditions under which the core sets and clears this field are explained in the field's description. (Only the Port Resume bit of the Host Port Control and Status register, HPRT.PrtRes, uses this access type).
Read, Self Set, and Write Clear (R_SS_WC)	Register field can be read by the application (Read), can be set to 1'b1 by the core on a certain internal or USB or AHB event (Self Set), and can be cleared to 1'b0 by the application with a register write of 1'b1 (Write Clear). A register write of 1'b0 has no effect on this field. The conditions under which the core sets this field are explained in detail in the field's description. (For example, interrupt bits.)
Read, Write Set, and Self Clear (R_WS_SC)	Register field can be read by the application (Read), can be set to 1'b1 by the application with a register write of 1'b1 (Write Set), and is cleared to 1'b0 by the core (Self Clear). The application cannot clear this type of field, and a register write of 1'b0 to this bit has no effect on this field. The conditions under which the core clears this field are explained in detail in the field's description. (For example, reset signals)
Read, Self Set, and Self Clear or Write Clear (R_SS_SC_WC)	Register field can be read by the application (Read), can be set to 1'b1 by the core on certain internal or USB or AHB events (Self Set), and can be cleared to 1'b0 either by the core itself (Self Clear) or by the application with a register write of 1'b1 (Write Clear). A register write of 1'b0 to this bit has to no effect on this field. The conditions under which the core sets or clears this field are explained in the field's description. (Only the Port Enable bit of the Host Port Control and Status register, HPRT.PrtEna, and the VStatus Done bit of the PHY Vendor Control register, GPVNDCTL.VStsDone, use this access type.)

Note: Always program Reserved fields with 0s. Treat read values from Reserved fields as unknowns(Xs).

33.5.2 Overview of Commonly Used Register Bits

This section provides an overview of the commonly used registers and bits. For a complete description of all the registers, see the corresponding sections.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-6 List of Commonly Used Register Bits

Bit Number	Register/Bit Name	Description
	"Control and Status Register (GOTGCTL)"	The OTG Control and Status register controls the behavior and reflects the status of the OTG function of the core.
11	Device HNP Enabled (DevHNPEn)	The application sets this bit when it successfully receives a SetFeature.SetHNPEnable command from the connected USB host. ▪ 1'b0: HNP is not enabled in the application ▪ 1'b1: HNP is enabled in the application
10	Host Set HNP Enable (HstSetHNPEn)	The application sets this bit when it has successfully enabled HNP (using the SetFeature.SetHNPEnable command) on the connected device. ▪ 1'b0: Host Set HNP is not enabled ▪ 1'b1: Host Set HNP is enabled
	"AHB Configuration Register (GAHBCFG)"	This register can be used to configure the core after power-on or a change in mode. This register mainly contains AHB system-related configuration parameters. Do not change this register after the initial programming. The application must program this register before starting any transactions on either the AHB or the USB.
5	DMA Enable (DMAEn)	▪ 1'b0: Core operates in Slave mode ▪ 1'b1: Core operates in a DMA mode
	"USB Configuration Register (GUSBCFG)"	This register can be used to configure the core after power-on or a changing to Host mode or Device mode. It contains USB and USB-PHY related configuration parameters. The application must program this register before starting any transactions on either the AHB or the USB. Do not make changes to this register after the initial programming.
30	Force Device Mode (ForceDevMode)	Writing a 1 to this bit forces the core to device mode irrespective of utmiotg_iddig input pin. ▪ 1'b0: Normal Mode ▪ 1'b1: Force Device Mode A fter setting the force bit, the application must wait at least 25 ms before the change to take effect. When the simulation is in scale down mode, waiting for 500 µs is sufficient.
29	Force Host Mode (ForceHstMode)	Writing a 1 to this bit forces the core to host mode irrespective of utmiotg_iddig input pin. ▪ 1'b0: Normal Mode ▪ 1'b1: Force Host Mode After setting the force bit, the application must wait at least 25 ms

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		before the change to take effect. When the simulation is in scale down mode, waiting for 500 µs is sufficient.
9	HNP-Capable (HNPCap)	The application uses this bit to control the DWC_otg core's HNP capabilities. ▪ 1'b0: HNP capability is not enabled. ▪ 1'b1: HNP capability is enabled.
8	SRP-Capable (SRPCap)	The application uses this bit to control the DWC_otg core SRP capabilities. If the core operates as a non-SRP-capable B-device, it cannot request the connected A-device (host) to activate VBUS and start a session. ▪ 1'b0: SRP capability is not enabled. ▪ 1'b1: SRP capability is enabled.
6	USB 2.0 High-Speed PHY or USB 1.1 Full-Speed Serial Transceiver Select	The application uses this bit to select either a high-speed UTMI+ or ULPI PHY, or a full-speed transceiver. ▪ 1'b0: USB 2.0 high-speed UTMI+ or ULPI PHY ▪ 1'b1: USB 1.1 full-speed serial transceiver This bit is always 0, with Read Only access.
4	ULPI or UTMI+ Select (ULPI_UTMI_Sel)	The application uses this bit to select either a UTMI+ interface or ULPI Interface. ▪ 1'b0: UTMI+ Interface ▪ 1'b1: ULPI Interface This bit is always 0, with Read Only access.
3	PHY Interface (PHYIf)	The application uses this bit to configure the core to support a UTMI+ PHY with an 8- or 16-bit interface. 1'b0: 8 bits 1'b1: 16 bits
	"Host Configuration Register (HCFG)"	This register configures the core after power-on. Do not make changes to this register after initializing the host.
23	Enable Scatter/gather DMA in Host mode (DescDMA)	When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. NOTE: This bit must be modified only once after a reset. The following combinations are available for programming: ▪ GAHBCFG.DMAEn=0, HCFG.DescDMA=0 => Slave mode ▪ GAHBCFG.DMAEn=0, HCFG.DescDMA=1 => Invalid ▪ GAHBCFG.DMAEn=1, HCFG.DescDMA=0 => Buffered DMA mode ▪ GAHBCFG.DMAEn=1, HCFG.DescDMA=1 => Scatter/Gather DMA mode In non Scatter/Gather DMA mode, this bit is reserved.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

2	FS- and LS-Only Support(FSLSSupp)	The application uses this bit to control the core's enumeration speed. Using this bit, the application can make the core enumerate as a FS host, even if the connected device supports HS traffic. Do not make changes to this field after initial programming. ▪ 1'b0: HS/FS/LS, based on the maximum speed supported by the connected device ▪ 1'b1: FS/LS-only, even if the connected device can support HS
	"Host Channel-n Characteristics Register (HCCHARn)"	
19:18	Endpoint Type (EPType)	Indicates the transfer type selected. ▪ 2'b00: Control ▪ 2'b01: Isochronous ▪ 2'b10: Bulk ▪ 2'b11: Interrupt
15	Endpoint Direction (EPDir)	Indicates whether the transaction is IN or OUT. ▪ 1'b0: OUT ▪ 1'b1: IN
14:11	Endpoint Number (EPNum)	Indicates the endpoint number on the device serving as the data source or sink.
10:0	Maximum Packet Size (MPS)	Indicates the maximum packet size of the associated endpoint.
	"Host Channel-n Transfer Size Register (HCTSIZn)"	
30:29	PID (Pid)	The application programs this field with the type of PID to use for the initial transaction. The host maintains this field for the rest of the transfer. ▪ 2'b00: DATA0 ▪ 2'b01: DATA2 ▪ 2'b10: DATA1 ▪ 2'b11: MDATA (non-control)
	"Device Configuration Register (DCFG)"	This register configures the core in Device mode after power-on or after certain control commands or enumeration. Do not make changes to this register after initial programming.
25:24	Periodic Scheduling Interval (PerSchIntvl)	PerSchIntvl must be programmed only for Scatter/Gather DMA mode. Description: This field specifies the amount of time the Internal DMA engine must allocate for fetching periodic IN endpoint data. Based on the number of periodic endpoints, this value must be specified as 25,50 or 75% of (micro)frame. ▪ When any periodic endpoints are active, the internal DMA

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		engine allocates the specified amount of time in fetching periodic IN endpoint data. ▪ When no periodic endpoints are active, then the internal DMA engine services non-periodic endpoints, ignoring this field. ▪ After the specified time within a (micro)frame, the DMA switches to fetching for non-periodic endpoints. ▪ 2'b00: 25% of (micro)frame. ▪ 2'b01: 50% of (micro)frame. ▪ 2'b10: 75% of (micro)frame. ▪ 2'b11: Reserved.
23	Enable Scatter/Gather DMA in Device mode (DescDMA)	When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. NOTE: This bit must be modified only once after a reset. The following combinations are available for programming: ▪ GAHBCFG.DMAEn=0,DCFG.DescDMA=0 => Slave mode ▪ GAHBCFG.DMAEn=0,DCFG.DescDMA=1 => Invalid ▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=0 => Buffered DMA mode ▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=1 => Scatter/Gather DMA mode
10:4	Device Address (DevAddr)	The application must program this field after every SetAddress control command.
1:0	Device Speed (DevSpd)	Indicates the speed at which the application requires the core to enumerate, or the maximum speed the application can support. However, the actual bus speed is determined only after the chirp sequence is completed, and is based on the speed of the USB host to which the core is connected. See "Device Initialization" in the Programming Guide for details. ▪ 2'b00: High speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) ▪ 2'b01: Full speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) ▪ 2'b10: Reserved ▪ 2'b11: Full speed (USB 1.1 transceiver clock is 48 MHz)
	"Device Endpoint-n Control Register (DIEPCTLn/DOEPCTLn)"	The application uses this register to control the behavior of each logical endpoint other than endpoint 0.
29	Set DATA1 PID (SetD1PID)	Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA1. This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode.
28	Set DATA0 PID (SetD0PID)	Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		register to DATA0. This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode.
19:18	Endpoint Type (EPType)	Applies to IN and OUT endpoints. This is the transfer type supported by this logical endpoint. ▪ 2'b00: Control ▪ 2'b01: Isochronous ▪ 2'b10: Bulk ▪ 2'b11: Interrupt
10:00	Maximum Packet Size (MPS)	Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint. This value is in bytes.

33.5.3 Global Registers

These registers are available in both Host and Device modes, and do not need to be reprogrammed when switching between these modes.

33.5.3.1 Control and Status Register (GOTGCTL)

✧ Offset: 000h

The OTG Control and Status register controls the behavior and reflects the status of the OTG function of the core.

Table 33-7 Control and Status Register: GOTGCTL

Field	Description	Mode	Reset	Access
31:26	Reserved	Host and Device		RO
27	Chirp On Enable (ChirpEn) This bit when programmed to 1'b1 results in the core asserting chirp_on before sending an actual Chirp "K" signal on USB. This bit is present only if OTG_BC_SUPPORT = 1.If OTG_BC_SUPPORT!=1, this bit is a reserved bit.	Device only	1'b0	RW
26:22	Multi Valued ID pin (MultValIdBc) Battery Charger ACA inputs in the following order: ▪ Bit 26 - rid_float. ▪ Bit 25 - rid_gnd ▪ Bit 24 - rid_a ▪ Bit 23 - rid_b ▪ Bit 22 - rid_c These bits are valid only if OTG_BC_SUPPORT=1, otherwise they are reserved.	Host and Device	Configurable. The reset value is unknown if OTG_BC_SUPPORT =1 and 0 otherwise.	RO
21	Reserved	Host and		RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Device		
20	OTG Version (OTGVer) Indicates the OTG revision. ▪ 1'b0: OTG Version 1.3. In this version the core supports Data line pulsing and VBus pulsing for SRP. ▪ 1'b1: OTG Version 2.0. In this version the core supports only Data line pulsing for SRP.		1'b0	RW
19	B-Session Valid (BSesVld) Indicates the Device mode transceiver status. ▪ 1'b0: B-session is not valid. ▪ 1'b1: B-session is valid. In OTG mode, you can use this bit to determine if the device is connected or disconnected. **Note:** If you do not enable OTG features (such as SRP and HNP), the read reset value will be 1.The vbus assigns the values internally for non- SRP or non-HNP configurations. In case of OTG_MODE=0, the reset value of this bit is 1'b0.	Device only	Configurable	RO
18	A-Session Valid (ASesVld) Indicates the Host mode transceiver status. ▪ 1'b0: A-session is not valid ▪ 1'b1: A-session is valid **Note:** If you do not enable OTG features (such as SRP and HNP), the read reset value will be 1.The vbus assigns the values internally for non- SRP or non-HNP configurations. In device mode, this bit is reserved.	Host only	Configurable	RO
17	Long/Short Debounce Time (DbncTime) Indicates the debounce time of a detected connection. ▪ 1'b0: Long debounce time, used for physical connections (100 ms + 2.5 μs) ▪ 1'b1: Short debounce time, used for soft connections (2.5 μs)	Host only	1'b0	RO
16	Connector ID Status (ConIDSts) Indicates the connector ID status on a connect event. ▪ 1'b0: The DWC_otg core is in A-Device mode ▪ 1'b1: The DWC_otg core is in B-Device mode	Host and Device	1'b1	RO
15:12	Reserved	Host and Device		RO
11	Device HNP Enabled (DevHNPEn) The application sets this bit when it successfully receives	Device Only	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	a SetFeature.SetHNPEnable command from the connected USB host. ▪ 1'b0: HNP is not enabled in the application ▪ 1'b1: HNP is enabled in the application	OTG configurations		
10	Host Set HNP Enable (HstSetHNPEn) The application sets this bit when it has successfully enabled HNP (using the SetFeature.SetHNPEnable command) on the connected device. ▪ 1'b0: Host Set HNP is not enabled ▪ 1'b1: Host Set HNP is enabled	Device Only OTG configurations	1'b0	R_W
9	HNP Request (HNPReq) The application sets this bit to initiate an HNP request to the connected USB host. The application can clear this bit by writing a 0 when the Host Negotiation Success Status Change bit in the OTG Interrupt register (GOTGINT.HstNegSucStsChng) is set. The core clears this bit when the HstNegSucStsChng bit is cleared. ▪ 1'b0: No HNP request ▪ 1'b1: HNP request	Device Only OTG configurations	1'b0	R_W
8	Host Negotiation Success (HstNegScs) The core sets this bit when host negotiation is successful. The core clears this bit when the HNP Request (HNPReq) bit in this register is set. ▪ 1'b0: Host negotiation failure ▪ 1'b1: Host negotiation success	Device only	1'b0	RO
7	B-Peripheral Session Valid OverrideValue (BvalidOvVal) This bit is used to set the Override value for Bvalid signal when GOTGCTL.BvalidOvEn is set. ▪ 1'b0: Bvalid value is 1'b0 when GOTGCTL.BvalidOvEn = 1. ▪ 1'b1: Bvalid value is 1'b1 when GOTGCTL.BvalidOvEn = 1.	Device only	1'b0	R_W
6	B-Peripheral Session Valid Override Enable (BvalidOvEn) This bit is used to enable/disable the software to override the Bvalid signal using the GOTGCTL.BvalidOvVal. ▪ 1'b1: Internally Bvalid received from the PHY is overridden with GOTGCTL.BvalidOvVal. ▪ 1'b0: Override is disabled and Bvalid signal from the respective PHY selected is used internally by the core.	Device only	1'b0	R_W
5	A-Peripheral Session Valid OverrideValue (AvalidOvVal) This bit is used to set the Override value for Avalid signal when GOTGCTL.AvalidOvEn is set.	Host only	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b0: Avalid value is 1'b0 when GOTGCTL.AvalidOvEn = 1. ▪ 1'b1: Avalid value is 1'b1 when GOTGCTL.AvalidOvEn = 1.			
4	A-Peripheral Session Valid Override Enable (AvalidOvEn) This bit is used to enable/disable the software to override the Avalid signal using the GOTGCTL.AvalidOvVal. ▪ 1'b1: Internally Avalid received from the PHY is overridden with GOTGCTL.AvalidOvVal. ▪ 1'b0: Override is disabled and Avalid signal from the respective PHY is used internally by the core.	Host only	1'b0	R_W
3	VBUS Valid OverrideValue (VbvalidOvVal) This bit is used to set the Override value for vbus valid signal when GOTGCTL.VbusvalidOvEn is set. ▪ 1'b0: vbusvalid value is 1'b0 when GOTGCTL.VbvalidOvEn = 1. ▪ 1'b1: vbusvalid value is 1'b1 when GOTGCTL.VbvalidOvEn = 1.	Host only	1'b0	R_W
2	VBUS Valid Override Enable (VbvalidOvEn) This bit is used to enable/disable the software to override the vbus-valid signal using the GOTGCTL.vbvalidOvVal. ▪ 1'b1: The vbus-valid signal received from the PHY is overridden with GOTGCTL.vbvalidOvVal. ▪ 1'b0: Override is disabled and avalid signal from the respective PHY is used internally by the core.	Host only	1'b0	R_W
1	Session Request (SesReq) The application sets this bit to initiate a session request on the USB. The application can clear this bit by writing a 0 when the Host Negotiation Success Status Change bit in the OTG Interrupt register (GOTGINT.HstNegSucStsChng) is set. The core clears this bit when the HstNegSucStsChng bit is cleared. If you use the USB 1.1 Full-Speed Serial Transceiver interface to initiate the session request, the application must wait until the VBUS discharges to 0.2 V, after the B-Session Valid bit in this register (GOTGCTL.BSesVld) is cleared. This discharge time varies between different PHYs and can be obtained from the PHY vendor. ▪ 1'b0: No session request ▪ 1'b1: Session request	Device only	1'b0	R_W
0	Session Request Success (SesReqScs) The core sets this bit when a session request initiation is successful.	Device only	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b0: Session request failure ▪ 1'b1: Session request success			

33.5.3.2 Interrupt Register (GOTGINT)

✧ Offset: 004h

The application reads this register whenever there is an OTG interrupt and clears the bits in this register to clear the OTG interrupt. It is shown in Figure 33-2.

Table 33-8 Interrupt Register: GOTGINT

Field	Description	Mode	Reset	Access
31:20	Reserved	Host and Device		RO
20	Multi-Valued input changed This bit when set indicates that there is a change in the value of at least one ACA pin value. This bit is present only if OTG_BC_SUPPORT = 1, otherwise it is reserved.		1'b0	R_SS_WC
19	Debounce Done (DbnceDone) The core sets this bit when the debounce is completed after the device connect. The application can start driving USB reset after seeing this interrupt. This bit is only valid when the HNP Capable or SRP Capable bit is set in the Core USB Configuration register (GUSBCFG.HNPCap or GUSBCFG.SRPCap, respectively).	Host Only	1'b0	R_SS_WC
18	A-Device Timeout Change (ADevTOUTChg) The core sets this bit to indicate that the A-device has timed out while waiting for the B-device to connect.	Host and Device	1'b0	R_SS_WC
17	Host Negotiation Detected (HstNegDet) The core sets this bit when it detects a host negotiation request on the USB.	Host and Device	1'b0	R_SS_WC
16：10	Reserved	Host and Device		RO
9	Host Negotiation Success Status Change (HstNegSucStsChng) The core sets this bit on the success or failure of a USB host negotiation request. The application must read the Host Negotiation Success bit of the OTG Control and Status register (GOTGCTL.HstNegScs) to check for success or failure.	Host and Device	1'b0	R_SS_WC
8	Session Request Success Status Change (SesReqSucStsChng)	Host and Device	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	The core sets this bit on the success or failure of a session request. The application must read the Session Request Success bit in the OTG Control and Status register (GOTGCTL.SesReqScs) to check for success or failure.			
7:3	Reserved	Host and Device		RO
2	Session End Detected (SesEndDet) The core sets this bit when the utmisrp_bvalid signal is deasserted.	Device only	1'b0	R_SS_WC
1:0	Reserved	Host and Device		RO

33.5.3.3 AHB Configuration Register (GAHBCFG)

✧ Offset: 008h

This register can be used to configure the core after power-on or a change in mode. This register mainly contains AHB system-related configuration parameters. Do not change this register after the initial programming. The application must program this register before starting any transactions on either the AHB or the USB.

Table 33-9 AHB Configuration Register: GAHBCFG

Field	Description	Mode	Reset	Access
31:23	Reserved	Host and Device		RO
23	AHB Single Support (AHBSingle) This bit when programmed supports Single transfers for the remaining data in a transfer when the DWC_otg core is operating in DMA mode. ▪ 1'b0: This is the default mode. When this bit is set to 1'b0, the remaining data in the transfer is sent using INCR burst size. ▪ 1'b1: When set to 1'b1, the remaining data in a transfer is sent using Single burst size. Note: if this feature is enabled, the AHB RETRY and SPLIT transfers still have INCR burst type. Enable this feature when the AHB Slave connected to the DWC_otg core does not support INCR burst (and when Split, and Retry transactions are not being used in the bus).	Host and Device	1'b0	R_W
22	Notify All Dma Write Transactions (NotiAllDmaWrit) This bit is programmed to enable the System DMA Done functionality for all the DMA write Transactions corresponding to the Channel/Endpoint. This bit is valid	Host and Device	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	only when GAHBCFG.RemMemSupp is set to 1. ▪ GAHBCFG.NotiAllDmaWrit = 1 – HS OTG core asserts int_dma_req for all the DMA write transactions on the AHB interface along with int_dma_done, chep_last_transact and chep_number signal informations. The core waits for sys_dma_done signal for all the DMA write transactions in order to complete the transfer of a particular Channel/Endpoint. ▪ GAHBCFG.NotiAllDmaWrit = 0 – HS OTG core asserts int_dma_req signal only for the last transaction of DMA write transfer corresponding to a particular Channel/Endpoint. Similarly, the core waits for sys_dma_done signal only for that transaction of DMA write to complete the transfer of a particular Channel/Endpoint.			
21	Remote Memory Support (RemMemSupp) This bit is programmed to enable the functionality to wait for the system DMA Done Signal for the DMA Write Transfers. ▪ GAHBCFG.RemMemSupp=1 – The int_dma_req output signal is asserted when HS OTG DMA starts write transfer to the external memory. When the core is done with the Transfers it asserts int_dma_done signal to flag the completion of DMA writes from HS OTG. The core then waits for sys_dma_done signal from the system to proceed further and complete the Data Transfer corresponding to a particular Channel/Endpoint. ▪ GAHBCFG.RemMemSupp=0 – The int_dma_req and int_dma_done signals are not asserted and the core proceeds with the assertion of the XferComp interrupt as soon as the DMA write transfer is done at the HS OTG Core Boundary and it doesn't wait for the sys_dma_done signal to complete the DATA transfers.	Host and Device	1'b0	R_W
20:9	Reserved	Host and Device		RO
8	Periodic TxFIFO Empty Level (PTxFEmpLvl) Indicates when the Periodic TxFIFO Empty Interrupt bit in	Host only	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	the Core Interrupt register (GINTSTS.PTxFEmp) is triggered. This bit is used only in Slave mode. ▪ 1'b0: GINTSTS.PTxFEmp interrupt indicates that the Periodic TxFIFO is half empty ▪ 1'b1: GINTSTS.PTxFEmp interrupt indicates that the Periodic TxFIFO is completely empty			
7	Non-Periodic TxFIFO Empty Level (NPTxFEmpLvl) This bit is used only in Slave mode. In host mode and with Shared FIFO with device mode, this bit indicates when the Non-Periodic TxFIFO Empty Interrupt bit in the Core Interrupt register (GINTSTS.NPTxFEmp) is triggered. With dedicated FIFO in device mode, this bit indicates when IN endpoint Transmit FIFO empty interrupt (DIEPINTn.TxFEmp) is triggered. Host mode and with Shared FIFO with device mode: ▪ 1'b0: GINTSTS.NPTxFEmp interrupt indicates that the Non-Periodic TxFIFO is half empty ▪ 1'b1: GINTSTS.NPTxFEmp interrupt indicates that the Non-Periodic TxFIFO is completely empty Dedicated FIFO in device mode: ▪ 1'b0: DIEPINTn.TxFEmp interrupt indicates that the IN Endpoint TxFIFO is half empty ▪ 1'b1: DIEPINTn.TxFEmp interrupt indicates that the IN Endpoint TxFIFO is completely empty	Host and Device	1'b0	R_W
6	Reserved	Host and Device		RO
5	DMA Enable (DMAEn) ▪ 1'b0: Core operates in Slave mode ▪ 1'b1: Core operates in a DMA mode	Host and Device	1'b0	R_W
4:1	Burst Length/Type (HBstLen) This field is used in both External and Internal DMA modes. In External DMA mode, these bits appear on dma_burst[3:0] ports, which can be used by an external wrapper to interface the External DMA Controller interface to Synopsys DW_ahb_dmac or ARM PrimeCell. External DMA Mode — defines the DMA burst length in terms of 32-bit words: ▪ 4'b0000: 1 word ▪ 4'b0001: 4 words ▪ 4'b0010: 8 words ▪ 4'b0011: 16 words ▪ 4'b0100: 32 words	Host and Device	4'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 4'b0101: 64 words ▪ 4'b0110: 128 words ▪ 4'b0111: 256 words ▪ Others: Reserved Internal DMA Mode — AHB Master burst type: ▪ 4'b0000 Single ▪ 4'b0001 INCR ▪ 4'b0011 INCR4 ▪ 4'b0101 INCR8 ▪ 4'b0111 INCR16 ▪ Others: Reserved Slave Mode: Only single burst is supported.			
0	Global Interrupt Mask (GlblIntrMsk) The application uses this bit to mask or unmask the interrupt line assertion to itself. Irrespective of this bit's setting, the interrupt status registers are updated by the core. ▪ 1'b0: Mask the interrupt assertion to the application. ▪ 1'b1: Unmask the interrupt assertion to the application.	Host and Device	1'b0	R_W

33.5.3.4 USB Configuration Register (GUSBCFG)

✧ Offset: 00Ch

This register can be used to configure the core after power-on or a changing to Host mode or Device mode.

It contains USB and USB-PHY related configuration parameters. The application must program this register before starting any transactions on either the AHB or the USB. Do not make changes to this register after the initial programming.

Table 33-10 USB Configuration Register: GUSBCFG

Field	Description	Mode	Reset	Access
31	Corrupt Tx packet This bit is for debug purposes only. Never set this bit to 1.	Host and Device	1'b0	WO
30	Force Device Mode (ForceDevMode) Writing a 1 to this bit forces the core to device mode irrespective of utmiotg_iddig input pin. ▪ 1'b0: Normal Mode ▪ 1'b1: Force Device Mode After setting the force bit, the application must wait at least 25 ms before the change to take effect. When the simulation is in scale down mode, waiting for 500 μs is	Host and Device	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	sufficient.			
29	Force Host Mode (ForceHstMode) Writing a 1 to this bit forces the core to host mode irrespective of utmiotg_iddig input pin. ▪ 1'b0: Normal Mode ▪ 1'b1: Force Host Mode After setting the force bit, the application must wait at least 25 ms before the change to take effect. When the simulation is in scale down mode, waiting for 500 µs is sufficient.	Host and Device	1'b0	R_W
28	Tx End Delay (TxEndDelay) Writing a 1 to this bit enables the TxEndDelay timers in the core as per the section 4.1.5 on Opmode of the USB 2.0 Transceiver Macrocell Interface (UTMI) version 1.05. ▪ 1'b0: Normal mode ▪ 1'b1: Introduce Tx end delay timers	Device Only	1'b0	R_W
27	IC_USB TrafficPullRemove Control (IC_USBTrafCtl) When this bit is set, pullup/pulldown resistors are detached from the USB during traffic signaling, per section 6.3.4 of the IC_USB specification. This bit is invalid.	Device	1'h0	R_W
26	IC_USB-Capable (IC_USBCap) The application uses this bit to control the DWC_otg core's IC_USB capabilities. ▪ 1'b0: IC_USB PHY Interface is not selected. ▪ 1'b1: IC_USB PHY Interface is selected. This bit is set to 1'b0 and the bit is read only.	Host and Device	Read description	RO/R_W
25	ULPI Interface Protect Disable Controls circuitry built into the PHY for protecting the ULPI interface when the link tri-states STP and data. Any pull-ups or pull-downs employed by this feature can be disabled. ▪ 1'b0: Enables the interface protect circuit ▪ 1'b1: Disables the interface protect circuit This bit is reserved and read-only.	Host only	1'b0	R_W/RO
24	Indicator Pass Through Controls wether the Complement Output is qualified with the Internal Vbus Valid comparator before being used in the Vbus State in the RX CMD. Please refer to the ULPI Specification for more detail.	Host Only	1'b0	R_W/RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b0: Complement Output signal is qualified with the Internal VbusValid comparator. ▪ 1'b1: Complement Output signal is not qualified with the Internal VbusValid comparator. This bit is reserved and read-only.			
23	Indicator Complement Controls the PHY to invert the ExternalVbusIndicator input signal, generating the Complement Output. Please refer to the ULPI Specification for more detail ▪ 1'b0: PHY does not invert ExternalVbusIndicator signal ▪ 1'b1: PHY does invert ExternalVbusIndicator signal This bit is reserved and read-only.	Host Only	1'b0	R_W/RO
22	TermSel DLine Pulsing Selection (TermSelDLPulse) This bit selects utmi_termselect to drive data line pulse during SRP. ▪ 1'b0: Data line pulsing using utmi_txvalid (default). ▪ 1'b1: Data line pulsing using utmi_termsel.	Device Only	1'b0	R_W
21	ULPI External VBUS Indicator (ULPIExtVbusIndicator) This bit indicates to the ULPI PHY to use an external VBUS over-current indicator. ▪ 1'b0: PHY uses internal VBUS valid comparator. ▪ 1'b1: PHY uses external VBUS valid comparator. This bit is invalid.	Host Only	1'b0	R_W
20	ULPI External VBUS Drive (ULPIExtVbusDrv) This bit selects between internal or external supply to drive 5V on VBUS, in ULPI PHY. ▪ 1'b0: PHY drives VBUS using internal charge pump (default). ▪ 1'b1: PHY drives VBUS using external supply. This bit is invalid.	Host Only	1'b0	R_W
19	ULPI Clock SuspendM (ULPIClkSusM) This bit sets the ClockSuspendM bit in the Interface Control register on ULPI PHY. 1'b0: PHY powers down internal clock during suspend. 1'b1: PHY does not power down internal clock. This bit is invalid.	Host and Device	1'b0	R_W
18	ULPI Auto Resume (ULPIAutoRes) This bit sets the AutoResume bit in the Interface Control register on the ULPIPHY. ▪ 1'b0: PHY does not use AutoResume feature. ▪ 1'b1: PHY uses AutoResume feature.	Host and Device	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This bit is invalid.			
17	ULPI FS/LS Select (ULPIFsLs) The application uses this bit to select the FS/LS serial interface for the ULPI PHY. This bit is valid only when the FS serial transceiver is selected on the ULPI PHY. ▪ 1'b0: ULPI interface ▪ 1'b1: ULPI FS/LS serial interface	Host and Device	1'b0	R_W
16	UTMIFS or I^2C Interface Select (OtgI2CSel) The application uses this bit to select the I^2C interface. ▪ 1'b0: UTMI USB 1.1 Full-Speed interface for OTG signals ▪ 1'b1: I 2C interface for OTG signals This bit is writable only if I ^{2}C and UTMIFS were specified for Enable I2C Interface	Host and Device	1'b0	RO/R_W
15	PHY Low-Power Clock Select (PhyLPwrClkSel) Selects either 480-MHz or 48-MHz (low-power) PHY mode. In FS and LS modes, the PHY can usually operate on a 48-MHz clock to save power. ▪ 1'b0: 480-MHz Internal PLL clock ▪ 1'b1: 48-MHz External Clock In 480 MHz mode, the UTMI interface operates at either 60 or 30-MHz, depending upon whether 8- or 16-bit data width is selected. In 48-MHz mode, the UTMI interface operates at 48 MHz in FS and LS modes. This bit drives the utmi_fsls_low_power core output signal, and is valid only for UTMI+ PHYs.	Host and Device	1'b0	R_W
14	Reserved	Host and Device		RO
13:10	USB Turnaround Time (USBTrdTim) Sets the turnaround time in PHY clocks. Specifies the response time for a MAC request to the Packet FIFO Controller (PFC) to fetch data from the DFIFO (SPRAM). This must be programmed to ▪ 4'h5: When the MAC interface is 16-bit UTMI+. ▪ 4'h9: When the MAC interface is 8-bit UTMI+. **Note:** The values above are calculated for the minimum AHB frequency of 30 MHz. USB turnaround time is critical for certification where long cables and 5-Hubs are used, so if you need the AHB to run at less than 30 MHz, and if USB turnaround time is not critical, these bits can be programmed to a larger value.	Device Only	4'h5	R_W
9	HNP-Capable (HNPCap)	Host and	1'h0	RO/R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	The application uses this bit to control the DWC_otg core's HNP capabilities. ▪ 1'b0: HNP capability is not enabled. ▪ 1'b1: HNP capability is enabled. If HNP functionality is disabled by the software, the OTG signals on the PHY domain must be tied to the appropriate values.	Device		
8	SRP-Capable (SRPCap) The application uses this bit to control the DWC_otg core SRP capabilities. If the core operates as a non-SRP-capable B-device, it cannot request the connected A-device (host) to activate VBUS and start a session. ▪ 1'b0: SRP capability is not enabled. ▪ 1'b1: SRP capability is enabled. If SRP functionality is disabled by the software, the OTG signals on the PHY domain must be tied to the appropriate values.	Host and Device	1'h0	R_W
7	ULPI DDR Select (DDRSel) The application uses this bit to select a Single Data Rate (SDR) or Double Data Rate (DDR) or ULPI interface. ▪ 1'b0: Single Data Rate ULPI Interface, with 8-bit-wide data bus ▪ 1'b1: Double Data Rate ULPI Interface, with 4-bit-wide data bus This bit is invalid.	Host and Device	1'h0	R_W
6	USB 2.0 High-Speed PHY or USB 1.1 Full-Speed Serial Transceiver Select (PHYSel) The application uses this bit to select either a high-speed UTMI+ or ULPI PHY, or a full-speed transceiver. ▪ 1'b0: USB 2.0 high-speed UTMI+ or ULPI PHY ▪ 1'b1: USB 1.1 full-speed serial transceiver This bit is always 0, with Read Only access.	Host and Device	1'h0	RO/R_W
5	Full-Speed Serial Interface Select (FSIntf) The application uses this bit to select either a unidirectional or bidirectional USB 1.1 full-speed serial transceiver interface. ▪ 1'b0: 6-pin unidirectional full-speed serial interface ▪ 1'b1: 3-pin bidirectional full-speed serial interface This bit is always 0, with Read Only access.	Host and Device	1'h0	RO/R_W
4	ULPI or UTMI+ Select (ULPI_UTMI_Sel) The application uses this bit to select either a UTMI+ interface or ULPI Interface.	Host and Device	1'h0	RO/R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b0: UTMI+ Interface ▪ 1'b1: ULPI Interface This bit takes effect only if GUSBCFG.PHYSel = 1'b0.			
3	PHY Interface (PHYIf) The application uses this bit to configure the core to support a UTMI+ PHY with an 8- or 16-bit interface. When a ULPI PHY is chosen, this must be set to 8-bit mode. ▪ 1'b0: 8 bits ▪ 1'b1: 16 bits	Host and Device	Configurable	RO/R_W
2:0	HS/FS Timeout Calibration (TOutCal) The number of PHY clocks that the application programs in this field is added to the high-speed/full-speed interpacket timeout duration in the core to account for any additional delays introduced by the PHY. This can be required, because the delay introduced by the PHY in generating the line state condition can vary from one PHY to another. The USB standard timeout value for high-speed operation is 736 to 816 (inclusive) bit times. The USB standard timeout value for full-speed operation is 16 to 18 (inclusive) bit times. The application must program this field based on the speed of enumeration. The number of bit times added per PHY clock are: High-speed operation: ▪ One 30-MHz PHY clock = 16 bit times ▪ One 60-MHz PHY clock = 8 bit times Full-speed operation: ▪ One 30-MHz PHY clock = 0.4 bit times ▪ One 60-MHz PHY clock = 0.2 bit times ▪ One 48-MHz PHY clock = 0.25 bit times Using the HS as an example, if you set ToutCal to '001' you add one 30MHz PHY clock or 16 bit times. If you set ToutCal to '010' you add two 30MHz PHY clocks or 32 bit times, and so on. The 3 bits allow you to add up to 7 PHY clocks, and the number of bit times depend on the speed, and the PHY clock you are using.	Host and Device	3'h0	R_W

33.5.3.5 Reset Register (GRSTCTL)

✧ Offset: 010h

The application uses this register to reset various hardware features inside the core.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-11 Reset Register: GRSTCTL

Field	Description	Mode	Reset	Access
31	AHB Master Idle (AHBIdle) Indicates that the AHB Master State Machine is in the IDLE condition.	Host and Device	1'b1	RO
30	DMA Request Signal (DMAReq) Indicates that the DMA request is in progress. Used for debug.	Host and Device	1'b1	RO
29:11	Reserved	Host and Device		RO
10:6	TxFIFO Number (TxFNum) This is the FIFO number that must be flushed using the TxFIFO Flush bit. This field must not be changed until the core clears the TxFIFO Flush bit. ▪ 5'h0: – Non-periodic TxFIFO flush in Host mode – Non-periodic TxFIFO flush in device mode when in shared FIFO operation – Tx FIFO 0 flush in device mode when in dedicated FIFO mode ▪ 5'h1: – Periodic TxFIFO flush in Host mode – Periodic TxFIFO 1 flush in Device mode when in shared FIFO operation – TXFIFO 1 flush in device mode when in dedicated FIFO mode ▪ 5'h2: – Periodic TxFIFO 2 flush in Device mode when in shared FIFO operation – TXFIFO 2 flush in device mode when in dedicated FIFO mode ... ▪ 5'hF: – Periodic TxFIFO 15 flush in Device mode when in shared FIFO operation – TXFIFO 15 flush in device mode when in dedicated FIFO mode ▪ 5'h10: – Flush all the transmit FIFOs in device or host mode.	Host and Device	5'b0	R_W
5	TxFIFO Flush (TxFFlsh)	Host and	1'b0	R_WS_S

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This bit selectively flushes a single or all transmit FIFOs, but cannot do so if the core is in the midst of a transaction. The application must write this bit only after checking that the core is neither writing to the TxFIFO nor reading from the TxFIFO. Verify using these registers: ▪ Read—NAK Effective Interrupt ensures the core is not reading from the FIFO. ▪ Write—GRSTCTL.AHBIdle ensures the core is not writing anything to the FIFO. Flushing is normally recommended when FIFOs are re-configured or when switching between Shared FIFO and Dedicated Transmit FIFO operation. FIFO flushing is also recommended during device endpoint disable. The application must wait until the core clears this bit before performing any operations. This bit takes eight clocks to clear, using the slower clock of phy_clk or hclk.	Device		C
4	RxFIFO Flush (RxFFlsh) The application can flush the entire RxFIFO using this bit, but must first ensure that the core is not in the middle of a transaction. The application must only write to this bit after checking that the core is neither reading from the RxFIFO nor writing to the RxFIFO. The application must wait until the bit is cleared before performing any other operations. This bit requires 8 clocks (slowest of PHY or AHB clock) to clear.	Host and Device	1'b0	R_WS_SC
3	IN Token Sequence Learning Queue Flush (INTknQFlsh) This bit is valid only if OTG_EN_DED_TX_FIFO = 0. The application writes this bit to flush the IN Token Sequence Learning Queue.	Device Only	1'b0	R_WS_SC
2	Host Frame Counter Reset (FrmCntrRst) The application writes this bit to reset the (micro)frame number counter inside the core. When the (micro)frame counter is reset, the subsequent SOF sent out by the core has a (micro)frame number of 0.	Host Only	1'b0	R_WS_SC
1	Reserved	Host and Device		RO
0	Core Soft Reset (CSftRst) Resets the hclk and phy_clock domains as follows: ▪ Clears the interrupts and all the CSR registers except the following register bits: – PCGCCTL.RstPdwnModule	Host and Device	1'b0	R_WS_SC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	– PCGCCTL.GateHclk – PCGCCTL.PwrClmp – GUSBCFG.DDRSel – GUSBCFG.PHYSel – GUSBCFG.FSIntf – GUSBCFG.ULPI_UTMI_Sel – GUSBCFG.PHYIf – HCFG.FSLSPclkSel – DCFG.DevSpd – DCTL.SftDiscon – GGPIO – GUSBCFG.TxEndDelay – GUSBCFG.TermSelDLPulse – GUSBCFG.ULPIClkSusM – GUSBCFG.ULPIAutoRes – GUSBCFG.ULPIFsLs – GPWRDN – GADPCTL ▪ All module state machines (except the AHB Slave Unit) are reset to the IDLE state, and all the transmit FIFOs and the receive FIFO are flushed. ▪ Any transactions on the AHB Master are terminated as soon as possible, after gracefully completing the last data phase of an AHB transfer. Any transactions on the USB are terminated immediately. ▪ When Hibernation or ADP feature is enabled, the PMU module is not reset by the Core Soft Reset. The application can write to this bit any time it wants to reset the core. This is a self-clearing bit and the core clears this bit after all the necessary logic is reset in the core, which can take several clocks, depending on the current state of the core. Once this bit is cleared software must wait at least 3 PHY clocks before doing any access to the PHY domain (synchronization delay). Software must also must check that bit 31 of this register is 1 (AHB Master is IDLE) before starting any operation. Typically software reset is used during software development and also when you dynamically change the PHY selection bits in the USB configuration registers listed above. When you change the PHY, the corresponding clock for the PHY is selected and used in the PHY domain. Once a new clock is selected, the PHY domain has to be			

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	reset for proper operation.			

33.5.3.6 Interrupt Register (GINTSTS)

✧ Offset: 014h

This register interrupts the application for system-level events in the current mode (Device mode or Host mode). It is shown in Figure 33-2.

Some of the bits in this register are valid only in Host mode, while others are valid in Device mode only. This register also indicates the current mode. to clear the interrupt status bits of type R_SS_WC, the application must write 1'b1 into the bit.

The FIFO status interrupts are read only; once software reads from or writes to the FIFO while servicing these interrupts, FIFO interrupt conditions are cleared automatically.

The application must clear the GINTSTS register at initialization before unmasking the interrupt bit to avoid any interrupts generated prior to initialization.

Table 33-12 Interrupt Register: GINTSTS

Field	Description	Mode	Reset	Access
31	Resume/Remote Wakeup Detected Interrupt (WkUpInt) Wakeup Interrupt during Suspend(L2) or LPM(L1) state. ▪ During Suspend(L2): – Device Mode - This interrupt is asserted only when Host Initiated Resume is detected on USB. – Host Mode - This interrupt is asserted only when Device Initiated Remote Wakeup is detected on USB. ▪ During LPM(L1): – Device Mode - This interrupt is asserted for either Host Initiated Resume or Device Initiated Remote Wakeup on USB. – Host Mode - This interrupt is asserted for either Host Initiated Resume or Device Initiated Remote Wakeup on USB.	Host and Device	1'b0	R_SS_WC
30	Session Request/New Session Detected Interrupt (SessReqInt) In Host mode, this interrupt is asserted when a session request is detected from the device. In Host mode, this interrupt is asserted when a session request is detected from the device.	Host and Device	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	In Device mode, this interrupt is asserted when the utmisrp_bvalid signal goes high. For more information on how to use this interrupt, see "FIFO RAM Allocation" in the Programming Guide.			
29	Disconnect Detected Interrupt (DisconnInt) This interrupt is asserted when a device disconnect is detected.	Host Only	1'b0	R_SS_WC
28	Connector ID Status Change (ConIDStsChng) This interrupt is asserted when there is a change in connector ID status.	Host and Device	1'b0	R_SS_WC
27	LPM Transaction Received Interrupt (LPM_Int) ▪ Device Mode - This interrupt is asserted when the device receives an LPM transaction and responds with a non-ERRORed response. ▪ Host Mode - This interrupt is asserted when the device responds to an LPM transaction with a non-ERRORed response or when the host core has completed LPM transactions for the programmed number of times (GLPMCFG.RetryCnt). This field is valid only if the Core LPM Configuration register's LPM- Capable (LPMCap) field is set to 1 and the User HW Config3 register's OTG_ENABLE_LPM bit is set to 1.	Host and Device	1'b0	R_SS_WC
26	Periodic TxFIFO Empty (PTxFEmp) This interrupt is asserted when the Periodic Transmit FIFO is either half or completely empty and there is space for at least one entry to be written in the Periodic Request Queue. The half or completely empty status is determined by the Periodic TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.PTxFEmpLvl).	Host Only	1'b1	RO
25	Host Channels Interrupt (HChInt) The core sets this bit to indicate that an interrupt is pending on one of the channels of the core (in Host mode). The application must read the Host All Channels Interrupt (HAINT) register to determine the exact number of the channel on which the interrupt occurred, and then read the corresponding Host Channel-n Interrupt (HCINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the HCINTn register to clear this bit.	Host Only	1'b0	RO
24	Host Port Interrupt (PrtInt) The core sets this bit to indicate a change in port status of one of the DWC_otg core ports in Host mode. The	Host Only	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	application must read the Host Port Control and Status (HPRT) register to determine the exact event that caused this interrupt. The application must clear the appropriate status bit in the Host Port Control and Status register to clear this bit.			
23	Reset Detected Interrupt (ResetDet) The core sets this status bit in device mode when reset is detected on the USB in L2 Suspend state. This bit is valid only when the core is Device Mode and is operating in Partial Power-Down, or Clock Gating modes of Suspend. This bit is not valid when device is in Hibernation mode of Suspend.	Device Only	1'b0	R_SS_WC
22	Data Fetch Suspended (FetSusp) This interrupt is valid only in DMA mode. This interrupt indicates that the core has stopped fetching data for IN endpoints due to the unavailability of TxFIFO space or Request Queue space. This interrupt is used by the application for an endpoint mismatch algorithm. For example, after detecting an endpoint mismatch, the application: ▪ Sets a global non-periodic IN NAK handshake ▪ Disables In endpoints ▪ Flushes the FIFO ▪ Determines the token sequence from the IN Token Sequence Learning Queue ▪ Re-enables the endpoints ▪ Clears the global non-periodic IN NAK handshake If the global non-periodic IN NAK is cleared, the core has not yet fetched data for the IN endpoint, and the IN token is received: the core generates an "IN token received when FIFO empty" interrupt. The OTG then sends the host a NAK response. To avoid this scenario, the application can check the GINTSTS.FetSusp interrupt, which ensures that the FIFO is full before clearing a global NAK handshake. Alternatively, the application can mask the "IN token received when FIFO empty" interrupt when clearing a global IN NAK handshake.	Device Only	1'b0	R_SS_WC
21	Incomplete Periodic Transfer (incomplP) In Host mode, the core sets this interrupt bit when there are incomplete periodic transactions still pending which are	Device only		

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	scheduled for the current microframe. Incomplete Isochronous OUT Transfer (incompISOOUT) The Device mode, the core sets this interrupt to indicate that there is at least one isochronous OUT endpoint on which the transfer is not completed in the current microframe. This interrupt is asserted along with the End of Periodic Frame Interrupt (EOPF) bit in this register.	Device only		
20	Incomplete Isochronous IN Transfer (incompISOIN) The core sets this interrupt to indicate that there is at least one isochronousIN endpoint on which the transfer is not completed in the current microframe. This interrupt is asserted along with the End of Periodic Frame Interrupt (EOPF) bit in this register. Note: This interrupt is not asserted in Scatter/Gather DMA mode.	Device only	1'b0	R_SS_WC
19	OUT Endpoints Interrupt (OEPInt) The core sets this bit to indicate that an interrupt is pending on one of the OUT endpoints of the core (in Device mode). The application must read the Device All Endpoints Interrupt (DAINT) register to determine the exact number of the OUT endpoint on which the interrupt occurred, and then read the corresponding Device OUT Endpoint-*n* Interrupt (DOEPINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the corresponding DOEPINTn register to clear this bit.	Device only	1'b0	RO
18	IN Endpoints Interrupt (IEPInt) The core sets this bit to indicate that an interrupt is pending on one of the IN endpoints of the core (in Device mode). The application must read the Device All Endpoints Interrupt (DAINT) register to determine the exact number of the IN endpoint on which the interrupt occurred, and then read the corresponding Device IN Endpoint-*n* Interrupt (DIEPINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the corresponding DIEPINTn register to clear this bit.	Device only	1'b0	RO
17	Endpoint Mismatch Interrupt (EPMis) **Note**: This interrupt is valid only in shared FIFO operation. ndicates that an IN token has been received for a non-periodic endpoint, but the data for another endpoint is present in the top of the Non-periodic Transmit FIFO and the IN endpoint mismatch count programmed by the	Device only	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	application has expired.			
16	Restore Done Interrupt (RstrDoneInt) The core sets this bit to indicate that the restore command after Hibernation was completed by the core. The core continues from Suspended state into the mode dictated by PCGCCTL.RestoreMode field. This bit is valid only when Hibernation feature is enabled (OTG_EN_PWROPT=2)	Host and Device mode	1'b0	R_SS_WC
15	End of Periodic Frame Interrupt (EOPF) Indicates that the period specified in the Periodic Frame Interval field of the Device Configuration register (DCFG.PerFrInt) has been reached in the current microframe.	Device only	1'b0	R_SS_WC
14	Isochronous OUT Packet Dropped Interrupt (ISOOutDrop) The core sets this bit when it fails to write an isochronous OUT packet into the RxFIFO because the RxFIFO does not have enough space to accommodate a maximum packet size packet for the isochronous OUT endpoint.	Device only	1'b0	R_SS_WC
13	Enumeration Done (EnumDone) The core sets this bit to indicate that speed enumeration is complete. The application must read the Device Status (DSTS) register to obtain the enumerated speed.	Device only	1'b0	R_SS_WC
12	USB Reset (USBRst) The core sets this bit to indicate that a reset is detected on the USB.	Device only	1'b0	R_SS_WC
11	USB Suspend (USBSusp) The core sets this bit to indicate that a suspend was detected on the USB. The core enters the Suspended state when there is no activity on the utmi_linestate signal for an extended period of time.	Device only	1'b0	R_SS_WC
10	Early Suspend (ErlySusp) The core sets this bit to indicate that an Idle state has been detected on the USB for 3 ms.	Device only	1'b0	R_SS_WC
9	I^2C Interrupt (I2CINT) The core sets this interrupt when I^2C access is completed on the I^2C interface. This field is used only if the I^2C interface was enabled. Otherwise, reads return 0.	Host and Device	1'b0	R_SS_WC
8	ULPI Carkit Interrupt (ULPICKINT) This field is used only if the Carkit interface was enabled. Otherwise, reads return 0.	Host and Device	1'b0	R_SS_WC
7	Global OUT NAK Effective (GOUTNakEff) Indicates that the Set Global OUT NAK bit in the Device	Device only	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Control register (DCTL.SGOUTNak), set by the application, has taken effect in the core. This bit can be cleared by writing the Clear Global OUT NAK bit in the Device Control register (DCTL.CGOUTNak).			
6	Global IN Non-Periodic NAK Effective (GINNakEff) Indicates that the Set Global Non-periodic IN NAK bit in the Device Control register (DCTL.SGNPInNak), set by the application, has taken effect in the core. That is, the core has sampled the Global IN NAK bit set by the application. This bit can be cleared by clearing the Clear Global Non-periodic IN NAK bit in the Device Control register (DCTL.CGNPInNak). This interrupt does not necessarily mean that a NAK handshake is sent out on the USB. The STALL bit takes precedence over the NAK bit.	Device only	1'b0	RO
5	Non-Periodic TxFIFO Empty (NPTxFEmp) This interrupt is valid only when OTG_EN_DED_TX_FIFO = 0. This interrupt is asserted when the Non-periodic TxFIFO is either half or completely empty, and there is space for at least one entry to be written to the Non-periodic Transmit Request Queue. The half or completely empty status is determined by the Non-periodic TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl).	Host and Device	1'b1	RO
4	RxFIFO Non-Empty (RxFLvl) Indicates that there is at least one packet pending to be read from the RxFIFO.	Host and Device	1'b0	RO
3	Start of (micro)Frame (Sof) In Host mode, the core sets this bit to indicate that an SOF (FS), micro-SOF (HS), or Keep-Alive (LS) is transmitted on the USB. The application must write a 1 to this bit to clear the interrupt. In Device mode, in the core sets this bit to indicate that an SOF token has been received on the USB. The application can read the Device Status register to get the current (micro)frame number. This interrupt is seen only when the core is operating at either HS or FS. **Note:** This register may return 1'b1 if read immediately after power on reset. If the register bit reads 1'b1 immediately after power on reset it does not indicate that an SOF has been sent (in case of host mode) or SOF has been received (in case of device mode). The read value of this interrupt is	Host and Device	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	valid only after a valid connection between host and device is established. If the bit is set after power on reset the application can clear the bit.			
2	OTG Interrupt (OTGInt) The core sets this bit to indicate an OTG protocol event. The application must read the OTG Interrupt Status (GOTGINT) register to determine the exact event that caused this interrupt. The application must clear the appropriate status bit in the GOTGINT register to clear this bit.	Host and Device	1'b0	RO
1	Mode Mismatch Interrupt (ModeMis) The core sets this bit when the application is trying to access: ▪ A Host mode register, when the core is operating in Device mode ▪ A Device mode register, when the core is operating in Host mode The register access is completed on the AHB with an OKAY response, but is ignored by the core internally and does not affect the operation of the core.	Host and Device	1'b0	R_SS_WC
0	Current Mode of Operation (CurMod) Indicates the current mode. ▪ 1'b0: Device mode ▪ 1'b1: Host mode	Host and Device	1'b0	RO

33.5.3.7 Interrupt Mask Register (GINTMSK)

✧ Offset: 018h

This register works with the "Interrupt Register (GINTSTS)" to interrupt the application. When an interrupt bit is masked, the interrupt associated with that bit is not generated. However, the GINTSTS register bit corresponding to that interrupt is still set.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-13 Interrupt Mask Register: GINTMSK

Field	Description	Mode	Reset	Access
31	Resume/Remote Wakeup Detected Interrupt Mask (WkUpIntMsk)	Host and Device	1'b0	R_W
30	Session Request/New Session Detected Interrupt Mask (SessReqIntMsk)	Host and Device	1'b0	R_W
29	Disconnect Detected Interrupt Mask (DisconnIntMsk)	Host Only	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28	Connector ID Status Change Mask (ConIDStsChngMsk)	Host and Device	1'b0	R_W
27	LPM Transaction Received Interrupt Mask (LPM_IntMsk)		1'b0	
26	Periodic TxFIFO Empty Mask (PTxFEmpMsk)	Host Only	1'b1	R_W
25	Host Channels Interrupt Mask (HChIntMsk)	Host Only	1'b0	R_W
24	Host Port Interrupt Mask (PrtIntMsk)	Host Only	1'b0	R_W
23	Reset Detected Interrupt Mask (ResetDetMsk)	Device Only	1'b0	R_W
22	Data Fetch Suspended Mask (FetSuspMsk)	Device Only	1'b0	R_W
21	Incomplete Periodic Transfer Mask (incomplPMsk) Incomplete Isochronous OUT Transfer Mask (incompISOOUTMsk)	Host only Device only	1'b0	R_W
20	Incomplete Isochronous IN Transfer Mask (incompISOINMsk) This bit is enabled only when device periodic endpoints are enabled in Dedicated TxFIFO mode.	Device only	1'b0	R_W
19	OUT Endpoints Interrupt Mask (OEPIntMsk)	Device only	1'b0	R_W
18	IN Endpoints Interrupt Mask (IEPIntMsk)	Device only	1'b0	R_W
17	Endpoint Mismatch Interrupt Mask (EPMisMsk)	Device only	1'b0	R_W
16	Restore Done Interrupt Mask (RstrDoneIntMsk) This field is valid only when Hibernation feature is enabled (OTG_EN_PWROPT=2).	Host and Device mode	1'b0	R_W
15	End of Periodic Frame Interrupt Mask (EOPFMsk)	Device only	1'b0	R_W
14	Isochronous OUT Packet Dropped Interrupt Mask (ISOOutDropMsk)	Device only	1'b0	R_W
13	Enumeration Done Mask (EnumDoneMsk)	Device only	1'b0	R_W
12	USB Reset Mask (USBRstMsk)	Device only	1'b0	R_W
11	USB Suspend Mask (USBSuspMsk)	Device only	1'b0	R_W
10	Early Suspend Mask (ErlySuspMsk)	Device only	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

9	I^2C Interrupt Mask (I2CIntMsk)	Host and Device	1'b0	R_W
8	ULPI Carkit Interrupt Mask (ULPICKINTMsk) I^2C Carkit Interrupt Mask (I2CCKINTMsk)	Host and Device	1'b0	R_W
7	Global OUT NAK Effective Mask (GOUTNakEffMsk)	Device only	1'b0	R_W
6	Global Non-periodic IN NAK Effective Mask (GINNakEffMsk)	Device only	1'b0	R_W
5	Non-periodic TxFIFO Empty Mask (NPTxFEmpMsk)	Host and Device	1'b1	R_W
4	Receive FIFO Non-Empty Mask (RxFLvlMsk)	Host and Device	1'b0	R_W
3	Start of (micro)Frame Mask (SofMsk)	Host and Device	1'b0	R_W
2	OTG Interrupt Mask (OTGIntMsk)	Host and Device	1'b0	R_W
1	Mode Mismatch Interrupt Mask (ModeMisMsk)	Host and Device	1'b0	R_W
0	Reserved	Host and Device		RO

33.5.3.8 Receive Status Debug Read/Status Read and Pop Registers (GRXSTSR/GRXSTSP)

✧ Offset for Read: 01Ch

✧ Offset for Pop: 020h

A read to the Receive Status Debug Read register returns the contents of the top of the Receive FIFO. A read to the Receive Status Read and Pop register additionally pops the top data entry out of the RxFIFO.

The receive status contents must be interpreted differently in Host and Device modes. The core ignores the receive status pop/read when the receive FIFO is empty and returns a value of 32'h0000_0000. The application must only pop the Receive Status FIFO when the Receive FIFO Non-Empty bit of the Core Interrupt register (GINTSTS.RxFLvl) is asserted.

Note:

- Use of these fields vary based on whether the HS OTG core is functioning as a host or a device.
- Do not read this register's reset value before configuring the core because the read value is "X" in the simulation.

Following table shows Host mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-14 Host Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP

Field	Description	Reset	Acces
31:21	Reserved		RO
20:17	Packet Status (PktSts) ▪ 4'b0010: IN data packet received ▪ 4'b0011: IN transfer completed (triggers an interrupt) ▪ 4'b0101: Data toggle error (triggers an interrupt) ▪ 4'b0111: Channel halted (triggers an interrupt) ▪ Others: Reserved	4'b0	RO
16:15	Data PID (DPID) Indicates the Data PID of the received packet	2'b0	RO
14:4	Byte Count (BCnt) Indicates the byte count of the received IN data packet.	11'b0	RO
3:0	Channel Number (ChNum) Indicates the channel number to which the current received packet belongs.	4'h0	RO

Following table shows Device mode

Table 33-15 Device Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP

Field	Description	Reset	Acces
31:25	Reserved		RO
24:21	Frame Number (FN) This is the least significant 4 bits of the (micro)frame number in which the packet is received on the USB. This field is supported only when isochronous OUT endpoints are supported.	4'b0	RO
20:17	Packet Status (PktSts) Indicates the status of the received packet ▪ 4'b0001: Global OUT NAK (triggers an interrupt) ▪ 4'b0010: OUT data packet received ▪ 4'b0011: OUT transfer completed (triggers an interrupt) ▪ 4'b0100: SETUP transaction completed (triggers an interrupt) ▪ 4'b0110: SETUP data packet received ▪ Others: Reserved	4'b0	RO
16:15	Data PID (DPID) Indicates the Data PID of the received OUT data packet ▪ 2'b00: DATA0 ▪ 2'b10: DATA1 ▪ 2'b01: DATA2 ▪ 2'b11: MDATA	2'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

14:4	Byte Count (BCnt) Indicates the byte count of the received data packet.	11'h0	RO
3:0	Endpoint Number (EPNum) Indicates the endpoint number to which the current received packet belongs.	4'b0	RO

33.5.3.9 Receive FIFO Size Register (GRXFSIZ)

✧ Offset: 024h

The application can program the RAM size that must be allocated to the RxFIFO.

Table 33-16 Receive FIFO Size Register: GRXFSIZ

Field	Description	Reset	Acces
31:16	Reserved		RO
15:0	RxFIFO Depth (RxFDep) This value is in terms of 32-bit words. ▪ Minimum value is 16 ▪ Maximum value is 3648.	16'd3648	R_W

33.5.3.10 Non-Periodic Transmit FIFO Size Register (GNPTXFSIZ)

✧ Offset: 028h

The application can program the RAM size and the memory start address for the Non-periodic TxFIFO.

Note: The fields of this register change, depending on host or device mode.

Table 33-17 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Host Mode and Device Shared FIFO Mode)

Field	Description	Reset	Acces
31:16	Non-periodic TxFIFO Depth (NPTxFDep) For host mode, this field is always valid. For Device mode, this field is invalid. This value is in terms of 32-bit words. ▪ Minimum value is 16 ▪ Maximum value is 3648	Configurable	RO/R_W
15:0	Non-periodic Transmit RAM Start Address (NPTxFStAddr) For host mode, this field is always valid. This field contains the memory start address for Non-periodic Transmit FIFO RAM. ▪ The application can write a new value in this field. Programmed values must not exceed the power-on value set	Configurable	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-18 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Device Dedicated FIFO Mode)

Field	Description	Reset	Acces
31:16	IN Endpoint TxFIFO 0 Depth (INEPTxF0Dep)This field is valid only for Device mode . This value is in terms of 32-bit words. ▪ Minimum value is 16 ▪ Maximum value is 3648	Configurable	RO/R_W
15:0	IN Endpoint FIFO0 Transmit RAM Start Address (INEPTxF0StAddr) For Device mode, this field is valid. This field contains the memory start address for IN Endpoint Transmit FIFO# 0. Programmed values must not exceed the power-on value set.	Configurable	RO/R_W

33.5.3.11 Non-Periodic Transmit FIFO/Queue Status Register (GNPTXSTS)

✧ Offset: 02Ch

In Device mode, this register is valid only in Shared FIFO operation.

This read-only register contains the free space information for the Non-periodic TxFIFO and the Non-periodic Transmit Request Queue.

Table 33-19 Non-Periodic Transmit FIFO/Queue Status Register: GNPTXSTS

Field	Description	Reset	Acces
31:24	Top of the Non-periodic Transmit Request Queue (NPTxQTop) Entry in the Non-periodic Tx Request Queue that is currently being processed by the MAC. ▪ Bits [30:27]: Channel/endpoint number ▪ Bits [26:25]: ▪ 2'b00: IN/OUT token – 2'b01: Zero-length transmit packet (device IN/host OUT) – 2'b10: PING/CSPLIT token – 2'b11: Channel halt command ▪ Bit [24]: Terminate (last entry for selected channel/endpoint)	7'b0	RO
23:16	Non-periodic Transmit Request Queue Space Available (NPTxQSpcAvail) Indicates the amount of free space available in the Non-periodic Transmit Request Queue. This queue holds both IN and OUT requests in Host mode. Device mode has only IN requests. ▪ 8'h0: Non-periodic Transmit Request Queue is full ▪ 8'h1: 1 location available ▪ 8'h2: 2 locations available	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ n : n locations available (0 ≤n ≤8) ▪ Others: Reserved		
15:0	Non-periodic TxFIFO Space Avail (NPTxFSpcAvail) Indicates the amount of free space available in the Non-periodic TxFIFO. Values are in terms of 32-bit words. ▪ 16'h0: Non-periodic TxFIFO is full ▪ 16'h1: 1 word available ▪ 16'h2: 2 words available ▪ 16'h*n*: *n* words available (where 0 ≤n ≤32,768) ▪ 16'h8000: 32,768 words available ▪ Others: Reserved	Configurable	RO

33.5.3.12 User HW Config1 Register (GHWCFG1)

✧ Offset: 044

This register contains the logical endpoint direction(s) selected.

Table 33-20 User HW Config1 Register: GHWCFG1

Field	Description	Reset	Acces
31:0	Endpoint Direction (epdir) This 32-bit field uses two bits per endpoint to determine the endpoint direction. Endpoint ▪ Bits [31:30]: Endpoint 15 direction ▪ Bits [29:28]: Endpoint 14 direction ... ▪ Bits [3:2]: Endpoint 1 direction ▪ Bits[1:0]: Endpoint 0 direction (always BIDIR) Direction ▪ 2'b00: BIDIR (IN and OUT) endpoint ▪ 2'b01: IN endpoint ▪ 2'b10: OUT endpoint ▪ 2'b11: Reserved This field is configured using "Name: OTG_EP_DIR_1(n)".	Configurable	RO

33.5.3.13 User HW Config2 Register (GHWCFG2)

✧ Offset: 048h

This register contains configuration options selected.

Table 33-21 User HW Config2 Register: GHWCFG2

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Field	Description	Reset	Acces
31	OTG_ENABLE_IC_USB IC_USB mode specified for mode of operation (parameter OTG_ENABLE _IC_USB) in coreConsultant. To choose IC_USB_MODE, both OTG_FSPHY_INTERFACE and OTG_ENABLE_IC_USB must be 1.	Configurable	RO
30:26	Device Mode IN Token Sequence Learning Queue Depth (TknQDepth) Range: 0-30 This field is configured using "Name: OTG_TOKEN_QUEUE_DEPTH".	Configurable	RO
25:24	Host Mode Periodic Request Queue Depth (PTxQDepth) ▪ 2'b00: 2 ▪ 2'b01: 4 ▪ 2'b10: 8 ▪ 2'b11: 16 This field is configured using parameter "Name: OTG_PERIO_TX_QUEUE_DEPTH"	Configurable	RO
23:22	Non-periodic Request Queue Depth (NPTxQDepth) ▪ 2'b00: 2 ▪ 2'b01: 4 ▪ 2'b10: 8 ▪ Others: Reserved This field is configured using parameter "Name: OTG_NPERIO_TX_QUEUE_DEPTH".	Configurable	RO
21	Reserved		RO
20	Multi Processor Interrupt Enabled (MultiProcIntrpt) ▪ 1'b0: No ▪ 1'b1: Yes This field is configured using parameter "Name: OTG_MULTI_PROC_INTRPT".	Configurable	RO
19	Dynamic FIFO Sizing Enabled (DynFifoSizing) ▪ 1'b0: No ▪ 1'b1: Yes This field is configured using parameter "Name: OTG_DFIFO_DYNAMIC".	Configurable	RO
18	Periodic OUT Channels Supported in Host Mode (PerioSupport) ▪ 1'b0: No ▪ 1'b1: Yes This field is configured using parameter "Name: OTG_EN_PERIO_HOST".	Configurable	RO
17:14	Number of Host Channels (NumHstChnl) Indicates the number of host channels supported by the core in Host mode. The range of this field is 0-15: 0 specifies 1 channel, 15	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	specifies 16 channels. This field is configured using parameter "Name: OTG_NUM_HOST_CHAN".		
13:10	Number of Device Endpoints (NumDevEps) Indicates the number of device endpoints supported by the core in Device mode in addition to control endpoint 0. The range of this field is 1-15. This field is configured using parameter "Name: OTG_NUM_EPS".	Configurable	RO
9:8	Full-Speed PHY Interface Type (FSPhyType) ▪ 2'b00: Full-speed interface not supported ▪ 2'b01: Dedicated full-speed interface ▪ 2'b10: FS pins shared with UTMI+ pins ▪ 2'b11: FS pins shared with ULPI pins This field is configured using parameter "Name: OTG_FSPHY_INTERFACE".	Configurable	RO
7:6	High-Speed PHY Interface Type (HSPhyType) ▪ 2'b00: High-Speed interface not supported ▪ 2'b01: UTMI+ ▪ 2'b10: ULPI ▪ 2'b11: UTMI+and ULPI This field is configured using parameter "Name: OTG_HSPHY_INTERFACE".	Configurable	RO
5	Point-to-Point (SingPnt) 1'b0: Multi-point application (hub and split support) 1'b1: Single-point application (no hub and no split support) This field is configured using parameter "Name: OTG_SINGLE_POINT".	Configurable	RO
4:3	Architecture (OtgArch) ▪ 2'b00: Slave-Only ▪ 2'b01: External DMA ▪ 2'b10: Internal DMA ▪ Others: Reserved This field is configured using parameter "Name: OTG_ARCHITECTURE".	Configurable	RO
2:0	Mode of Operation (OtgMode) ▪ 3'b000: HNP- and SRP-Capable OTG (Host and Device) ▪ 3'b001: SRP-Capable OTG (Host and Device) ▪ 3'b010: Non-HNP and Non-SRP Capable OTG (Host and Device) ▪ 3'b011: SRP-Capable Device ▪ 3'b100: Non-OTG Device ▪ 3'b101: SRP-Capable Host ▪ 3'b110: Non-OTG Host	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ Others: Reserved This field is configured using parameter "Name: OTG_MODE".		

33.5.3.14 User HW Config3 Register (GHWCFG3)

✧ Offset: 04Ch

This register contains the configuration options selected.

Table 33-22 User HW Config3 Register: GHWCFG3

Field	Description	Reset	Acces
31:16	DFIFO Depth (DfifoDepth minus EP_LOC_CNT) This value is in terms of 32-bit words. ▪ Minimum value is 32 ▪ Maximum value is 32,768 This field is configured using parameter "Name: OTG_DFIFO_DEPTH" and EP_LOC_CNT. For more information on EP_LOC_CNT, see "Endpoint Information Controller (EPINFO_CTL)".	Configurable	RO
15	OTG_ENABLE_LPM LPM mode specified for Mode of Operation (parameter OTG_ENABLE_LPM) in coreConsultant configuration.	Configurable	RO
14	OTG_BC_SUPPORT This bit indicates the HS OTG controller support for Battery Charger. ▪ 0 - No Battery Charger Support ▪ 1 - Battery Charger support present.	Configurable	RO
13	OTG_ENABLE_HSIC HSIC mode specified for Mode of Operation in coreConsultant configuration (parameter OTG_ENABLE_HSIC). **Value Range:** 0-1 ▪ 1: HSIC-capable with shared UTMI PHY interface ▪ 0: Non-HSIC-capable **Parameter Name:** OTG_ENABLE_HSIC **Dependencies:** OTG_HSPHY_INTERFACE **Default:** Non-HSIC-capable		RO
12	OTG_ADP_SUPPORT This bit indicates whether ADP logic is present within or external to the HS OTG controller ▪ 0 - No ADP logic present with HS OTG controller ▪ 1- ADP logic is present along with HS OTG controller.		RO
11	Reset Style for Clocked always Blocks in RTL (RstType) ▪ 1'b0: Asynchronous reset is used in the core ▪ 1'b1: Synchronous reset is used in the core	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This field is configured using parameter "Name: OTG_SYNC_RESET_TYPE".		
10	Optional Features Removed (OptFeature) Indicates whether the User ID register, GPIO interface ports, and SOF toggle and counter ports were removed for gate count optimization by enabling Remove Optional Features? during coreConsultant configuration. ▪ 1'b0: No ▪ 1'b1: Yes This field is configured using parameter "Name: OTG_RM_OPT_FEATURES".	Configurable	RO
9	Vendor Control Interface Support (VndctlSupt) ▪ 1'b0: Vendor Control Interface is not available on the core. ▪ 1'b1: Vendor Control Interface is available. This field is configured using parameter "Name: OTG_VENDOR_CTL_INTERFACE".	Configurable	RO
8	I^2C Selection (I2CIntSel) ▪ 1'b0: I 2C Interface is not available on the core. ▪ 1'b1: I 2C Interface is available on the core. This field is configured using parameter "Name: OTG_I2C_INTERFACE".	Configurable	RO
7	OTG Function Enabled (OtgEn) The application uses this bit to indicate the DWC_otg core's OTG capabilities. ▪ 1'b0: Not OTG capable ▪ 1'b1: OTG Capable This field is configured using parameter "Name: OTG_MODE".	Configurable	RO
6:4	Width of Packet Size Counters (PktSizeWidth) ▪ 3'b000: 4 bits ▪ 3'b001: 5 bits ▪ 3'b010: 6 bits ▪ 3'b011: 7 bits ▪ 3'b100: 8 bits ▪ 3'b101: 9 bits ▪ 3'b110: 10 bits ▪ Others: Reserved This field is configured using parameter "Name: OTG_PACKET_COUNT_WIDTH".	Configurable	RO
3:0	Width of Transfer Size Counters (XferSizeWidth) ▪ 4'b0000: 11 bits ▪ 4'b0001: 12 bits ... ▪ 4'b1000: 19 bits	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ Others: Reserved This field is configured using parameter "Name: OTG_TRANS_COUNT_WIDTH".		

33.5.3.15 User HW Config4 Register (GHWCFG4)

✧ Offset: 050h

This register contains the configuration options selected.

Note: Bit [31] is available only when Scatter/Gather DMA mode is enabled. When Scatter/Gather DMA mode is disabled, this field is reserved.

Table 33-23 User HW Config4 Register: GHWCFG4

Field	Description	Reset	Acces
31	Scatter/Gather DMA ▪ 1'b1: Dynamic configuration This field is configured using parameter "Name: OTG_EN_DESC_DMA" .	Configurable	RO
30	Scatter/Gather DMA configuration ▪ 1'b0: Non-Scatter/Gather DMA configuration ▪ 1'b1: Scatter/Gather DMA configuration This field is configured using parameter "Name: OTG_EN_DESC_DMA".	Configurable	RO
29:26	Number of Device Mode IN Endpoints Including Control Endpoints (INEps) Range 0 -15 ▪ 0:1 IN Endpoint ▪ 1:2 IN Endpoints .... ▪ 15:16 IN Endpoints This field is configured using parameter "Name: OTG_NUM_IN_EPS".	Configurable	RO
25	Enable Dedicated Transmit FIFO for device IN Endpoints (DedFifoMode) ▪ 1'b0: Dedicated Transmit FIFO Operation not enabled. ▪ 1'b1: Dedicated Transmit FIFO Operation enabled. This field is configured using parameter "Name: OTG_EN_DED_TX_FIFO".	Configurable	RO
24	session_end Filter Enabled (SessEndFltr) ▪ 1'b0: No filter ▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_SESSIONEND_FILTER"	Configurable	RO
23	"b_valid" Filter Enabled (BValidFltr) ▪ 1'b0: No filter	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_B_VALID_FILTER".		
22	"a_valid" Filter Enabled (AValidFltr) ▪ 1'b0: No filter ▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_A_VALID_FILTER".	Configurable	RO
21	"vbus_valid" Filter Enabled (VBusValidFltr) ▪ 1'b0: No filter ▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_VBUSVALID_FILTER".	Configurable	RO
20	"iddig" Filter Enable (IddgFltr) ▪ 1'b0: No filter ▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_IDDIG_FILTER".	Configurable	RO
19:16	Number of Device Mode Control Endpoints in Addition to Endpoint 0 (NumCtlEps) Range: 0-15 This field is configured using parameter "Name: OTG_NUM_CRL_EPS".	Configurable	RO
15:14	UTMI+ PHY/ULPI-to-Internal UTMI+ Wrapper Data Width (PhyDataWidth) When a ULPI PHY is used, an internal wrapper converts ULPI to UTMI+. ▪ 2'b00: 8 bits ▪ 2'b01: 16 bits ▪ 2'b10: 8/16 bits, software selectable ▪ Others: Reserved This field is configured using parameter "Name: OTG_HSPHY_DWIDTH".	Configurable	RO
13:6	Reserved		RO
6	Enable Hibernation ▪ 1'b0: Hibernation feature not enabled ▪ 1'b1: Hibernation feature enabled This field is configured using parameter "Name: OTG_EN_PWROPT".	Configurable	RO
5	Minimum AHB Frequency Less Than 60 MHz (AhbFreq) ▪ 1'b0: No ▪ 1'b1: Yes This field is configured using parameter "Name: OTG_MIN_AHB_FREQ_LESSTHAN_60".	Configurable	RO
4	Enable Partial Power Down	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b0: Partial Power Down Not Enabled ▪ 1'b1: Partial Power Down Enabled This field is configured using parameter "Name: OTG_EN_PWROPT".		
3:0	Number of Device Mode Periodic IN Endpoints (NumDevPerioEps) Range: 0-15 This field is configured using parameter "Name: OTG_NUM_PERIO_EPS".	Configurable	RO

33.5.3.16 DFIFO Software Config Register (GDFIFOCFG)

✧ Offset: 05Ch

Table 33-24 Global DFIFO Software Config Register: GDFIFOCFG

Field	Description	Mode	Reset	Acces
31:16	EPInfoBaseAddr This field provides the start address of the EP info controller.	Host and Device	EPINFO_BASEADDR	R_W
15:0	GDFIFOCfg This field is for dynamic programming of the DFIFO Size. This value takes effect only when the application programs a non zero value to this register. The value programmed must conform to the guidelines described in "FIFO RAM Allocation" in the Programming Guide. The DWC_otg core does not have any corrective logic if the FIFO sizes are programmed incorrectly.	Host and Device	OTG_DFIFO_DEPTH	R_W

33.5.3.17 Host Periodic Transmit FIFO Size Register (HPTXFSIZ)

✧ Offset: 100h

This register holds the size and the memory start address of the Periodic TxFIFO.

Table 33-25 Host Periodic Transmit FIFO Size Register: HPTXFSIZ

Field	Description	Reset	Acces
31:16	Host Periodic TxFIFO Depth (PTxFSize) This value is in terms of 32-bit words. ▪ Minimum value is 16 ▪ Maximum value is 3648	Configurable	R_W
15:0	Host Periodic TxFIFO Start Address (PTxFStAddr) The power-on reset value of this register is the sum of the Largest Rx Data FIFO Depth and Largest Non-periodic Tx Data FIFO Depth specified .These parameters are:	Configurable	RO/R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	In shared FIFO operation:- ▪ OTG_RX_DFIFO_DEPTH(3648) + OTG_TX_NPERIO_DFIFO_DEPTH(1024) In dedicated FIFO mode:- ▪ OTG_RX_DFIFO_DEPTH(3648) + OTG_TX_HNPERIO_DFIFO_DEPTH(3648)		

33.5.3.18 Device Periodic Transmit FIFO-*n* Size Register (DPTXFSIZn)

✧ FIFO_number: 1 ≤ n ≤ 15

✧ Offset: 104h + (FIFO_number - 1) * 04h

This register is valid only in shared FIFO operation (OTG_EN_DED_TX_FIFO = 0).

This register holds the memory start address of each periodic TxFIFO to be implemented in Device mode. Each periodic FIFO holds the data for one periodic IN endpoint. This register is repeated for each periodic FIFO instantiated.

Table 33-26 Device Periodic Transmit FIFO-n Register: DPTXFSIZn

Field	Description	Reset	Acces
31:16	Device Periodic TxFIFO Size (DPTxFSize) This value is in terms of 32-bit words. ▪ Minimum value is 4 ▪ Maximum value is 768	Configurable	RO
15:0	Device Periodic TxFIFO RAM Start Address (DPTxFStAddr) This field specifies the start address in the RAM for this periodic FIFO. The power- on reset value of this register is the sum of the Largest Rx Data FIFO Depth, Largest Non-periodic Tx Data FIFO Depth, and all lower numbered Largest Device Mode Periodic Tx Data FIFOn Depth specified . The formula used is: OTG_RX_DFIFO_DEPTH + OTG_TX_NPERIO_DFIFO_DEPTH + SUM of OTG_TX_DPERIO_DFIFO_DEPTH_'x' (where x=1 to n-1). When n = 1, the above expression becomes OTG_RX_DFIFO_DEPTH + OTG_TX_NPERIO_DFIFO_DEPTH. If at POR, the calculated value (C) exceeds 65535, then the Reset value becomes Reset Value(A) = (C - 65536).	Configurable	RO/R_W

33.5.3.19 Device IN Endpoint Transmit FIFO Size Register: (DIEPTXFn)

✧ FIFO_number: 1 ≤ n ≤ 15

✧ Offset: 104h + (FIFO_number - 1) * 04h

This register is valid only in dedicated FIFO mode.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

This register holds the size and memory start address of IN endpoint TxFIFOs implemented in Device mode. Each FIFO holds the data for one IN endpoint. This register is repeated for instantiated IN endpoint FIFOs 1 to 15. For IN endpoint FIFO 0 use GNPTXFSIZ register for programming the size and memory start address.

Table 33-27 Device In Endpoint Transmit FIFO Size Register: (DIEPTXFn)

Field	Description	Reset	Acces
31:16	IN Endpoint TxFIFO Depth (INEPnTxFDep) This value is in terms of 32-bit words. Minimum value is 16 Maximum value is 3648	Configurable	R_W
15:0	IN Endpoint FIFO*n* Transmit RAM Start Address (INEPnTxFStAddr) This field contains the memory start address for IN endpoint Transmit FIFOn (0 < n <= 15). The power-on reset value of this register is calculated according to the following formula: OTG_RX_DFIFO_DEPTH + SUM of OTG_TX_DINEP_DFIFO_DEPTH_'x' (where x = 0 to n - 1) If at POR the calculated value (C) exceeds 65535, then the Reset value becomes Reset Value(A) = (C - 65536). Example: If start address of IN endpoint FIFO 1 is OTG_RX_DFIFO_DEPTH + OTG_TX_DINEP_DFIFO_DEPTH_ 0 and start address of IN endpoint FIFO 2 is OTG_RX_DFIFO_DEPTH + OTG_TX_DINEP_DFIFO_DEPTH_ 0 + OTG_TX_DINEP_DFIFO_DEPTH_ 1.	Configurable	RO/R_W

33.5.4 Host Mode Registers

These registers affect the operation of the core in the Host mode. Host mode registers must not be accessed in Device mode, as the results are undefined. Host Mode registers can be categorized as follows:

33.5.4.1 Host Configuration Register (HCFG)

✧ Offset: 400h

This register configures the core after power-on. Do not make changes to this register after initializing the host.

Table 33-28 Host Configuration Register: HCFG

Field	Description	Reset	Acces
31	Mode Change Ready Timer Enable (ModeChTimEn)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This bit is used to enable or disable the host core to wait for 200 PHY clock cycles at the end of Resume to change the opmode signal to the PHY to 00 after Suspend or LPM. ▪ 1'b0: The Host core waits for either 200 PHY clock cycles or a linestate of SE0 at the end of resume to change the opmode from 2'b10 to 2'b00. ▪ 1'b1: The Host core waits only for a linestate of SE0 at the end of resume to change the opmode from 2'b10 to 2'b00.		
30:27	Reserved		RO
26	Enable Periodic Scheduling (PerSchedEna) Applicable in Scatter/Gather DMA mode only. Enables periodic scheduling within the core. Initially, the bit is reset. The core will not process any periodic channels. As soon as this bit is set, the core will get ready to start scheduling periodic channels. In non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R_W
25:24	Frame List Entries (FrListEn). The value in the register specifies the number of entries in the Frame list. This field is valid only in Scatter/Gather DMA mode. ▪ 2'b00: 8 Entries ▪ 2'b01:16 Entries ▪ 2'b10: 32 Entries ▪ 2'b11: 64 Entries In modes other than Scatter/Gather DMA mode, these bits are reserved.	2'b00	R/W
23	Enable Scatter/gather DMA in Host mode (DescDMA) When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. NOTE: This bit must be modified only once after a reset. The following combinations are available for programming: ▪ GAHBCFG.DMAEn=0, HCFG.DescDMA=0 => Slave mode ▪ GAHBCFG.DMAEn=0, HCFG.DescDMA=1 => Invalid ▪ GAHBCFG.DMAEn=1, HCFG.DescDMA=0 => Buffered DMA mode ▪ GAHBCFG.DMAEn=1, HCFG.DescDMA=1 => Scatter/Gather DMA mode In non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R/W
22:16	Reserved	8'd2	R_W
15:8	Resume Validation Period (ResValid) This field is effective only when HCFG.Ena32KHzS is set. It controls the resume period when the core resumes from suspend. The core counts the ResValid number of clock cycles to detect a valid resume when this is set.	8'd2	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

7	Enable 32-KHz Suspend Mode (Ena32KHzS) This bit can only be set if the USB 1.1 Full-Speed Serial Transceiver Interface has been selected. If USB 1.1 Full-Speed Serial Transceiver Interface has not been selected, this bit must be zero. When the USB 1.1 Full-Speed Serial Transceiver Interface is chosen and this bit is set, the core expects the 48-MHz PHY clock to be switched to 32 KHz during a suspend.	1'd0	R_W
6:3	Reserved		RO
2	FS- and LS-Only Support (FSLSSupp) The application uses this bit to control the core's enumeration speed. Using this bit, the application can make the core enumerate as a FS host, even if the connected device supports HS traffic. Do not make changes to this field after initial programming. ▪ 1'b0: HS/FS/LS, based on the maximum speed supported by the connected device ▪ 1'b1: FS/LS-only, even if the connected device can support HS	1'b0	R_W
1:0	FS/LS PHY Clock Select (FSLSPclkSel) When the core is in FS Host mode ▪ 2'b00: Internal PHY clock is running at 30/60 MHZ (For UTMI+/ULPI PHY Interfaces) ▪ 2'b01: Internal PHY clock is running at 48MHZ (For 1.1 FS transceiver Interface) ▪ Others: Reserved When the core is in LS Host mode ▪ 2'b00: Internal PHY clock is running at 30/60 MHZ (For UTMI+/ULPI PHY Interfaces) ▪ 2'b10: Internal PHY clock is running at 6 MHZ and the external clock is running at 48MHZ. When you select a 6 MHz clock during LS Mode, you must do a soft reset (for 1.1 FS transceiver Interface) ▪ Others: Reserved Notes: ▪ When Core in FS mode, the internal and external clocks have the same frequency. ▪ When Core in LS mode, – If FSLSPclkSel = 2'b00: Internal and external clocks have the same frequency – If FSLSPclkSel = 2'b10: Internal clock is divided by eight version of external 48 MHz clock (utmifs_clk).	2'b0	R_W

33.5.4.2 Host Frame Interval Register (HFIR)

✧ Offset: 404h

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

This register stores the frame interval information for the current speed to which the DWC_otg core has enumerated.

Table 33-29 Host Frame Interval Register: HFIR

Field	Description	Reset	Acces
31:17	Reserved		RO
16	Reload Control (HFIRRldCtrl) This bit allows dynamic reloading of the HFIR register during runtime. ▪ 1'b0: The HFIR cannot be reloaded dynamically ▪ 1'b1: the HFIR can be dynamically reloaded during runtime. This bit needs to be programmed during initial configuration and its value must not be changed during runtime.	1'b0	R_W
15:0	Frame Interval (FrInt) The value that the application programs to this field specifies the interval between two consecutive SOFs (FS) or micro-SOFs (HS) or Keep-Alive tokens (LS). This field contains the number of PHY clocks that constitute the required frame interval. The default value set in this field for a FS operation when the PHY clock frequency is 60 MHz. The application can write a value to this register only after the Port Enable bit of the Host Port Control and Status register (HPRT.PrtEnaPort) has been set. If no value is programmed, the core calculates the value based on the PHY clock specified in the FS/LS PHY Clock Select field of the Host Configuration register (HCFG.FSLSPclkSel). ▪ 125 μs * (PHY clock frequency for HS) ▪ 1 ms * (PHY clock frequency for FS/LS)	16'd60000	R_W

33.5.4.3 Host Frame Number/Frame Time Remaining Register (HFNUM)

✧ Offset: 408h

This register indicates the current frame number. It also indicates the time remaining (in terms of the number of PHY clocks) in the current (micro)frame.

Table 33-30 Host Frame Number/Frame Time Remaining Register: HFNUM

Field	Description	Reset	Acces
31:16	Frame Time Remaining (FrRem) Indicates the amount of time remaining in the current microframe (HS) or frame (FS/LS), in terms of PHY clocks. This field decrements on each PHY clock. When it reaches zero, this field is reloaded with the value in the Frame Interval register and a new SOF is transmitted on the USB.	16'h0	RO
15:0	Frame Number (FrNum)	16'h3FFF	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This field increments when a new SOF is transmitted on the USB, and is reset to 0 when it reaches 16'h3FFF.		

33.5.4.4 Host Periodic Transmit FIFO/Queue Status Register (HPTXSTS)

✧ Offset: 410h

This read-only register contains the free space information for the Periodic TxFIFO and the Periodic Transmit Request Queue.

Table 33-31 Host Periodic Transmit FIFO/Queue Status Register: HPTXSTS

Field	Description	Reset	Acces
31:24	Top of the Periodic Transmit Request Queue (PTxQTop) This indicates the entry in the Periodic Tx Request Queue that is currently being processes by the MAC. This register is used for debugging. ▪ Bit [31]: Odd/Even (micro)frame – 1'b0: send in even (micro)frame – 1'b1: send in odd (micro)frame ▪ Bits [30:27]: Channel/endpoint number ▪ Bits [26:25]: Type – 2'b00: IN/OUT – 2'b01: Zero-length packet – 2'b10: CSPLIT – 2'b11: Disable channel command ▪ Bit [24]: Terminate (last entry for the selected channel/endpoint)	8'h0	RO
23:16	Periodic Transmit Request Queue Space Available (PTxQSpcAvail) Indicates the number of free locations available to be written in the Periodic Transmit Request Queue. This queue holds both IN and OUT requests. ▪ 8'h0: Periodic Transmit Request Queue is full ▪ 8'h1: 1 location available ▪ 8'h2: 2 locations available ▪ n : n locations available (0 ≤n ≤16) ▪ Others: Reserved	Configurable	RO
15:0	Periodic Transmit Data FIFO Space Available (PTxFSpcAvail) Indicates the number of free locations available to be written to in the Periodic TxFIFO. Values are in terms of 32-bit words ▪ 16'h0: Periodic TxFIFO is full ▪ 16'h1: 1 word available ▪ 16'h2: 2 words available ▪ 16'hn: n words available (where 0 ≤n ≤32,768) ▪ 16'h8000: 32,768 words available	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ Others: Reserved		

33.5.4.5 Host All Channels Interrupt Register (HAINT)

✧ Offset: 414h

When a significant event occurs on a channel, the Host All Channels Interrupt register interrupts the application using the Host Channels Interrupt bit of the Core Interrupt register (GINTSTS.HChInt). This is shown in Figure 33-2. There is one interrupt bit per channel, up to a maximum of 16 bits. Bits in this register are set and cleared when the application sets and clears bits in the corresponding Host Channel-n Interrupt register.

Table 33-32 Host All Channels Interrupt Register: HAINT

Field	Description	Reset	Acces
31:16	Reserved		RO
15:0	Channel Interrupts (HAINT) One bit per channel: Bit 0 for Channel 0, bit 15 for Channel 15	16'h0	RO

33.5.4.6 Host All Channels Interrupt Mask Register (HAINTMSK)

✧ Offset: 418h

The Host All Channel Interrupt Mask register works with the Host All Channel Interrupt register to interrupt the application when an event occurs on a channel. There is one interrupt mask bit per channel, up to a maximum of 16 bits.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-33 Host All Channels Interrupt Mask Register: HAINTMSK

Field	Description	Reset	Acces
31:16	Reserved		RO
15:0	Channel Interrupt Mask (HAINTMsk) One bit per channel: Bit 0 for channel 0, bit 15 for channel 15	16'h0	R_W

33.5.4.7 Host Port Control and Status Register (HPRT)

✧ Offset: 440h

This register is available only in Host mode. Currently, the OTG Host supports only one port.

A single register holds USB port-related information such as USB reset, enable, suspend, resume, connect status, and test mode for each port. It is shown in Figure 33-2. The R_SS_WC bits in this register can trigger an interrupt to the application through the Host Port Interrupt bit of the Core Interrupt register (GINTSTS.PrtInt). On a Port Interrupt, the application must read this register and clear the bit that caused the interrupt. For the R_SS_WC bits, the application must write a 1 to the bit to clear the interrupt.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-34 Host Port Control and Status Register: HPRT

Field	Description	Reset	Acces
31:19	Reserved		RO
18:17	Port Speed (PrtSpd) Indicates the speed of the device attached to this port. ▪ 2'b00: High speed ▪ 2'b01: Full speed ▪ 2'b10: Low speed ▪ 2'b11: Reserved	2'h0	RO
16:13	Port Test Control (PrtTstCtl) The application writes a nonzero value to this field to put the port into a Test mode, and the corresponding pattern is signaled on the port. ▪ 4'b0000: Test mode disabled ▪ 4'b0001: Test_J mode ▪ 4'b0010: Test_K mode ▪ 4'b0011: Test_SE0_NAK mode ▪ 4'b0100: Test_Packet mode ▪ 4'b0101: Test_Force_Enable ▪ Others: Reserved For more information on moving the DWC_otg host core into test mode, see "Moving the Host Core to Test Mode" on page 89.	4'h0	R_W
12	Port Power (PrtPwr) The application uses this field to control power to this port (write 1'b1 to set to 1'b1 and write 1'b0 to set to 1'b0), and the core can clear this bit on an over current condition. ▪ 1'b0: Power off ▪ 1'b1: Power on	1'b0	RO
11:10	Port Line Status (PrtLnSts) Indicates the current logic level USB data lines ▪ Bit [10]: Logic level of D+ ▪ Bit [11]: Logic level of D-	2'b0	RO
9	Reserved		RO
8	Port Reset (PrtRst) When the application sets this bit, a reset sequence is started on this port. The application must time the reset period and clear this bit after the reset sequence is complete. ▪ 1'b0: Port not in reset ▪ 1'b1: Port in reset To start a reset on the port, the application must leave this bit set for at least the minimum duration mentioned below, as specified in the USB 2.0 specification, Section 7.1.7.5. The application can	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	leave it set for another 10 ms in addition to the required minimum duration, before clearing the bit, even though there is no maximum limit set by the USB standard. ▪ High speed: 50 ms ▪ Full speed/Low speed: 10 ms		
7	Port Suspend (PrtSusp) The application sets this bit to put this port in Suspend mode. The core only stops sending SOFs when this is set. To stop the PHY clock, the application must set the Port Clock Stop bit, which asserts the suspend input pin of the PHY. The read value of this bit reflects the current suspend status of the port. This bit is cleared by the core after a remote wakeup signal is detected or the application sets the Port Reset bit or Port Resume bit in this register or the Resume/Remote Wakeup Detected Interrupt bit or Disconnect Detected Interrupt bit in the Core Interrupt register (GINTSTS.WkUpInt or GINTSTS.DisconnInt, respectively). ▪ 1'b0: Port not in Suspend mode ▪ 1'b1: Port in Suspend mode	1'b0	R_WS_SC
6	Port Resume (PrtRes) The application sets this bit to drive resume signaling on the port. The core continues to drive the resume signal until the application clears this bit. If the core detects a USB remote wakeup sequence, as indicated by the Port Resume/Remote Wakeup Detected Interrupt bit of the Core Interrupt register (GINTSTS.WkUpInt), the core starts driving resume signaling without application intervention and clears this bit when it detects a disconnect condition. The read value of this bit indicates whether the core is currently driving resume signaling. ▪ 1'b0: No resume driven ▪ 1'b1: Resume driven When LPM is enabled and the core is in the L1 (Sleep) state, setting this bit results in the following behavior: The core continues to drive the resume signal until a pre-determined time specified in the GLPMCFG.HIRD_Thres[3:0] field. If the core detects a USB remote wakeup sequence, as indicated by the Port L1 Resume/Remote L1 Wakeup Detected Interrupt bit of the Core Interrupt register (GINTSTS.L1WkUpInt), the core starts driving resume signaling without application intervention and clears this bit at the end of the resume. The read value of this bit indicates whether the core is currently driving resume	1'b0	R_W_SS_SC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	signaling. ▪ 1'b0: No resume driven ▪ 1'b1: Resume driven		
5	Port Overcurrent Change (PrtOvrCurrChng) The core sets this bit when the status of the Port Overcurrent Active bit (bit 4) in this register changes.	1'b0	R_SS_WC
4	Port Overcurrent Active (PrtOvrCurrAct) Indicates the overcurrent condition of the port. ▪ 1'b0: No overcurrent condition ▪ 1'b1: Overcurrent condition	1'b0	RO
3	Port Enable/Disable Change (PrtEnChng) The core sets this bit when the status of the Port Enable bit [2] of this register changes.	1'b0	R_SS_WC
2	Port Enable (PrtEna) A port is enabled only by the core after a reset sequence, and is disabled by an overcurrent condition, a disconnect condition, or by the application clearing this bit. The application cannot set this bit by a register write. It can only clear it to disable the port. This bit does not trigger any interrupt to the application. ▪ 1'b0: Port disabled ▪ 1'b1: Port enabled	1'b0	R_SS_SC_WC
1	Port Connect Detected (PrtConnDet) The core sets this bit when a device connection is detected to trigger an interrupt to the application using the Host Port Interrupt bit of the Core Interrupt register (GINTSTS.PrtInt). The application must write a 1 to this bit to clear the interrupt.	1'b0	R_SS_WC
0	Port Connect Status (PrtConnSts) ▪ 0: No device is attached to the port. ▪ 1: A device is attached to the port.	1'b0	RO

33.5.4.7.1 Moving the Host Core to Test Mode

To move the DWC_otg core to test mode, you must set HPRT.Port Test Control. Complete the following steps to move the DWC_otg core to test mode:

1. Power on the core.
2. Load the DWC_otg driver.
3. Connect an HS device and enumerate to HS mode.
4. Access the HPRT register to send test packets.
5. Remove the device and connect to fixture (OPT) port.

The DWC_otg host core continues sending out test packets.

6. Test the eye diagram.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.5.4.8 Host Channel-*n* Characteristics Register (HCCHARn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 500h + (Channel_number * 20h)

Table 33-35 Host Channel-n Characteristics Register: HCCHARn

Field	Description	Reset	Acces
31	Channel Enable (ChEna) When Scatter/Gather mode is enabled ▪ 1'b0: Indicates that the descriptor structure is not yet ready. ▪ 1'b1: Indicates that the descriptor structure and data buffer with data is setup and this channel can access the descriptor. When Scatter/Gather mode is disabled This field is set by the application and cleared by the OTG host. ▪ 1'b0: Channel disabled ▪ 1'b1: Channel enabled	1'b0	R_WS_SC
30	Channel Disable (ChDis) The application sets this bit to stop transmitting/receiving data on a channel, even before the transfer for that channel is complete. The application must wait for the Channel Disabled interrupt before treating the channel as disabled.	1'b0	R_WS_SC_SS
29	Odd Frame (OddFrm) This field is set (reset) by the application to indicate that the OTG host must perform a transfer in an odd (micro)frame. This field is applicable for only periodic (isochronous and interrupt) transactions. ▪ 1'b0: Even (micro)frame ▪ 1'b1: Odd (micro)frame This field is not applicable for Scatter/Gather DMA mode and need not be programmed by the application and is ignored by the core.	1'b0	R_W
28:22	Device Address (DevAddr) This field selects the specific device serving as the data source or sink.	7'h0	R_W
21:20	Multi Count (MC) / Error Count (EC) When the Split Enable bit of the Host Channel-n Split Control register (HCSPLTn.SpltEna) is reset (1'b0), this field indicates to the host the number of transactions that must be executed per microframe for this periodic endpoint. For non periodic transfers, this field is used only in DMA mode, and specifies the number packets to be fetched for this channel before the internal DMA engine changes aubitration.	2'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 2'b00: Reserved This field yields undefined results. ▪ 2'b01: 1 transaction ▪ 2'b10: 2 transactions to be issued for this endpoint per microframe ▪ 2'b11: 3 transactions to be issued for this endpoint per microframe When HCSPLTn.SpltEna is set (1'b1), this field indicates the number of immediate retries to be performed for a periodic split transactions on transaction errors. This field must be set to at least 2'b01.		
19:18	Endpoint Type (EPType) Indicates the transfer type selected. ▪ 2'b00: Control ▪ 2'b01: Isochronous ▪ 2'b10: Bulk ▪ 2'b11: Interrupt	2'b0	R_W
17	Low-Speed Device (LSpdDev) This field is set by the application to indicate that this channel is communicating to a low-speed device. The application must program this bit when a low speed device is connected to the host through an FS HUB. The HS OTG Host core uses this field to drive the XCVR_SELECT signal to 2'b11 while communicating to the LS Device through the FS hub. **Note:** In a peer to peer setup, the HS OTG Host core ignores this bit even if it is set by the application software.	1'b0	R_W
16	Reserved		RO
15	Endpoint Direction (EPDir) Indicates whether the transaction is IN or OUT. ▪ 1'b0: OUT ▪ 1'b1: IN	1'b0	R_W
14:11	Endpoint Number (EPNum) Indicates the endpoint number on the device serving as the data source or sink.	4'h0	R_W
10:0	Maximum Packet Size (MPS) Indicates the maximum packet size of the associated endpoint.	11'h0	R_W

33.5.4.9 Host Channel-n Split Control Register (HCSPLTn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 504h + (Channel_number * 20h)

Table 33-36 Host Channel-n Split Control Register: HCSPLTn

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Field	Description	Reset	Acces
31	Split Enable (SpltEna) The application sets this field to indicate that this channel is enabled to perform split transactions.	1'b0	R_W
30:17	Reserved		RO
16	Do Complete Split (CompSplt) The application sets this field to request the OTG host to perform a complete split trsnsaction.	1'b0	R_W
15:14	Transaction Position (XactPos) This field is used to determine whether to send all, first, middle, or last payloads with each OUT transaction. ▪ 2'b11: All. This is the entire data payload is of this transaction (which is less than or equal to 188 bytes). ▪ 2'b10: Begin. This is the first data payload of this transaction (which is larger than 188 bytes). ▪ 2'b00: Mid. This is the middle payload of this transaction (which is larger than 188 bytes). ▪ 2'b01: End. This is the last payload of this transaction (which is larger than 188 bytes).	2'h0	R_W
13:7	Hub Address (HubAddr) This field holds the device address of the transaction translator's hub.	7'h0	R_W
6:0	Port Address (PrtAddr) This field is the port number of the recipient transaction translator.	7'h0	R_W

33.5.4.10 Host Channel-*n* Interrupt Register (HCINTn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 508h + (Channel_number * 20h)

This register indicates the status of a channel with respect to USB- and AHB-related events. It is shown in Figure 33-2. The application must read this register when the Host Channels Interrupt bit of the Core Interrupt register (GINTSTS.HChInt) is set. Before the application can read this register, it must first read the Host All Channels Interrupt (HAINT) register to get the exact channel number for the Host Channel-n Interrupt register. The application must clear the appropriate bit in this register to clear the corresponding bits in the HAINT and GINTSTS registers.

Table 33-37 Host Channel-n Interrupt Register: HCINTn

Field	Description	Reset	Acces
31:11	Reserved		RO
13	Descriptor rollover interrupt (DESC_LST_ROLLIntr) This bit is valid only when Scatter/Gather DMA mode is enabled.	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	The core sets this bit when the corresponding channel's descriptor list rolls over. For non Scatter/Gather DMA mode, this bit is reserved.		
12	Excessive Transaction Error (XCS_XACT_ERR) This bit is valid only when Scatter/Gather DMA mode is enabled. The core sets this bit when 3 consecutive transaction errors occurred on the USB bus. XCS_XACT_ERR will not be generated for Isochronous channels. For non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R_SS_WC
11	BNA (Buffer Not Available) Interrupt (BNAIntr) This bit is valid only when Scatter/Gather DMA mode is enabled. The core generates this interrupt when the descriptor accessed is not ready for the Core to process. BNA will not be generated for Isochronous channels. For non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R_SS_WC
10	Data Toggle Error (DataTglErr) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.	1'b0	R_SS_WC
9	Frame Overrun (FrmOvrun) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core	1'b0	R_SS_WC
8	Babble Error (BblErr) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.	1'b0	R_SS_WC
7	Transaction Error (XactErr) Indicates one of the following errors occurred on the USB. ▪ CRC check failure ▪ Timeout ▪ Bit stuff error ▪ False EOP In Scatter/Gather DMA mode, the interrupt due to this bit is not set.	1'b0	R_SS_WC
6	NYET Response Received Interrupt (NYET) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.	1'b0	R_SS_WC
5	ACK Response Received/Transmitted Interrupt (ACK) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.	1'b0	R_SS_WC
4	NAK Response Received Interrupt (NAK) In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.	1'b0	R_SS_WC
3	STALL Response Received Interrupt (STALL) In Scatter/Gather DMA mode, the interrupt due to this bit is	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	masked in the core.		
2	AHB Error (AHBErr) This is generated only in DMA mode when there is an AHB error during AHB read/write. The application can read the corresponding channel's DMA address register to get the error address.	1'b0	R_SS_WC
1	Channel Halted (ChHltd) In non Scatter/Gather DMA mode, it indicates the transfer completed abnormally either because of any USB transaction error or in response to disable request by the application or because of a completed transfer. In Scatter/Gather DMA mode, this indicates that transfer completed due to any of the following ▪ EOL being set in descriptor ▪ AHB error ▪ Excessive transaction errors ▪ In response to disable request by the application ▪ Babble ▪ Stall	1'b0	R_SS_WC
0	Transfer Completed (XferCompl) For Scatter/Gather DMA mode, it indicates that current descriptor processing got completed with IOC bit set in its descriptor. In non Scatter/Gather DMA mode, it indicates that Transfer completed normally without any errors.	1'b0	R_SS_WC

33.5.4.11 Host Channel-*n* Interrupt Mask Register (HCINTMSKn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 50Ch + (Channel_number * 20h)

This register reflects the mask for each channel status described in the previous section.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-38 Host Channel-n Interrupt Mask Register: HCINTMSKn

Field	Description	Reset	Acces
31:11	Reserved	1'b0	RO
13	Descriptor rollover interrupt Mask register (DESC_LST_ROLLIntrMsk) This bit is valid only when Scatter/Gather DMA mode is enabled. In non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R_W
12	Reserved	1'b0	RO
11	Reserved	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

11	BNA (Buffer Not Available) Interrupt mask register (BNAIntrMsk) This bit is valid only when Scatter/Gather DMA mode is enabled. In non Scatter/Gather DMA mode, this bit is reserved	1'b0	R_W
10	Data Toggle Error Mask (DataTglErrMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	RO
9	Frame Overrun Mask (FrmOvrunMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	R_W
8	Babble Error Mask (BblErrMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	R_W
7	Transaction Error Mask (XactErrMsk) This bit is not applicable in Scatter/Gather DMA mode	1'b0	R_W
6	NYET Response Received Interrupt Mask (NyetMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	R_W
5	ACK Response Received/Transmitted Interrupt Mask (AckMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	R_W
4	NAK Response Received Interrupt Mask (NakMsk) This bit is not applicable in Scatter/Gather DMA mode.	1'b0	R_W
3	STALL Response Received Interrupt Mask (StallMsk) This bit is not applicable in Scatter/Gather DMA mode.		
2	AHB Error Mask (AHBErrMsk) Note: This bit is only accessible when OTG_ARCHITECTURE = 2		
1	Channel Halted Mask (ChHltdMsk)	1'b0	R_W
0	Transfer Completed Mask (XferComplMsk)	1'b0	R_W

33.5.4.12 Host Channel-*n* Transfer Size Register (HCTSIZn)

✧ Channel_number: 0≤n≤15

✧ Offset: 510h + (Channel_number * 20h)

In Scatter/Gather DMA mode, the HCTSIZn register is defined as described in the table below.

Table 33-39 Host Channel-n Transfer Size Register: HCTSIZn

Field	Description	Reset	Acces
30:29	PID (Pid) The application programs this field with the type of PID to use for the initial transaction. The host maintains this field for the rest of the transfer. ▪ 2'b00: DATA0 ▪ 2'b01: DATA2 ▪ 2'b10: DATA1 ▪ 2'b11: MDATA (non-control)	2'b00	R_W
28:19	Reserved	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

18:16	Reserved	1'b0	RO
15:8	NTD (Number of Transfer Descriptors) **(Non Isochronous)** This value is in terms of number of descriptors. Maximum number of descriptor that can be present in the list is 64. The values can be from 0 to 63. ▪ 0 - 1 descriptor. ▪ 63 - 64 descriptors This field indicates the total number of descriptors present in that list. The core will wrap around after servicing NTD number of descriptors for that list. **(Isochronous)** This field indicates the number of descriptors present in that list.µframe The possible values for FS are ▪ 1 - 2 descriptors ▪ 3 - 4 descriptors ▪ 7 - 8 descriptors ▪ 15 - 16 descriptors ▪ 31 - 32 descriptors ▪ 63 - 64 descriptors The possible values for HS are ▪ 7 - 8 descriptors ▪ 15 - 16 descriptors ▪ 31 - 32 descriptors ▪ 63 - 64 descriptors ▪ 127 - 128 descriptors ▪ 255 - 256 descriptors	8'h0	R_W
7:0	SCHED_INFO (Schedule information) Every bit in this 8 bit register indicates scheduling for that microframe. Bit 0 indicates scheduling for 1st microframe and bit 7 indicates scheduling for 8th microframe in that frame. A value of 8'b11111111 indicates that the corresponding interrupt channel is scheduled to issue a token every microframe in that frame. A value of 8'b10101010 indicates that the corresponding interrupt channel is scheduled to issue a token every alternate microframe starting with second microframe. Note that this field is applicable only for periodic (Isochronous and Interrupt) channels.	8'h0	R_W
In Non-Scatter/Gather mode, HCTSIZn is defined as follows:			
31	Do Ping (DoPng) This bit is used only for OUT transfers. Setting this field to 1 directs the host to do PING protocol.	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Note: Do not set this bit for IN transfers. If this bit is set for IN transfers it disables the		
30:29	PID (Pid) The application programs this field with the type of PID to use for the initial transaction. The host maintains this field for the rest of the transfer. ▪ 2'b00: DATA0 ▪ 2'b01: DATA2 ▪ 2'b10: DATA1 ▪ 2'b11: MDATA (non-control)/SETUP (control)	2'b00	R_W
28:19	Packet Count (PktCnt) This field is programmed by the application with the expected number of packets to be transmitted (OUT) or received (IN). The host decrements this count on every successful transmission or reception of an OUT/IN packet. Once this count reaches zero, the application is interrupted to indicate normal completion.	10'h0	R_W
18:0	Transfer Size (XferSize) For an OUT, this field is the number of data bytes the host sends during the transfer. For an IN, this field is the buffer size that the application has Reserved for the transfer. The application is expected to program this field as an integer multiple of the maximum packet size for IN transactions (periodic and non-periodic).	19'h0	R_W

33.5.4.13 Host Channel-n DMA Address Register (HCDMAn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 514h + (Channel_number * 20h)

This register is used by the OTG host in the internal DMA mode to maintain the current buffer pointer for IN/OUT transactions. The starting DMA address must be DWORD-aligned.

Table 33-40 Host Channel-n DMA Address Register: HCDMAn

Field	Description	Reset	Acces
Buffer DMA Mode			
31:0	DMA Address (DMAAddr) This field holds the start address in the external memory from which the data for the endpoint must be fetched or to which it must be stored. This register is incremented on every AHB transaction.	"X" if not program med as the register is in SPRAM	R_W
Descriptor DMA Mode			

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td colspan="4">Note: For Scatter/Gather DMA mode, this address is the start of the page address where the descriptor list is located.</td></tr>
<tr><td>31:N
(Isoc)

31:9
(Non Isoc)</td><td>DMA Address (DMAAddr)
Non-Isochronous:
This field holds the start address of the 512 bytes page. The first descriptor in the list should be located in this address. The first descriptor may be or may not be ready. The core starts processing the list from the CTD value.
Isochronous:
This field holds the address of the 2*(nTD+1) bytes of locations in which the isochronous descriptors are present where N is based on nTD as per Table below
<table>
<tr><td>31:N</td><td>N-1:3</td><td>2:0</td></tr>
<tr><td>Base Address</td><td>Offset</td><td>000</td></tr>
</table>
<table>
<tr><td colspan="2">HS ISOC</td></tr>
<tr><td>nTD</td><td>N</td></tr>
<tr><td>7</td><td>6</td></tr>
<tr><td>15</td><td>7</td></tr>
<tr><td>31</td><td>8</td></tr>
<tr><td>63</td><td>9</td></tr>
<tr><td>127</td><td>10</td></tr>
<tr><td>255</td><td>11</td></tr>
</table>
<table>
<tr><td colspan="2">FS ISOC</td></tr>
<tr><td>nTD</td><td>N</td></tr>
<tr><td>1</td><td>4</td></tr>
<tr><td>3</td><td>5</td></tr>
<tr><td>7</td><td>6</td></tr>
<tr><td>15</td><td>7</td></tr>
<tr><td>31</td><td>8</td></tr>
<tr><td>63</td><td>9</td></tr>
</table></td><td>23'h0</td><td>R_W</td></tr>
<tr><td>N-1:3
(Isoc)

8:3
(Non Isoc)</td><td>Current Transfer Desc (CTD):
Non Isochronous:
This value is in terms of number of descriptors. The values can be from 0 to 63.
0 - 1 descriptor.
63- 64 descriptors.
This field indicates the current descriptor processed in the list. This field is updated both by application and the core. For example, if the application enables the channel after programming CTD=5, then the core will start processing the 6th descriptor. The address is obtained by adding a value of (8bytes*5=) 40(decimal) to DMAAddr.
Isochronous:
CTD for isochronous is based on the current frame/µframe value. Need to be set to zero by application.</td><td>6'h0</td><td>R/W</td></tr>
<tr><td>2:0</td><td>Reserved</td><td>3'h0</td><td>RO</td></tr>
</table>

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.5.4.14 Host Channel-n DMA Buffer Address Register (HCDMABn)

✧ Channel_number: 0 n 15

✧ Offset: 51Ch + (Channel_number * 20h)

This register is present only in case of Scatter/Gather DMA. It is implemented in RAM instead of flop-based implementation. This register holds the current buffer address.

Table 33-41 Host Channel-n DMA Buffer Address Register: HCDMABn

Field	Description	Reset	Acces
31:0	Holds the current buffer address. This register is updated as and when the data transfer for the corresponding end point is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.	"X" if not program med as the registeris in SPRAM	RO

33.5.4.15 Host Frame List Base Address Register (HFLBAddr)

✧ Offset: 41Ch

This register is present only in case of Scatter/Gather DMA. It is implemented as flops. This register holds the starting address of the Frame list information.

Table 33-42 Host Frame List Base Address Register: HFLBAddr

Field	Description	Reset	Acces
31:0	The starting address of the Frame list. This register is used only for Isochronous and Interrupt Channels.	32'h0	R/W

33.5.5 Device Mode Registers

These registers are visible only in Device mode and must not be accessed in Host mode, as the results are unknown. Some of them affect all the endpoints uniformly, while others affect only a specific endpoint.

Device Mode registers fall into two categories:

Device Logical IN Endpoint-Specific Registers

It's instantiates one set of endpoint registers per logical endpoint. A logical endpoint is unidirectional: it can be either IN or OUT. To represent a bidirectional endpoint, two logical endpoints are required, one for the IN direction and the other for the OUT direction. This is also true for control endpoints.

The registers and register fields described in this section can pertain to IN or OUT endpoints, or both, or specific endpoint types as noted.

33.5.5.1 Device Configuration Register (DCFG)

✧ Offset: 800h

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

This register configures the core in Device mode after power-on or after certain control commands or enumeration. Do not make changes to this register after initial programming.

Table 33-43 Device Configuration Register: DCFG

Field	Description	Reset	Acces
31:26	Resume Validation Period (ResValid) This field controls the period when the core resumes from a suspend. When this bit is set, the core counts for the ResValid number of clock cycles to detect a valid resume. This field is effective only when DCFG.Ena32KHzSusp is set.	6'd2	R_W
25:24	Periodic Scheduling Interval (PerSchIntvl) PerSchIntvl must be programmed only for Scatter/Gather DMA mode. **Description**: This field specifies the amount of time the Internal DMA engine must allocate for fetching periodic IN endpoint data. Based on the number of periodic endpoints, this value must be specified as 25,50 or 75% of (micro)frame. ▪ When any periodic endpoints are active, the internal DMA engine allocates the specified amount of time in fetching periodic IN endpoint data. ▪ When no periodic endpoints are active, then the internal DMA engine services non- periodic endpoints, ignoring this field. ▪ After the specified time within a (micro)frame, the DMA switches to fetching for non- periodic endpoints. ▪ 2'b00: 25% of (micro)frame. ▪ 2'b01: 50% of (micro)frame. ▪ 2'b10: 75% of (micro)frame. ▪ 2'b11: Reserved.	2'b00	R_W
23	Enable Scatter/Gather DMA in Device mode (DescDMA). When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. **NOTE:** This bit must be modified only once after a reset. The following combinations are available for programming: ▪ GAHBCFG.DMAEn=0,DCFG.DescDMA=0 => Slave mode ▪ GAHBCFG.DMAEn=0,DCFG.DescDMA=1 => Invalid ▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=0 => Buffered DMA mode ▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=1 => Scatter/Gather DMA mode	1'b0	R_W
22:18	IN Endpoint Mismatch Count (EPMisCnt) This field is valid only in shared FIFO operation.	5'h8	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	The application programs this filed with a count that determines when the core generates an Endpoint Mismatch interrupt (GINTSTS.EPMis). The core loads this value into an internal counter and decrements it. The counter is reloaded whenever there is a match or when the counter expires. The width of this counter depends on the depth of the Token Queue.		
17:13	Reserved		RO
13	Enable Device OUT NAK (EnDevOutNak) This bit enables setting NAK for Bulk OUT endpoints after the transfer is completed when the core is operating in Device Descriptor DMA mode. ▪ 1'b0: The core does not set NAK after Bulk OUT transfer complete ▪ 1'b1: The core sets NAK after Bulk OUT transfer complete This bit is valid only when OTG_EN_DESC_DMA == 1'b1	1'b0	R_W
12:11	Periodic Frame Interval (PerFrInt) Indicates the time within a (micro)frame at which the application must be notified using the End Of Periodic Frame Interrupt. This can be used to determine if all the isochronous traffic for that (micro)frame is complete. ▪ 2'b00: 80% of the (micro)frame interval ▪ 2'b01: 85% ▪ 2'b10: 90% ▪ 2'b11: 95%	2'h0	R_W
10:4	Device Address (DevAddr) The application must program this field after every SetAddress control command.	7'h0	R_W
3	Enable 32-KHz Suspend Mode (Ena32KHzS) When the USB 1.1 Full-Speed Serial Transceiver Interface is chosen and this bit is set, the core expects the 48-MHz PHY clock to be switched to 32 KHz during a suspend. This bit can only be set if USB 1.1 Full-Speed Serial Transceiver Interface has been selected. If USB 1.1 Full-Speed Serial Transceiver Interface has not been selected, this bit must be zero.	1'd0	R_W
2	Non-Zero-Length Status OUT Handshake (NZStsOUTHShk) The application can use this field to select the handshake the core sends on receiving a nonzero-length data packet during the OUT transaction of a control transfer's Status stage. ▪ 1'b1: Send a STALL handshake on a nonzero-length status OUT transaction and do not send the received OUT packet to the application. ▪ 1'b0: Send the received OUT packet to the application	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	(zero-length or nonzero- length) and send a handshake based on the NAK and STALL bits for the endpoint in the Device Endpoint Control register.		
1:0	Device Speed (DevSpd) Indicates the speed at which the application requires the core to enumerate, or the maximum speed the application can support. However, the actual bus speed is determined only after the chirp sequence is completed, and is based on the speed of the USB host to which the core is connected. ▪ 2'b00: High speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) ▪ 2'b01: Full speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) ▪ 2'b10: Low speed (USB 1.1 FS transceiver clock is 48 MHz) ▪ 2'b11: Full speed (USB 1.1 FS transceiver clock is 48 MHz)	2'b0	R_W

33.5.5.2 Device Control Register (DCTL)

✧ Offset: 804h

Table 33-44 evice Control Register: DCTL

Field	Description	Reset	Acces
31:17	Reserved		RO
17	Enable Continue on BNA (EnContOnBNA) This bit enables the DWC_otg core to continue on BNA for Bulk OUT endpoints. With this feature enabled, when a Bulk OUT endpoint receives a BNA interrupt the core starts processing the descriptor that caused the BNA interrupt after the endpoint re-enables the endpoint. ▪ 1'b0: After receiving BNA interrupt, the core disables the endpoint. When the endpoint is re-enabled by the application, the core starts processing from the DOEPDMA descriptor. ▪ 1'b1: After receiving BNA interrupt, the core disables the endpoint. When the endpoint is re-enabled by the application, the core starts processing from the descriptor that received the BNA interrupt. This bit is valid only when OTG_EN_DESC_DMA == 1'b1. It is a one-time programmable after reset bit like any other DCTL register bits.	1'b0	R_W
16	Set NAK automatically on babble (NakOnBble). The core sets NAK automatically for the endpoint on which babble is received.	1'b0	R_W
15	Ignore frame number for isochronous endpoints (IgnrFrmNum)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Slave Mode (GAHBCFG.DMAEn=0): This bit is not valid in Slave mode and should not be programmed to 1.Non- Scatter/Gather DMA mode (GAHBCFG.DMAEn=1,DCFG.DescDMA=0): This bit is not used when Threshold mode is enabled and should not be programmed to 1. In non-Scatter/Gather DMA mode, the application receives transfer complete interrupt after transfers for multiple (micro)frames are completed. ▪ When Scatter/Gather DMA mode is disabled, this field is used by the application to enable periodic transfer interrupt. The application can program periodic endpoint transfers for multiple (micro)frames. – 0: Periodic transfer interrupt feature is disabled; the application must program transfers for periodic endpoints every (micro)frame – 1: Packets are not flushed when an ISOC IN token is received for an elapsed frame. The core ignores the frame number, sending packets as soon as the packets are ready, and the corresponding token is received. This field is also used by the application to enable periodic transfer interrupts. Scatter/Gather DMA Mode (GAHBCFG.DMAEn=1,DCFG.DescDMA=1): This bit is not applicable to high-speed, high-bandwidth transfers and should not be programmed to 1. In addition, this bit is not used when Threshold mode is enabled and should not be programmed to 1. ▪ 0: The core transmits the packets only in the frame number in which they are intended to be transmitted. – 1: Packets are not flushed when an ISOC IN token is received for an elapsed frame. The core ignores the frame number, sending packets as soon as the packets are ready, and the corresponding token is received. When this bit is set, there must be only one packet per descriptor.		
14:13	Global Multi Count (GMC) GMC must be programmed only once after initialization. Applicable only for Scatter/Gather DMA mode. This indicates the number of packets to be serviced for that end point before moving to the next end point. It is only for non- periodic end points. ▪ 2'b00: Invalid.	2'h0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 2'b01: 1 packet. ▪ 2'b10: 2 packets. ▪ 2'b11: 3 packets. The value of this field automatically changes to 2'h1 when DCFG.DescDMA is set to 1. When Scatter/Gather DMA mode is disabled, this field is reserved and reads 2'b00.		
12	Reserved		RO
11	Power-On Programming Done (PWROnPrgDone) The application uses this bit to indicate that register programming is completed after a wake-up from Power Down mode.	1'b0	WO
10	Clear Global OUT NAK (CGOUTNak) A write to this field clears the Global OUT NAK.	1'b0	WO
9	Set Global OUT NAK (SGOUTNak) A write to this field sets the Global OUT NAK. The application uses this bit to send a NAK handshake on all OUT endpoints. The application must set the this bit only after making sure that the Global OUT NAK Effective bit in the Core Interrupt Register (GINTSTS.GOUTNakEff) is cleared.	1'b0	WO
8	Clear Global Non-periodic IN NAK (CGNPInNak) A write to this field clears the Global Non-periodic IN NAK.	1'b0	WO
7	Set Global Non-periodic IN NAK (SGNPInNak) A write to this field sets the Global Non-periodic IN NAK.The application uses this bit to send a NAK handshake on all non-periodic IN endpoints. The core can also set this bit when a timeout condition is detected on a non-periodic endpoint in shared FIFO operation. The application must set this bit only after making sure that the Global IN NAK Effective bit in the Core Interrupt Register (GINTSTS.GINNakEff) is cleared.	1'b0	WO
6:4	Test Control (TstCtl) ▪ 3'b000: Test mode disabled ▪ 3'b001: Test_J mode ▪ 3'b010: Test_K mode ▪ 3'b011: Test_SE0_NAK mode ▪ 3'b100: Test_Packet mode ▪ 3'b101: Test_Force_Enable Others: Reserved	3'h0	R_W
3	Global OUT NAK Status (GOUTNakSts) ▪ 1'b0: A handshake is sent based on the FIFO Status and	1'b0	WO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	the NAK and STALL bit settings. ▪ 1'b1: No data is written to the RxFIFO, irrespective of space availability. Sends a NAK handshake on all packets, except on SETUP transactions. All isochronous OUT packets are dropped.		
2	Global Non-periodic IN NAK Status (GNPINNakSts) ▪ 1'b0: A handshake is sent out based on the data availability in the transmit FIFO. ▪ 1'b1: A NAK handshake is sent out on all non-periodic IN endpoints, irrespective of the data availability in the transmit FIFO.	1'b0	WO
1	Soft Disconnect (SftDiscon) The application uses this bit to signal the DWC_otg core to do a soft disconnect. As long as this bit is set, the host does not see that the device is connected, and the device does not receive signals on the USB. The core stays in the disconnected state until the application clears this bit. The minimum duration for which the core must keep this bit set is specified in Table 33-53. ▪ 1'b0: Normal operation. When this bit is cleared after a soft disconnect, the core drives the phy_opmode_o signal on the UTMI+ to 2'b00, which generates a device connect event to the USB host. When the device is reconnected, the USB host restarts device enumeration. ▪ 1'b1: The core drives the phy_opmode_o signal on the UTMI+ to 2'b01, which generates a device disconnect event to the USB host. Note: This bit can be also used for ULPI/FS Serial interfaces. Note: This bit is not impacted by a soft reset.	1'b1	R_W
0	Remote Wakeup Signaling (RmtWkUpSig) When the application sets this bit, the core initiates remote signaling to wake the USB host. The application must set this bit to instruct the core to exit the Suspend state. As specified in the USB 2.0 specification, the application must clear this bit 1-15 ms after setting it.If LPM is enabled and the core is in the L1 (Sleep) state, when the application sets this bit, the core initiates L1 remote signaling to wake up the USB host. The application must set this bit to instruct the core to exit the Sleep state. As specified in the LPM specification, the hardware automatically clears this bit 50 µs ($T_{L1DevDrvResume}$) after being set by the application. The application must not set this bit when GLPMCFG bRemoteWake from the previous LPM transaction is zero.	1'b0	R_W

Following table lists the minimum duration under various conditions for which the Soft Disconnect

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

(SftDiscon) bit must be set for the USB host to detect a device disconnect. To accommodate clock jitter, it is recommended that the application add some extra delay to the specified minimum duration.

Table 33-45 Minimum Duration for Soft Disconnect

Operating Speed	Device State	Minimum Duration
High speed	Suspended	1 ms + 2.5 μs
High speed	Idle	3 ms + 2.5 μs
High speed	Not Idle or Suspended (Performing transactions)	125 μs
Full speed/Low speed	Suspended	1 ms + 2.5 μs
Full speed/Low speed	Idle	2.5 μs
Full speed/Low speed	Not Idle or Suspended (Performing transactions)	2.5 μs

33.5.5.3 Device Status Register (DSTS)

✧ Offset: 808h

This register indicates the status of the core with respect to USB-related events. It must be read on interrupts from Device All Interrupts (DAINT) register.

Table 33-46 Device Status Register: DSTS

Field	Description	Reset	Acces
31:22	Reserved		RO
21:8	Frame or Microframe Number of the Received SOF (SOFFN) When the core is operating at high speed, this field contains a microframe number. When the core is operating at full or low speed, this field contains a frame number. **Note:** This register may return a non zero value if read immediately after power on reset. In case the register bit reads non zero immediately after power on reset it does not indicate that SOF has been received from the host. The read value of this interrupt is valid only after a valid connection between host and device is established.	14'h0	RO
7:4	Reserved		RO
3	Erratic Error (ErrticErr) The core sets this bit to report any erratic errors (phy_rxvalid_i/phy_rxvldh_i or phy_rxactive_i is asserted for at least 2 ms, due to PHY error) seen on the UTMI+. Due to erratic errors, the DWC_otg core goes into Suspended state and an interrupt is generated to the application with Early Suspend bit of the Core Interrupt register (GINTSTS.ErlySusp). If the early suspend is asserted due to an erratic error, the	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	application can only perform a soft disconnect recover.		
2:1	Enumerated Speed (EnumSpd) Indicates the speed at which the DWC_otg core has come up after speed detection through a chirp sequence. ▪ 2'b00: High speed (PHY clock is running at 30 or 60 MHz) ▪ 2'b01: Full speed (PHY clock is running at 30 or 60 MHz) ▪ 2'b10: Low speed (PHY clock is running at 48 MHz, internal phy_clk at 6 MHz) ▪ 2'b11: Full speed (PHY clock is running at 48 MHz) Low speed is not supported for devices using a UTMI+ PHY.	2'h01	RO
0	Suspend Status (SuspSts) In Device mode, this bit is set as long as a Suspend condition is detected on the USB. The core enters the Suspended state when there is no activity on the utmi_linestate signal for an extended period of time. The core comes out of the suspend: ▪ When there is any activity on the utmi_linestate signal ▪ When the application writes to the Remote Wakeup Signaling bit in the Device Control register (DCTL.RmtWkUpSig).	1'b0	RO

33.5.5.4 Device IN Endpoint Common Interrupt Mask Register (DIEPMSK)

✧ Offset: 810h

This register works with each of the Device IN Endpoint Interrupt (DIEPINTn) registers for all endpoints to generate an interrupt per IN endpoint. The IN endpoint interrupt for a specific status in the DIEPINTn register can be masked by writing to the corresponding bit in this register. Status bits are masked by default.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-47 Device IN Endpoint Common Interrupt Mask Register: DIEPMSK

Field	Description	Reset	Acces
31:14	Reserved		RO
13	NAK interrupt Mask (NAKMsk)	1'h0	R_W
12:10	Reserved		RO
9	BNA Interrupt Mask (BNAInIntrMsk) This bit is valid only when Device Descriptor DMA is enabled.	1'b0	R_W
8	Fifo Underrun Mask (TxfifoUndrnMsk)	1'b0	R_W
7	Reserved		RO
6	IN Endpoint NAK Effective Mask (INEPNakEffMsk)	1'b0	R_W
5	IN Token received with EP Mismatch Mask (INTknEPMisMsk)	1'b0	R_W
4	IN Token Received When TxFIFO Empty Mask	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	(INTknTXFEmpMsk)		
3	Timeout Condition Mask (TimeOUTMsk) (Non-isochronous endpoints)	1'b0	R_W
2	AHB Error Mask (AHBErrMsk)	1'b0	R_W
1	Endpoint Disabled Interrupt Mask (EPDisbldMsk)	1'b0	R_W
0	Transfer Completed Interrupt Mask (XferComplMsk)	1'b0	R_W

33.5.5.5 Device OUT Endpoint Common Interrupt Mask Register (DOEPMSK)

✧ Offset: 814h

This register works with each of the Device OUT Endpoint Interrupt (DOEPINTn) registers for all endpoints to generate an interrupt per OUT endpoint. The OUT endpoint interrupt for a specific status in the DOEPINTn register can be masked by writing into the corresponding bit in this register. Status bits are masked by default.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-48 Device OUT Endpoint Common Interrupt Mask Register: DOEPMSK

Field	Description	Reset	Acces
31:15	Reserved		RO
14	NYET Interrupt Mask (NYETMsk)	1'h0	R_W
13	NAK Interrupt Mask (NAKMsk)	1'h0	R_W
12	Babble Interrupt Mask (BbleErrMsk)	1'h0	R_W
11:10	Reserved		RO
9	BNA interrupt Mask (BnaOutIntrMsk)	1'h0	R_W
8	OUT Packet Error Mask (OutPktErrMsk)	1'b0	R_W
7	Reserved		RO
6	Back-to-Back SETUP Packets Received Mask (Back2BackSETup) Applies to control OUT endpoints only.	1'b0	R_W
5	Reserved		RO
4	OUT Token Received when Endpoint Disabled Mask (OUTTknEPdisMsk) Applies to control OUT endpoints only.	1'b0	R_W
3	SETUP Phase Done Mask (SetUPMsk) Applies to control endpoints only.	1'b0	R_W
2	AHB Error (AHBErrMsk)	1'b0	R_W
1	Endpoint Disabled Interrupt Mask (EPDisbldMsk)	1'b0	R_W
0	Transfer Completed Interrupt Mask (XferComplMsk)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.5.5.6 Device All Endpoints Interrupt Register (DAINT)

✧ Offset: 818h

When a significant event occurs on an endpoint, a Device All Endpoints Interrupt register interrupts the application using the Device OUT Endpoints Interrupt bit or Device IN Endpoints Interrupt bit of the Core Interrupt register (GINTSTS.OEPInt or GINTSTS.IEPInt, respectively). This is shown in Figure 33-2. There is one interrupt bit per endpoint, up to a maximum of 16 bits for OUT endpoints and 16 bits for IN endpoints.

For a bidirectional endpoint, the corresponding IN and OUT interrupt bits are used. Bits in this register are set and cleared when the application sets and clears bits in the corresponding Device Endpoint-n Interrupt register (DIEPINTn/DOEPINTn).

Table 33-49 Device All Endpoints Interrupt Register: DAINT

Field	Description	Reset	Acces
31:16	OUT Endpoint Interrupt Bits (OutEPInt) One bit per OUT endpoint: Bit 16 for OUT endpoint 0, bit 31 for OUT endpoint 15	16'h0	RO
15:0	IN Endpoint Interrupt Bits (InEpInt) One bit per IN Endpoint: Bit 0 for IN endpoint 0, bit 15 for endpoint 15	16'h0	RO

33.5.5.7 Device All Endpoints Interrupt Mask Register (DAINTMSK)

✧ Offset: 81Ch

The Device Endpoint Interrupt Mask register works with the Device Endpoint Interrupt register to interrupt the application when an event occurs on a device endpoint. However, the Device All Endpoints Interrupt (DAINT) register bit corresponding to that interrupt is still set.

✧ Mask Interrupt: 1'b0

✧ Unmask Interrupt: 1'b1

Table 33-50 Device Endpoints Interrupt Mask Register: DAINTMSK

Field	Description	Reset	Acces
31:16	OUT EP Interrupt Mask Bits (OutEpMsk) One per OUT Endpoint: Bit 16 for OUT EP 0, bit 31 for OUT EP 15 The value of this field depends on the number of OUT endpoints that are configured.	16'h0	R_W
15:0	IN EP Interrupt Mask Bits (InEpMsk) One bit per IN Endpoint: Bit 0 for IN EP 0, bit 15 for IN EP 15 The value of this field depends on the number of IN endpoints	16'h0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	that are configured.		

33.5.5.8 Device IN Token Sequence Learning Queue Read Register 1 (DTKNQR1)

✧ Offset: 820h

This register is valid only in non-periodic Shared FIFO operation (OTG_EN_DED_TX_FIFO = 0).

The queue is 4 bits wide to store the endpoint number. A read from this register returns the first 5 endpoint entries of the IN Token Sequence Learning Queue. When the queue is full, the new token is pushed into the queue and oldest token is discarded.

Table 33-51 Device IN Token Sequence Learning Queue Read Register 1: DTKNQR1

Field	Description	Reset	Acces
31:8	Endpoint Token (EPTkn) Four bits per token represent the endpoint number of the token: ▪ Bits [31:28]: Endpoint number of Token 5 ▪ Bits [27:24]: Endpoint number of Token 4 ..... ▪ Bits [15:12]: Endpoint number of Token 1 ▪ Bits [11:8]: Endpoint number of Token 0	24'h0	RO
7	Wrap Bit (WrapBit) This bit is set when the write pointer wraps. It is cleared when the learning queue is cleared.	1'b0	RO
6:5	Reserved		RO
4:0	IN Token Queue Write Pointer (INTknWPtr)	5'h0	RO

33.5.5.9 Device IN Token Sequence Learning Queue Read Register 2 (DTKNQR2)

✧ Offset: 0824h

This register is valid only in shared non-periodic Shared FIFO operation (OTG_EN_DED_TX_FIFO = 0).

A read from this register returns the next 8 endpoint entries of the learning queue.

Table 33-52 Device IN Token Sequence Learning Queue Register 2: DTKNQR2

Field	Description	Reset	Acces
31:0	Endpoint Token (EPTkn) Four bits per token represent the endpoint number of the token: ▪ Bits [31:28]: Endpoint number of Token 13 ▪ Bits [27:24]: Endpoint number of Token 12 ..... ▪ Bits [7:4]: Endpoint number of Token 7 ▪ Bits [3:0]: Endpoint number of Token 6	32'h0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.5.5.10 Device IN Token Sequence Learning Queue Read Register 3 (DTKNQR3)

✧ Offset: 0830h

This register is valid only in non-periodic Shared FIFO operation (OTG_EN_DED_TX_FIFO = 0).

A read from this register returns the next 8 endpoint entries of the learning queue.

Table 33-53 Device IN Token Sequence Learning Queue Register 3: DTKNQR3

Field	Description	Reset	Acces
31:0	Endpoint Token (EPTkn) Four bits per token represent the endpoint number of the token: ▪ Bits [31:28]: Endpoint number of Token 21 ▪ Bits [27:24]: Endpoint number of Token 20 ..... ▪ Bits [7:4]: Endpoint number of Token 15 ▪ Bits [3:0]: Endpoint number of Token 14	32'h0	RO

33.5.5.11 Device IN Token Sequence Learning Queue Read Register 4 (DTKNQR4)

✧ Offset: 0834h

This register is valid only in non-periodic Shared FIFO operation (OTG_EN_DED_TX_FIFO = 0).

A read from this register returns the last 8 endpoint entries of the learning queue.

Table 33-54 Device IN Token Sequence Learning Queue Register 4: DTKNQR4

Field	Description	Reset	Acces
31:0	Endpoint Token (EPTkn) Four bits per token represent the endpoint number of the token: ▪ Bits [31:28]: Endpoint number of Token 29 ▪ Bits [27:24]: Endpoint number of Token 28 ..... ▪ Bits [7:4]: Endpoint number of Token 23 ▪ Bits [3:0]: Endpoint number of Token 22	32'h0	RO

33.5.5.12 Device VBUS Discharge Time Register (DVBUSDIS)

✧ Offset: 0828h

This register specifies the VBUS discharge time after VBUS pulsing during SRP.

Table 33-55 Device VBUS Discharge Time Register: DVBUSDIS

Field	Description	Reset	Acces
31:16	Reserved		RO
15:0	Device VBUS Discharge Time (DVBUSDis) Specifies the VBUS discharge time after VBUS pulsing during SRP. This value equals:	30 MHz: 16'h0B8F 60 MHz: 16'h17D7	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	VBUS discharge time in PHY clocks / 1,024 The value you use depends whether the PHY is operating at 30 MHz (16-bit data width) or 60 MHz (8-bit data width). Depending on your VBUS load, this value can need adjustment.		

33.5.5.13 Device VBUS Pulsing Time Register (DVBUSPULSE)

✧ Offset: 082Ch

This register specifies the VBUS pulsing time during SRP.

Table 33-56 Device VBUS Pulsing Time Register (DVBUSPULSE)

Field	Description	Reset	Acces
31:12	Reserved		RO
11:0	Device VBUS Pulsing Time (DVBUSPulse) Specifies the VBUS pulsing time during SRP. This value equals: VBUS pulsing time in PHY clocks / 1,024 The value you use depends whether the PHY is operating at 30 MHz (16-bit data width) or 60 MHz (8-bit data width).	30 MHz: 12'h2C6 60 MHz: 12'h5B8	R_W

33.5.5.14 Device Threshold Control Register (DTHRCTL)

✧ Offset: 830h

This register is valid only for device mode in Dedicated FIFO operation (OTG_EN_DED_TX_FIFO=1). Thresholding is not supported in Slave mode and so this register must not be programmed in Slave mode.

For threshold support, the AHB must be run at 60 MHz or higher.

Table 33-57 Device Threshold Control Register (DTHRCTL)

Field	Description	Reset	Acces
31:28	Reserved		RO
27	Arbiter Parking Enable (ArbPrkEn) This bit controls internal DMA arbiter parking for IN endpoints. When thresholding is enabled and this bit is set to one, then the arbiter parks on the IN endpoint for which there is a token received on the USB. This is done to avoid getting into underrun conditions. By default the parking is enabled.	1'b1	R_W
26	Reserved		RO
25:17	Receive Threshold Length (RxThrLen) This field specifies Receive thresholding size in DWORDS. This field also specifies the amount of data received on the USB before the core can start transmitting on the AHB. The threshold length has to be at least eight DWORDS. The recommended value for ThrLen is to be the same as the programmed AHB Burst Length (GAHBCFG.HBstLen).	9'h8	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

16	Receive Threshold Enable (RxThrEn) When this bit is set, the core enables thresholding in the receive direction. Note: Synopsys recommends that you do not enable RxThrEn, because it may cause issues in the RxFIFO especially during error conditions such as RxError and Babble.	1'b0	R_W
15:13	Reserved		RO
12:11	AHB Threshold Ratio (AHBThrRatio) These bits define the ratio between the AHB threshold and the MAC threshold for the transmit path only. The AHB threshold always remains less than or equal to the USB threshold, because this does not increase overhead. Both the AHB and the MAC threshold must be DWORD-aligned. The application needs to program TxThrLen and the AHBThrRatio to make the AHB Threshold value DWORD aligned. If the AHB threshold value is not DWORD aligned, the core might not behave correctly. When programming the TxThrLen and AHBThrRatio, the application must ensure that the minimum AHB threshold value does not go below 8 DWORDS to meet the USB turnaround time requirements.	2'h0	R_W
10:2	Transmit Threshold Length (TxThrLen) This field specifies Transmit thresholding size in DWORDS. This field also forms the MAC threshold and specifies the amount of data, in bytes, to be in the corresponding endpoint transmit FIFO before the core can start a transmit on the USB. When the value of AHBThrRatio is 2'h00, the threshold length must be at least 8 DWORDS. If the AHBThrRatio is nonzero, the application must ensure that the AHB threshold value does not go below the recommended 8 DWORDs. This field controls both isochronous and non-isochronous IN endpoint thresholds. The recommended value for ThrLen is to be the same as the programmed AHB Burst Length (GAHBCFG.HBstLen).	9'h8	R_W
1	ISO IN Endpoints Threshold Enable. (ISOThrEn) When this bit is set, the core enables thresholding for isochronous IN endpoints.	1'b0	R_W
0	Non-ISO IN Endpoints Threshold Enable. (NonISOThrEn) When this bit is set, the core enables thresholding for Non Isochronous IN endpoints.	1'b0	R_W

33.5.5.15 Device IN Endpoint FIFO Empty Interrupt Mask Register: (DIEPEMPMSK)

✧ Offset: 834 h

This register is valid only in Dedicated FIFO operation (OTG_EN_DED_TX_FIFO = 1).

This register is used to control the IN endpoint FIFO empty interrupt generation (DIEPINTn.TxfEmp).

✧ Mask interrupt: 1'b0

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

✧ Unmask interrupt: 1'b1

Table 33-58 Device IN Endpoint FIFO Empty Interrupt Mask Register: DIEPEMPMSK

Field	Description	Reset	Acces
31:16	Reserved		RO
15:0	IN EP Tx FIFO Empty Interrupt Mask Bits (InEpTxfEmpMsk) These bits acts as mask bits for DIEPINTn. TxFEmp interrupt One bit per IN Endpoint: ▪ Bit 0 for IN endpoint 0 ... ▪ Bit 15 for endpoint 15	16'h0	R_W

33.5.5.16 Device Each Endpoint Interrupt Register (DEACHINT)

✧ Offset: 838h

Dependency: This register is available in device mode and only when parameter "Name: OTG_MULTI_PROC_INTRPT".

There is one interrupt bit per endpoint, up to a maximum of 16 bits for OUT endpoints and 16 bits for IN endpoints. For a bidirectional endpoint, the corresponding IN and OUT interrupt bits are used.

✧ Bits in this register are set and cleared when the application sets and clears bits in the corresponding Device Endpoint-n Interrupt register (DIEPINTn/DOEPINTn).

✧ The interrupt is automatically cleared once the DOEPINTn / DIEPINTn interrupt is cleared by the application.

Table 33-59 Device Each Endpoint Interrupt Register: DEACHINT

Field	Description	Reset	Acces
31:16	OUT Endpoint Interrupt Bits (EchOutEPInt) One bit per OUT endpoint: ▪ Bit 16 for OUT endpoint 0 ... ▪ Bit 31 for OUT endpoint 15	16'h0	RO
15:0	IN Endpoint Interrupt Bits (EchInEpInt) One bit per IN Endpoint: ▪ Bit 0 for IN endpoint 0 ... ▪ Bit 15 for IN endpoint 15	16'h0	RO

33.5.5.17 Device Each Endpoint Interrupt Register Mask (DEACHINTMSK)

✧ Offset: 83Ch

Dependency: This register is available only when parameter "Name: OTG_MULTI_PROC_INTRPT".

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The Device Each Endpoint Interrupt Mask register works with the Device Each Endpoint Interrupt register to interrupt the application when an event occurs on a device endpoint. However, the Device Each Endpoints Interrupt (DEACHINT) register bit corresponding to that interrupt remains set.

✧ Mask Interrupt: 1.b0
✧ Unmask Interrupt: 1.b1

Table 33-60 Device Each Endpoint Interrupt Register Mask: DEACHINTMSK

Field	Description	Reset	Acces
31:16	OUT EP Interrupt Mask Bits (EchOutEpMsk) One per OUT Endpoint: ▪ Bit 16 for IN endpoint 0 ... ▪ Bit 31 for IN endpoint 15	16'h0	R_W
15:0	IN EP Interrupt Mask Bits (EchInEpMsk) One bit per IN Endpoint: ▪ Bit 0 for IN endpoint 0 ... ▪ Bit 15 for IN endpoint 15	16'h0	R_W

33.5.5.18 Device Each In Endpoint-*n* Interrupt Register (DIEPEACHMSKn)

✧ Offset 840h
✧ Endpoint_number: 0=< n =< 15
✧ Offset for IN endpoints: 840h + (Endpoint_number * 4h)

Dependency: This register is available in device mode and only when parameter "Name: OTG_MULTI_PROC_INTRPT".

These registers are endpoint-specific mask registers for (DIEPINTn). The IN endpoint interrupt for a specific status in the DIEPINTn register can be masked by writing 0 to the corresponding bit in this register. Status bits are masked by default.

✧ Mask interrupt: 1'b0
✧ Unmask interrupt: 1'b1

Table 33-61 Device Each In Endpoint-n Interrupt Register: DIEPEACHMSKn

Field	Description	Reset	Acces
31:14	Reserved		RO
13	NAK interrupt Mask (NAKMsk)	1'b0	R_W
12:10	Reserved		RO
9	BNA interrupt Mask (BNAInIntrMsk)	1'b0	R_W
8	Fifo Under run Mask (TxfifoUndrnMsk)	1'b0	R_W
7	Reserved		RO
6	IN Endpoint NAK Effective Mask (INEPNakEffMsk)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

5	IN Token received with EP Mismatch Mask (INTknEPMisMsk)	1'b0	R_W
4	IN Token Received When TxFIFO Empty Mask (INTknTXFEmpMsk)	1'b0	R_W
3	Timeout Condition Mask (TimeOUTMsk) (Non-isochronous endpoints)	1'b0	R_W
2	AHB Error Mask (AHBErrMsk)	1'b0	R_W
1	Endpoint Disabled Interrupt Mask (EPDisbldMsk)	1'b0	R_W
0	Transfer Completed Interrupt Mask (XferComplMsk)	1'b0	R_W

33.5.5.19 Device Each Out Endpoint-*n* Interrupt Register (DOEPEACHMSKn)

✧ Offset 880h

✧ Endpoint_number: 0 =< n =< 15

✧ Offset for OUT endpoints: 880h + (Endpoint_number * 4h)

Dependency: This register is available in device mode and only when parameter OTG_MULTI_PROC_INTRPT=1.

These registers are endpoint specific mask registers for (DOEPINTn). The OUT endpoint interrupt for a specific status in the DOEPINTn register can be masked by writing 0 to the corresponding bit in this register. Status bits are masked by default.

✧ Mask interrupt: 1'b0

✧ Unmask interrupt: 1'b1

Table 33-62 Device Each Out Endpoint-n Interrupt Register: DOEPEACHMSKn

Field	Description	Reset	Acces
31:15	Reserved		RO
14	NYET interrupt Mask (NYETMsk)	1'b0	R_W
13	NAK interrupt Mask (NAKMsk)	1'b0	R_W
12	Babble interrupt Mask (BbleErrMsk)	1'b0	R_W
11:10	Reserved		RO
9	BNA interrupt Mask (BnaOutIntrMsk)	1'b0	R_W
8	OUT Packet Error Mask (OutPktErrMsk)	1'b0	R_W
7	Reserved		RO
6	Back-to-Back SETUP Packets Received Mask (Back2BackSETup) Applies to control OUT endpoints only.	1'b0	R_W
5	Reserved		RO
4	OUT Token Received when Endpoint Disabled Mask (OUTTknEPdisMsk) Applies to control OUT endpoints only.	1'b0	R_W
3	SETUP Phase Done Mask (SetUPMsk) Applies to control endpoints only.	1'b0	R_W
2	AHB Error (AHBErrMsk)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	Endpoint Disabled Interrupt Mask (EPDisbldMsk)	1'b0	R_W
0	Transfer Completed Interrupt Mask (XferComplMsk)	1'b0	R_W

33.5.5.20 Device Control IN Endpoint 0 Control Register (DIEPCTL0)

✧ Offset: 900h

This section describes the Control IN Endpoint 0 Control register. Nonzero control endpoints use registers for endpoints 1-15.

Table 33-63 Device Control IN Endpoint 0 Control Register: DIEPCTL0

Field	Description	Reset	Acces
31	Endpoint Enable (EPEna) ▪ When Scatter/Gather DMA mode is enabled, for IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. ▪ When Scatter/Gather DMA mode is disabled—such as in buffer-pointer based DMA mode—this bit indicates that data is ready to be transmitted on the endpoint. The core clears this bit before setting the following interrupts on this endpoint: ▪ Endpoint Disabled ▪ Transfer Completed	1'b0	R_WS_SC
30	Endpoint Disable (EPDis) The application sets this bit to stop transmitting data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled Interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint. This bit is valid only when DMA mode is enabled.	1'b0	R_WS_SC
29:28	Reserved		RO
27	Set NAK (SNAK) A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for an endpoint after a SETUP packet is received on that endpoint.	1'b0	WO
26	Clear NAK (CNAK) A write to this bit clears the NAK bit for the endpoint.	1'b0	WO
25:22	TxFIFO Number (TxFNum) For Shared FIFO operation, this value is always set to 0, indicating that control IN endpoint 0 data is always written in the Non-Periodic Transmit FIFO. For Dedicated FIFO operation, this value is set to the FIFO number that is	4'h0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	assigned to IN Endpoint 0.		
21	STALL Handshake (Stall) The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority.	1'b0	R_WS_SC
20	Reserved		RO
19:18	Endpoint Type (EPType) Hardcoded to 00 for control.	2'h0	RO
17	NAK Status (NAKSts) Indicates the following: 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status 1'b1: The core is transmitting NAK handshakes on this endpoint. When this bit is set, either by the application or core, the core stops transmitting data, even if there is data available in the TxFIFO. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.	1'b0	RO
16	Reserved		RO
15	USB Active Endpoint (USBActEP) This bit is always set to 1, indicating that control endpoint 0 is always active in all configurations and interfaces.	1'b1	RO
14:11	Next Endpoint (NextEp) Applies to non-periodic IN endpoints only. Indicates the endpoint number to be fetched after the data for the current endpoint is fetched. The core can access this field, even when the Endpoint Enable (EPEna) bit is not set. This field is not valid in Slave mode. **Note**: This field is valid only for Shared FIFO operations.	4'b0	R_W
10:2	Reserved		RO
1:0	Maximum Packet Size (MPS) Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint.	2'h0	R_W

33.5.5.21 Device Control OUT Endpoint 0 Control Register (DOEPCTL0)

✧ Offset: B00h

This section describes the Control OUT Endpoint 0 Control register. Nonzero control endpoints use registers for endpoints 1-15.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 33-64 Device OUT Endpoint 0 Control Register: DOEPCTL0

Field	Description	Reset	Acces
31	Endpoint Enable (EPEna) When Scatter/Gather DMA mode is enabled, for OUT endpoints this bit indicates that the descriptor structure and data buffer to receive data is setup. ▪ When Scatter/Gather DMA mode is disabled—(such as for buffer-pointer based DMA mode)—this bit indicates that the application has allocated the memory to start receiving data from the USB. The core clears this bit before setting any of the following interrupts on this endpoint: ▪ SETUP Phase Done ▪ Endpoint Disabled ▪ Transfer Completed **Note**: In DMA mode, this bit must be set for the core to transfer SETUP data packets into memory.	1'b0	R_WS_SC
30	Endpoint Disable (EPDis) The application cannot disable control OUT endpoint 0.	1'b0	RO
29:28	Reserved		RO
27	Set NAK (SNAK) A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set bit on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.	1'b0	WO
26	Clear NAK (CNAK) A write to this bit clears the NAK bit for the endpoint.	1'b0	WO
25:22	Reserved		RO
21	STALL Handshake (Stall) The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.	1'b0	R_WS_SC
20	Snoop Mode (Snp) This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.	1'b0	R_W
19:18	Endpoint Type (EPType) Hardcoded to 2'b00 for control.	2'h0	RO
17	NAK Status (NAKSts) Indicates the following:	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit, the core stops receiving data, even if there is space in the RxFIFO to accommodate the incoming packet. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.		
16	Reserved		RO
15	USB Active Endpoint (USBActEP) This bit is always set to 1, indicating that a control endpoint 0 is always active in all configurations and interfaces.	1'b1	RO
14:2	Reserved		RO
1:0	Maximum Packet Size (MPS) The maximum packet size for control OUT endpoint 0 is the same as what is programmed in control IN Endpoint 0. ▪ 2'b00: 64 bytes ▪ 2'b01: 32 bytes ▪ 2'b10: 16 bytes ▪ 2'b11: 8 bytes	2'h0	RO

33.5.5.22 Device Endpoint-*n* Control Register (DIEPCTLn/DOEPCTLn)

✧ Endpoint_number: 1 ≤n ≤15

✧ Offset for IN endpoints: 900h + (Endpoint_number * 20h)

✧ Offset for OUT endpoints: B00h + (Endpoint_number * 20h)

The application uses this register to control the behavior of each logical endpoint other than endpoint 0.

Table 33-65 Device Endpoint-n Control Register: DIEPCTLn/DOEPCTLn

Field	Description	Reset	Acces
31	Endpoint Enable (EPEna) Applies to IN and OUT endpoints. ▪ When Scatter/Gather DMA mode is enabled, ▪ For IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. ▪ For OUT endpoint it indicates that the descriptor structure and data buffer to ▪ receive data is setup. ▪ When Scatter/Gather DMA mode is enabled—such as for buffer-pointer based DMA mode: – For IN endpoints, this bit indicates that data is ready to be transmitted on the endpoint.	1'b0	R_WS_SC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	– For OUT endpoints, this bit indicates that the application has allocated the memory to start receiving data from the USB. – The core clears this bit before setting any of the following interrupts on this endpoint. ▪ SETUP Phase Done ▪ Endpoint Disabled ▪ Transfer Completed **Note**: For control endpoints in DMA mode, this bit must be set to be able to transfer SETUP data packets in memory.		
30	Endpoint Disable (EPDis) Applies to IN and OUT endpoints. The application sets this bit to stop transmitting/receiving data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.	1'b0	R_WS_SC
29	Set DATA1 PID (SetD1PID) Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA1. This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. Set Odd (micro)frame (SetOddFr) Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro)frame (EO_FrNum) field to odd (micro)frame. This field is not applicable for Scatter/Gather DMA mode.	1'b0	WO WO
28	Set DATA0 PID (SetD0PID) Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA0. This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. In non-Scatter/Gather DMA mode: Set Even (micro)frame (SetEvenFr) Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro)frame (EO_FrNum) field to even (micro) frame. When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is in the transmit descriptor structure. The frame in which to receive data is updated in receive descriptor	1'b0	WO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	structure.		
27	Set NAK (SNAK) Applies to IN and OUT endpoints. A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for OUT endpoints on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.	1'b0	WO
26	Clear NAK (CNAK) Applies to IN and OUT endpoints. A write to this bit clears the NAK bit for the endpoint.	1'b0	WO
25:22	TxFIFO Number (TxFNum) ***Shared FIFO Operation***—non-periodic endpoints must set this bit to zero. Periodic endpoints must map this to the corresponding Periodic TxFIFO number. ▪ 4'h0: Non-Periodic TxFIFO ▪ Others: Specified Periodic TxFIFO.number **Note**: An interrupt IN endpoint can be configured as a non-periodic endpoint for applications such as mass storage. The core treats an IN endpoint as a non-periodic endpoint if the TxFNum field is set to 0. Otherwise, a separate periodic FIFO must be allocated for an interrupt IN endpoint, and the number of this FIFO must be programmed into the TxFNum field. Configuring an interrupt IN endpoint as a non-periodic endpoint saves the extra periodic FIFO area. ***Dedicated FIFO Operation***—these bits specify the FIFO number associated with this endpoint. Each active IN endpoint must be programmed to a separate FIFO number. This field is valid only for IN endpoints.	4'h0	R_W
21	STALL Handshake (Stall) Applies to non-control, non-isochronous IN and OUT endpoints only. The application sets this bit to stall all tokens from the USB host to this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Only the application can clear this bit, never the core. Applies to control endpoints only. The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.	1'h0	R_W R_WS_SC
20	Snoop Mode (Snp)	1'b0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Applies to OUT endpoints only. This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.		
19:18	Endpoint Type (EPType) Applies to IN and OUT endpoints. This is the transfer type supported by this logical endpoint. ▪ 2'b00: Control ▪ 2'b01: Isochronous ▪ 2'b10: Bulk ▪ 2'b11: Interrupt	2'h0	R_W
17	NAK Status (NAKSts) Applies to IN and OUT endpoints. Indicates the following: ▪ 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. ▪ 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit: The core stops receiving any data on an OUT endpoint, even if there is space in the RxFIFO to accommodate the incoming packet. For non-isochronous IN endpoints: The core stops transmitting any data on an IN endpoint, even if there data is available in the TxFIFO. For isochronous IN endpoints: The core sends out a zero-length data packet, even if there data is available in the TxFIFO. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.	1'b0	RO
16	Endpoint Data PID (DPID) Applies to interrupt/bulk IN and OUT endpoints only. Contains the PID of the packet to be received or transmitted on this endpoint. The application must program the PID of the first packet to be received or transmitted on this endpoint, after the endpoint is activated. The applications use the SetD1PID and SetD0PID fields of this register to program either DATA0 or DATA1 PID. ▪ 1'b0: DATA0 ▪ 1'b1: DATA1 This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. Even/Odd (Micro)Frame (EO_FrNum) In non-Scatter/Gather DMA mode: Applies to isochronous IN and OUT endpoints only. Indicates the (micro)frame number in which the core transmits/receives isochronous data for this endpoint. The application must program the	1'b0	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	even/odd (micro) frame number in which it intends to transmit/receive isochronous data for this endpoint using the SetEvnFr and SetOddFr fields in this register. ▪ 1'b0: Even (micro)frame ▪ 1'b1: Odd (micro)frame When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is provided in the transmit descriptor structure. The frame in which data is received is updated in receive descriptor structure.		
15	USB Active Endpoint (USBActEP) Applies to IN and OUT endpoints. Indicates whether this endpoint is active in the current configuration and interface. The core clears this bit for all endpoints (other than EP 0) after detecting a USB reset. After receiving the SetConfiguration and SetInterface commands, the application must program endpoint registers accordingly and set this bit.	1'b0	R_W_SC
14:11	Next Endpoint (NextEp) Applies to non-periodic IN endpoints only. Indicates the endpoint number to be fetched after the data for the current endpoint is fetched. The core can access this field, even when the Endpoint Enable (EPEna) bit is low. This field is not valid in Slave mode operation. **Note**: This field is valid only for Shared FIFO operations.	4'b0	R_W
10:0	Maximum Packet Size (MPS) Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current	11'h0	R_W

33.5.5.23 Device Endpoint-*n* Interrupt Register (DIEPINTn/DOEPINTn)

✧ Endpoint_number: 0 ≤n ≤15
✧ Offset for IN endpoints: 908h + (Endpoint_number * 20h)
✧ Offset for OUT endpoints: B08h + (Endpoint_number * 20h)

This register indicates the status of an endpoint with respect to USB- and AHB-related events. It is shown in Figure 33-2. The application must read this register when the OUT Endpoints Interrupt bit or IN Endpoints Interrupt bit of the Core Interrupt register (GINTSTS.OEPInt or GINTSTS.IEPInt, respectively) is set. Before the application can read this register, it must first read the Device All Endpoints Interrupt (DAINT) register to get the exact endpoint number for the Device Endpoint-n Interrupt register. The application must clear the appropriate bit in this register to clear the corresponding bits in the DAINT and GINTSTS registers.

Table 33-66 Device Endpoint-n Interrupt Register: DIEPINTn/DOEPINTn

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Field	Description	Reset	Acces
31:15	Reserved		RO
14	NYET interrupt (NYETIntrpt) The core generates this interrupt when a NYET response is transmitted for a non isochronous OUT endpoint.	1'b0	R_SS_WC
13	NAK interrupt (NAKIntrpt) The core generates this interrupt when a NAK is transmitted. In case of isochronous IN endpoints the interrupt gets generated when a zero length packet is transmitted due to unavailability of data in the TXFifo.	1'b0	R_SS_WC
12	BbleErr (Babble Error) interrupt (BbleErrIntrpt) The core generates this interrupt when babble is received for the endpoint.	1'b0	R_SS_WC
11	PktDrpSts (Packet Dropped Status) This bit indicates to the application that an ISOC OUT packet has been dropped. This bit does not have an associated mask bit and does not generate an interrupt. **Dependency**: This bit is valid in non Scatter/Gather DMA mode when periodic transfer interrupt feature is selected.	1'b0	R_SS_WC
10	Reserved		RO
9	BNA (Buffer Not Available) Interrupt (BNAIntr) The core generates this interrupt when the descriptor accessed is not ready for the Core to process, such as Host busy or DMA done. **Dependency**: This bit is valid only when Scatter/Gather DMA mode is enabled.	1'b0	R_SS_WC
8	FIFO Underrun (TxfifoUndrn) Applies to IN endpoints only The core generates this interrupt when it detects a transmit FIFO underrun condition for this endpoint. **Dependency**: This interrupt is valid only when both of the following conditions are true: ▪ Parameter OTG_EN_DED_TX_FIFO=1 ▪ Thresholding is enabled OUT Packet Error (OutPktErr) Applies to OUT endpoints only This interrupt is asserted when the core detects an overflow or a CRC error for an OUT packet. **Dependency**: This interrupt is valid only when both of the following conditions are true: ▪ Parameter OTG_EN_DED_TX_FIFO=1 ▪ Thresholding is enabled.	1'b0	R_SS_WC R_SS_WC
7	Transmit FIFO Empty (TxFEmp) This bit is valid only for IN Endpoints This interrupt is asserted when the TxFIFO for this endpoint is either half or completely empty. The half or completely empty status is determined by	1'b1	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	the TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl)).		
6	N Endpoint NAK Effective (INEPNakEff) This bit should be cleared by writing a 1'b1 before writing a 1'b1 to corresponding DIEPCTLn.CNAK. The interrupt indicates that the IN endpoint NAK bit set by the application has taken effect in the core. This interrupt does not guarantee that a NAK handshake is sent on the USB. A STALL bit takes priority over a NAK bit. This bit is applicable only when the endpoint is enabled. Back-to-Back SETUP Packets Received (Back2BackSETup) Applies to Control OUT endpoints only. This bit indicates that the core has received more than three back-to-back SETUP packets for this particular endpoint. This bit is not valid in Slave mode.	1'b0 1'b0	R_SS_WC R_SS_WC
5	IN Token Received with EP Mismatch (INTknEPMis) Applies to non-periodic IN endpoints only. Indicates that the data in the top of the non-periodic TxFIFO belongs to an endpoint other than the one for which the IN token was received. This interrupt is asserted on the endpoint for which the IN token was received. Status Phase Received For Control Write (StsPhseRcvd) This interrupt is valid only for Control OUT endpoints and only in Scatter Gather DMA mode. This interrupt is generated only after the core has transferred all the data that the host has sent during the data phase of a control write transfer, to the system memory buffer. The interrupt indicates to the application that the host has switched from data phase to the status phase of a Control Write transfer. The application can use this interrupt to ACK or STALL the Status phase, after it has decoded the data phase. This is applicable only in case of Scatter Gather DMA mode.	1'b0	R_SS_WC
4	IN Token Received When TxFIFO is Empty (INTknTXFEmp) Indicates that an IN token was received when the associated TxFIFO (periodic/non- periodic) was empty. This interrupt is asserted on the endpoint for which the IN token was received. OUT Token Received When Endpoint Disabled (OUTTknEPdis) Indicates that an OUT token was received when the endpoint was not yet enabled. This interrupt is asserted on the endpoint for which the OUT token was received.	1'b0	R_SS_WC

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

3	Timeout Condition (TimeOUT) ▪ In shared TX FIFO mode, applies to non-isochronous IN endpoints only. ▪ In dedicated FIFO mode, applies only to Control IN endpoints. ▪ In Scatter/Gather DMA mode, the TimeOUT interrupt is not asserted. Indicates that the core has detected a timeout condition on the USB for the last IN token on this endpoint. SETUP Phase Done (SetUp) Applies to control OUT endpoints only. Indicates that the SETUP phase for the control endpoint is complete and no more back-to-back SETUP packets were received for the current control transfer. On this interrupt, the application can decode the received SETUP data packet.	1'b0	R_SS_WC
2	AHB Error (AHBErr) Applies to IN and OUT endpoints. This is generated only in Internal DMA mode when there is an AHB error during an AHB read/write. The application can read the corresponding endpoint DMA address register to get the error address.	1'b0	R_SS_WC
1	Endpoint Disabled Interrupt (EPDisbld) Applies to IN and OUT endpoints. This bit indicates that the endpoint is disabled per the application's request.	1'b0	R_SS_WC
0	Transfer Completed Interrupt (XferCompl) Applies to IN and OUT endpoints. ▪ When Scatter/Gather DMA mode is enabled ▪ For IN endpoint this field indicates that the requested data from the descriptor is moved from external system memory to internal FIFO. ▪ For OUT endpoint this field indicates that the requested data from the internal FIFO is moved to external system memory. This interrupt is generated only when the corresponding endpoint descriptor is closed, and the IOC bit for the corresponding descriptor is set. ▪ When Scatter/Gather DMA mode is disabled, this field indicates that the programmed transfer is complete on the AHB as well as on the USB, for this endpoint.	1'b0	R_SS_WC

33.5.5.24 Device Endpoint 0 Transfer Size Register (DIEPTSIZ0/DOEPTSIZ0)

✧ Offset for IN endpoints: 910h

✧ Offset for OUT endpoints: B10h

The application must modify this register before enabling endpoint 0. Once endpoint 0 is enabled using Endpoint Enable bit of the Device Control Endpoint 0 Control registers (DIEPCTL0.EPEna/DOEPCTL0.EPEna), the core modifies this register. The application can only read thisregister once the core has cleared the Endpoint Enable bit.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Nonzero endpoints use the registers for endpoints 1-15.

When Scatter/Gather DMA mode is enabled, this register must not be programmed by the application. If the application reads this register when Scatter/Gather DMA mode is enabled, the core returns all zeros.

Table 33-67 Device IN Endpoint 0 Transfer Size Register: DIEPTSIZ0

Field	Description	Reset	Acces
31:21	Reserved		RO
20:19	Packet Count (PktCnt) Indicates the total number of USB packets that constitute the Transfer Size amount of data for endpoint 0. This field is decremented every time a packet (maximum size or short packet) is read from the TxFIFO.	2'b0	R_W
18:7	Reserved		RO
6:0	Transfer Size (XferSize) Indicates the transfer size in bytes for endpoint 0. The core interrupts the application only after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. The core decrements this field every time a packet from the external memory is written to the TxFIFO.	7'h0	R_W

Table 33-68 Device OUT Endpoint 0 Transfer Size Register: DOEPTSIZ0

Field	Description	Reset	Acces
31	Reserved		RO
30:29	SETUP Packet Count (SUPCnt) This field specifies the number of back-to-back SETUP data packets the endpoint can receive. ▪ 2'b01: 1 packet ▪ 2'b10: 2 packets ▪ 2'b11: 3 packets	2'h0	R_W
28:20	Reserved		RO
19	Packet Count (PktCnt) This field is decremented to zero after a packet is written into the RxFIFO.	1'b0	R_W
18:7	Reserved		RO
6:0	Transfer Size (XferSize) Indicates the transfer size in bytes for endpoint 0. The core interrupts the application only after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. The core decrements this field every time a packet is read from the RxFIFO and written to the external memory.	7'h0	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

33.5.5.25 Device Endpoint-*n* Transfer Size Register (DIEPTSIZn/DOEPTSIZn)

✧ Endpoint_number: 1 ≤n ≤15
✧ Offset for IN endpoints: 910h + (Endpoint_number * 20h)
✧ Offset for OUT endpoints: B10h + (Endpoint_number * 20h)

The application must modify this register before enabling the endpoint. Once the endpoint is enabled using Endpoint Enable bit of the Device Endpoint-n Control registers (DIEPCTLn.EPEna/DOEPCTLn.EPEna), the core modifies this register. The application can only read this register once the core has cleared the Endpoint Enable bit.

This register is used only for endpoints other than Endpoint 0.

Table 33-69 Device Endpoint-n Transfer Size Register: DIEPTSIZn/DOEPTSIZn

Field	Description	Reset	Acces
31	Reserved		RO
30:29	Multi Count (MC) Applies to IN endpoints only. For periodic IN endpoints, this field indicates the number of packets that must be transmitted per microframe on the USB. The core uses this field to calculate the data PID for isochronous IN endpoints. ▪ 2'b01: 1 packet ▪ 2'b10: 2 packets ▪ 2'b11: 3 packets	2'b0	R_W
	For non-periodic IN endpoints, this field is valid only in Internal DMA mode. It specifies the number of packets the core must fetch for an IN endpoint before it switches to the endpoint pointed to by the Next Endpoint field of the Device Endpoint-*n* Control register (DIEPCTLn.NextEp).		RO
	Received Data PID (RxDPID) Applies to isochronous OUT endpoints only. This is the data PID received in the last packet for this endpoint. ▪ 2'b00: DATA0 ▪ 2'b01: DATA2 ▪ 2'b10: DATA1 ▪ 2'b11: MDATA		RO
	SETUP Packet Count (SUPCnt) Applies to control OUT Endpoints only. This field specifies the number of back-to-back SETUP data packets the endpoint can receive. ▪ 2'b01: 1 packet ▪ 2'b10: 2 packets ▪ 2'b11: 3 packets		R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

28:19	Packet Count (PktCnt) Indicates the total number of USB packets that constitute the Transfer Size amount of data for this endpoint (PktCnt = XferSize / MPS). ▪ IN Endpoints: This field is decremented every time a packet (maximum size or short packet) is read from the TxFIFO. ▪ OUT Endpoints: This field is decremented every time a packet (maximum size or short packet) is written to the RxFIFO.	10'h0	R_W
18:0	Transfer Size (XferSize) This field contains the transfer size in bytes for the current endpoint. The transfer size (XferSize) = Sum of buffer sizes across all descriptors in the list for the endpoint. In Buffer DMA, the core only interrupts the application after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. ▪ IN Endpoints: The core decrements this field every time a packet from the external memory is written to the TxFIFO. ▪ OUT Endpoints: The core decrements this field every time a packet is read from the RxFIFO and written to the external memory.	19'h0	R_W

Note: Note for Descriptor DMA

The maximum transfer size supported is 219 bytes

The maximum packet count supported = XferSize / Max Packet Size

The Transfer Size can span across multiple descriptors

33.5.5.26 Device Endpoint-*n* DMA Address Register (DIEPDMAn/DOEPDMAn)

✧ Endpoint_number: 0 ≤n ≤15

✧ Offset for IN endpoints: 914h + (Endpoint_number * 20h)

✧ Offset for OUT endpoints: B14h + (Endpoint_number * 20h)

These registers are implemented in RAM instead of flop-based implementation.

Table 33-70 Device Endpoint-n DMA Address Register: DIEPDMAn/DOEPDMAn

Field	Description	Reset	Acces
31:0	DMA Address (DMAAddr) Holds the start address of the external memory for storing or fetching endpoint data. **Note**: For control endpoints, this field stores control OUT data packets as well as SETUP transaction data packets. When more than three SETUP packets are received back-to-back, the SETUP data packet in the memory is overwritten. This register is incremented on every AHB transaction. The application can give only a DWORD-aligned address.	"X" if not programmed as the register is in SPRAM	R_W

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ When Scatter/Gather DMA mode is not enabled, the application programs the start address value in this field. ▪ When Scatter/Gather DMA mode is enabled, this field indicates the base pointer for the descriptor list.		

33.5.5.27 Device Endpoint-*n* DMA Buffer Address Register (DIEPDMABn/DOEPDMABn)

✧ Endpoint_number: 0 ≤ n ≤15
✧ Offset for IN endpoints: 91Ch + (Endpoint_number * 20h)
✧ Offset for OUT endpoints: B1Ch + (Endpoint_number * 20h)

These fields are present only in case of Scatter/Gather DMA. These registers are implemented in RAM instead of flop-based implementation.

Table 33-71 Device Endpoint-n DMA Buffer Address Register: DIEPDMABn/DOEPDMABn

Field	Description	Reset	Acces
31:0	DMA Buffer Address (DMABufferAddr) Holds the current buffer address.This register is updated as and when the data transfer for the corresponding end point is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.	"X" if not programmed as the register is in SPRAM	R_O

33.5.5.28 Device IN Endpoint Transmit FIFO Status Register (DTXFSTSn)

✧ Endpoint_number: 0 ≤ n ≤15
✧ Offset for IN endpoints: 918h + (Endpoint_number * 20h)

This read-only register contains the free space information for the Device IN endpoint TxFIFO.

Table 33-72 Device IN Endpoint Transmit FIFO Status Register: DTXFSTSn

Field	Description	Reset	Acces
31:16	Reserved.		RO
15:0	IN Endpoint TxFIFO Space Avail (INEPTxFSpcAvail) Indicates the amount of free space available in the Endpoint TxFIFO. Values are in terms of 32-bit words. ▪ 16'h0: Endpoint TxFIFO is full ▪ 16'h1: 1 word available ▪ 16'h2: 2 words available ▪ 16'hn: n words available (where 0 ≤n ≤32,768) ▪ 16'h8000: 32,768 words available ▪ Others: Reserved	Configurable	RO

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 7

BOOT

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

34 XBurst Boot ROM Specification

The M200 contains an internal 32KB boot ROM. The CPU boots from the boot ROM after reset.

34.1 Boot Select

The boot sequence of the M200 is controlled by boot_sel [2:0]. The configuration is shown as follow:

Table 34-1 Boot Configuration of M200

boot_sel[2:0]	Boot method
111	USB boot @ USB 2.0 device, EXTCLK=24MHz
110	NAND boot @ CS1
101	MMC/SD boot @ MSC0 (MMC/SD use GPIO Port A)
100	MMC/SD boot @ MSC1 (MMC/SD use GPIO Port E)
011	eMMC boot @ MSC0 (use GPIO Port A)
010	NOR boot @ CS2 (just for FPGA testing)
001	USB boot @ USB 2.0 device, EXTCLK=26MHz
000	SPI boot @ SPI0/CE0

34.2 Boot Procedure

After reset, the boot program on the internal boot ROM executes as follows:

1. Disable all interrupts and read boot_sel[2:0] to determine the boot method.
2. If it is boot from NAND flash, one flag at the beginnig of NAND are read to know the NAND information including Bus width, NAND type, Page cycle(2 or 3 cycles) and its Page size(512B, 2KB, 4KB 8KB or 16KB). Then 26KB code are read out from NAND to cache, if the 26KB reading failed, the next 26KB backup in NAND will be read. Then branch to cache at 512 bytes offset.
3. There 26KB backup reading failed, the 26KB backup at 128th, 256 th , …, and finally 1024th page will be tried in consecutive order.
4. If it is boot from MMC/SD card at MSC0, its function pins MSC0_D0, MSC0_CLK, MSC0_CMD are initialized, the boot program loads the 26KB code from MMC/SD card to cache and jump to it. Only one data bus which is MSC0_D0 is used.
5. If it is boot from eMMC boot partition1 at MSC0, its function pins MSC0_D0, MSC0_CLK, MSC0_CMD are initialized, the boot program loads the 26KB code from eMMC boot partition1 to cache and jump to it. Only one data bus which is MSC0_D0 is used.
6. If it is boot from USB, a block of code will be received through USB cable connected with host PC and be stored in cache. Then branch to this area in cache.
7. If it is boot from MMC/SD card at MSC1, its function pins MSC1_D0, MSC1_CLK, MSC1_CMD are initialized, the boot program loads the 26KB code from MMC/SD card to cache and jump to it. Only one data bus which is MSC1_D0 is used.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: The M200's cache is 32KB, mapped address is from 0x80000000 to 0x80008000 in bootrom。

Figure 34-1 Boot sequence diagram of M200

34.3 SPL Structure

For some special requirement, SPL(Second Program Loader) data needs to contain different information according to the different boot modes. The basic structure as shown below. SC_Boot KEY (1.5K) use for security boot, SPL signature is used to store some information for kinds of boot (512B), and SPL.text (not more than 24K,and 512 byte aligned) is real code segment. he first 4k bytes of address space is used for stack and data

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

space. Except NOR boot , all the boot model must jump to 0x80001800 to execute SPL code and all the boot model contains the this SPL structure show as Figure 41 -2. Every boot model has its unique SPL signature, Introduces the SPL signature structure behind.

Figure 34-2 The basic SPL structure

34.4 SPL Parameters

In order to support variable length of SPL size and increase the loading speed by changing the clock frequency. The SPL parameter is created to accomplish this function. This mechanism is only used for NAND/SD/MMC boot, and this mechanism is optional.

Parameters in a storage space in a certain structure is placed, and the size is 256 bytes .When sd card boot, the parameters will be placed at first 256 bytes space of the MBR. When NAND boot, the parameters will be placed at the end of the SPL signature. SPL Parameters's location and structure show below:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-3 SPL parameters location

Table 34-2 SPL parameters structure

Offset	0x1 : 0x0	0x3 : 0x2	0x7 : 0x4	0xb : 0x8	0xf : 0xc
0x00	INGE		SPL length	CPU freq	Reserved
0x10	NAND Timing[0]		NAND Timing[1]	NAND Timing[2]	NAND Timing[3]
0x20	Desc[0].saddr	Desc[0].paddr	Desc[0].value	Desc[0].poll_h	Desc[0].poll_l
0x30	Desc[1].saddr	Desc[1].paddr	Desc[1].value	Desc[1].poll_h	Desc[1].poll_l
0x40	Desc[2].saddr	Desc[2].paddr	Desc[2].value	Desc[2].poll_h	Desc[2].poll_l
…	…	…	…	…	…
0xf0	Desc[13].saddr	Desc[13].paddr	Desc[13].value	Desc[13].poll_h	Desc[13].poll_l

Domain in SPL parameter structure:

INGE：

SPL Params ID，must store ‘I’ ‘N’ ‘G’ ‘E’, if this area is not store “INGE”,bootrom will use default params to boot.

SPL length：

The length of SPL in bytes, and must be 512 byte aligned. the size of SPL must be not more than 26K.

CPU freq：

Cpu frequency conversion value in HZ.

NAND Timing[0~3]：

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

After frequency conversion, the timing parameters of NAND.

```
struct {
        unsigned set_rw:8;
        unsigned wait_rw:8;
        unsigned hold_rw:8;
        unsigned set_cs:8;
        unsigned wait_cs:8;
        unsigned trr:8;
        unsigned tedo:8;
        unsigned trpre:8;
        unsigned twpre:8;
        unsigned tds:8;
        unsigned tdh:8;
        unsigned twpst:8;
        unsigned tdqsre:8;
        unsigned trhw:8;
        unsigned t1:8;
        unsigned t2:8;
    };
```

Descriptor：

A maximum of 14 groups descriptor, each descriptor is used to configure an cpm register.

The content of descriptor definition is as follows:

Saddr: Set Value Addr(This is lower 16 bits address , th higher 16bit address is 0xb000).

paddr: Polling Value Addr(This is lower 16 bits address , th higher 16bit address is 0xb000).

value: The configuration value of the “saddr” register

poll_h: Polling high bit mask(32bit).

poll_l: Polling low bit mask(32bit).

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-4 SPL configuration change Procedure

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

34.5 NAND Boot Specification

If CPU boots from NAND flash (CS1), the boot ROM will read one flag, that is first 512bytes from SPL which is stored in NAND flash, to know the NAND information including row cycle(2 or 3), its page size(512, 2KB, 4KB, 8KB or 16KB bytes) and the timings of CPM controller and NFI controller.

The content and definition of the flag are shown as follow:

Table 34-3 The definition of the flag in NAND flash

Name	Location (in byte)	length (in byte)	Value	Description
BusWidth_flag	0 - 63	64	0x55 or 0xaa	Bus width 0x55: 8 bit NAND 0xaa: 16 bit NAND
NANDType_flag	64 - 127	64	0x55 or 0xaa	NAND type. 0x55: common NAND 0xaa: toggle NAND
Other_flag	128 - 159	32		SPLflag: 4bytes Rowcycle : 4bytes Pagesize: 4bytes Free: 20bytes
reserved	160 - 255	96		
Timing_flag	256 - 512	256		CPM's timing : 16bytes NFI's timing : 16bytes Free: 224bytes

The BusWidth_flag containing 64 bytes locates at the beginning of NAND, if the Bus Width of NAND is 8 bit, the BusWidth_flag should be filled with 0x55 for all 64 bytes, or else it should be filled with 0xaa;The NANDType_flag containing 64 bytes locates behind the BusWidth_flag, if the type of NAND is common, the NANDType_flag should be filled with 0x55 for all 64 bytes, or else it should be filled with 0xaa;The Other_flag containing 32 bytes locates behind the NANDType_flag, the first 4bytes store SPLflag, the value should be 0x214c5053('SPL!'); the following 8bytes store the value rowcycle and pagesize, while the next 20bytes is free. Pay attention to the next 96 bytes is reserved. the space, which offset 256bytes of the beginning of NAND, stores timing of CPM and NFI controller.

At first, the first 256 bytes (which is a PN* unit) in NAND containing the flag will be read out to a buffer assuming the Bus Width is 8 bit and the NAND Type is common. The BusWidth_flag will be get from the buffer to detect the Bus Width. If there is no 0x55 or 0xaa in BusWidth_flag, 128th, 256th, ..., 1024th page will be tried in sequence. If failed at 1024th page, bootrom will jump to sd_boot. If BusWidth_flag is valid,the NANDType_flag will be obtained from the buffer to know the type of NAND.then,the first 512bytes will be read out again, assume the rowcycle, pagesize and timing of CPM and NFI.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

26KB codes in NAND will be loaded up to cache and branch to it at 512 bytes offset. Hardware PN and 64-bit BCH ECC will be used for every 256 bytes during reading. The ECC(112 bytes per 256 bytes data) stores in the data area of a NAND page behind the page storing code data. If no ECC error is detected or ECC error is correctable(number of error bits <= 64), NAND boots successfully. If uncorrectable error occurred, next 26KB backup at 128 pages behind will be tried. 128th, 256th, ..., 1024th page will be tried in sequence. If failed at 1024th page, bootrom will jump to sd boot(msc1).

The distribution and structure of the boot code in NAND and the procedure of the M200 NAND boot is shown as blow

NOTE: PN is short for pseudorandom noise which is used for supporting TLC (three-level cell) NAND.

Figure 34-5 the distribution and structure of the boot code in NAND

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-6 M200 NAND Boot Procedure

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

34.6 USB Boot Specification

When boot_sel[2:0] is selected as USB boot, the internal boot ROM downloads user program from the USB port to internal SRAM and branches to the internal SRAM to execute the program.

M200 supports the external main crystal whose frequencys are 12MHz,19.2M,24M,26M,48M.
The boot program supports both high-speed (480MHz) and full-speed (12MHz) transfer modes. The boot program uses the following two transfer types.

Table 34-4 Transfer Types Used by the Boot Program

Transfer Type	Description
Control Transfer	Used for transmitting standard requests and vendor requests.
Bulk Transfer	Used for responding to vendor requests and transmitting a user program.

The following figure shows an overview of the USB communication flow.

Figure 34-7 USB Communication Flow

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The vendor ID and product ID for the USB boot device are 0xa108 and 0x4785 respectively. The Configuration for USB is for Control Endpoint 0 with Max Packet Size equals 64 bytes, Bulk IN at Endpoint 1 with Max Packet Size equals 512 bytes in high-speed and 64 bytes in full-speed, Bulk OUT at Endpoint 1 with Max Packet Size equals 512 bytes in high-speed and 64 bytes in full-speed.

The USB boot program provides six vendor requests through control endpoint for user to download/upload data to/from device, and to branch to a target address to execute user program. The six vendor requests are VR_GET_CPU_INFO (0x00), VR_SET_DATA_ADDRESS (0x01), VR_SET_DATA_LENGTH (0x02), VR_FLUSH_CACHES (0x03), VR_PROGRAM_START1 (0x04) and VR_PROGRAM_START2 (0x05). User program is transferred through Bulk IN or Bulk OUT endpoint.

When M200 is reset with boot_sel[2:0] equals 111b or 001b ,the internal boot ROM will switch to USB boot mode and wait for USB requests from host. After connecting the USB device port to host, host will recognize the connection of a USB device, and start device enumeration. After finishing the device enumeration, user can send VR_GET_CPU_INFO (0x00) to query the device CPU information. If user wants to download/upload a program to/from device, two vendor requests VR_SET_DATA_ADDRESS (0x01) and VR_SET_DATA_LENGTH (0x02) should be sent first to tell the device the address and length in byte of the subsequent transferring data. Then data can be transferred through bulk-out/bulk-in endpoint. After this first stage program has been transferred to device, user can send vendor request VR_PROGRAM_START1 (0x04) to let the CPU to execute the program. This first stage program must not greater than 26KB and is normally used to init GPIO and SDRAM of the target board. At the end of the first stage program, it can return back to the internal boot ROM by jumping to ra ($31) register. Thus user can download a new program to the SDRAM of the target board like the first stage, and send vendor request VR_FLUSH_CACHES (0x03) and VR_PROGRAM_START2 (0x05) to let the CPU to execute the new program. Next figure is the typical procedure of USB boot.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-8 Typical Procedure of USB Boot

Following tables list all the vendor requests that USB boot program supports:

Table 34-5 Vendor Request 0 Setup Command Data Structure

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.
1	bRequest	1	00H	VR_GET_CPU_INFO: get CPU information.
2	wValue	2	0000H	Not in used.
4	wIndex	2	0000H	Not in used.
6	wLength	2	0008H	8 bytes.

Table 34-6 Vendor Request 1 Setup Command Data Structure

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.
1	bRequest	1	01H	VR_SET_DATA_ADDRESS: set address for next bulk-in/bulk-out transfer.
2	wValue	2	xxxxH	MSB (bit[31:16]) of the data address.
4	wIndex	2	xxxxH	LSB (bit[15:0]) of the data address.
6	wLength	2	0000H	Not in used.

Table 34-7 Vendor Request 2 Setup Command Data Structure

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.
1	bRequest	1	02H	VR_SET_DATA_LENGTH: set length in byte for next bulk-in/bulk-out transfer.
2	wValue	2	xxxxH	MSB (bit[31:16]) of the data length.
4	wIndex	2	xxxxH	LSB (bit[15:0]) of the data length.
6	wLength	2	0000H	Not in used.

Table 34-8 Vendor Request 3 Setup Command Data Structure

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.
1	bRequest	1	03H	VR_FLUSH_CACHES: flush I-Cache and D-Cache.
2	wValue	2	0000H	Not in used.
4	wIndex	2	0000H	Not in used.
6	wLength	2	0000H	Not in used.

Table 34-9 Vendor Request 4 Setup Command Data Structure

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	bRequest	1	04H	VR_PROGRAM_START1: transfer data from D-Cache to I-Cache and branch to address in I-Cache. **NOTE:** After downloading program from host to device for the first time, you can only use this request to start the program. Since the USB boot program will download data to D-Cache after reset. This request will transfer data from D-Cache to I-Cache and execute the program in I-Cache.
2	wValue	2	xxxxH	MSB (bit[31:16]) of the program entry point.
4	wIndex	2	xxxxH	LSB (bit[15:0]) of the program entry point.
6	wLength	2	0000H	Not in used.

Table 34-10 Vendor Request 5 Setup Command Data Structure

Offset	Field	Size	Value	Description
0	bmRequestType	1	40H	D7 0: Host to Device. D6-D5 2: Vendor. D4-D0 0: Device.
1	bRequest	1	05H	VR_PROGRAM_START2: branch to target address directly.
2	wValue	2	xxxxH	MSB (bit[31:16]) of the program entry point.
4	WIndex	2	xxxxH	LSB (bit[15:0]) of the program entry point.
6	WLength	2	0000H	Not in used.

34.7 MSC0 Boot Specification

All cards can boot from MMC/SD Boot from MSC0, the boot program will load 26KB code starting at address 0x4400 from MMC/SD card to cache. First the boot program initializes MSC0_D0, MSC0_CLK, MSC0_CMD as function pins. Only one data pin MSC0_D0 is used. Then the boot program sends CMD55 to test if it's SD or MMC card and initializes the card. At last it loads 26KB code from the card to cache and branches to execute the code in cache.

When initializing the card, the clock of EXTCLK/128 is used when the EXTCLK is 48MHz or 26MHz, the clock of EXTCLK/64 is used when the EXTCLK is 24MHz or 19.2MHz. the clock of EXTCLK/32 is used when the EXTCLK is 12MHz . And then reading data, the clock of EXTCLK/4 is used.

The procedure of the M200 MMC/SD boot is shown as follow:

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-9 Typical Procedure of MSC0 Boot

SPL Structure show below , SPL signature fist 4 bytes is sd card magic code is used for Identification of SD,the remaining of the signature is unused and filled with 0xff. The first 512 byte of MBR store cpu configuration information.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-10 SPL Structure of MSC0/MSC1 Boot

34.8 eMMC Boot Specification

If eMMC is MultiMediaCard System Specification Ver. 4.4 compatible, you can use eMMC boot method. you should write boot code to boot partition1 , then the boot program will load 26KB code from eMMC boot partition1 area to cache. First the boot program initializes MSC0_D0, MSC0_CLK, MSC0_CMD as function pins. Only one data pin MSC0_D0 is used. Then the boot program sends CMD0 to set eMMC in boot mode. At last it loads 26KB code from the card to cache and branches to execute the code in cache. and the clock of EXTCLK/4 is used.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

The procedure of the M200 eMMC boot is shown as follow:

Figure 34-11 M200 eMMC Boot Procedure

SPL Structure show below , SPL signature fist 4 bytes is sd card magic code is used for Identification of SD,the remaining of the signature is unused and filled with 0xff.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 34-12 SPL Structure of eMMC Boot

34.9 msc1 boot Specification

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

It is the same as msc0 boot. The purpose of msc1 boot is that,you can use it to burn other devices.

34.10 SPI boot Specification

When boot_sel[2:0] is selected as SPI boot(boot_sel = 000), CPU will goes into SPI boot，AT first cpu fetch a 16 bytes flag information table（The first 16 bytes in SPL signature） from the SPI device。CPU check the flag table to decide the SPI address length ， SPI device type（NOR flash or NAND flash），clk rate， receive code length and the check data。SPI_boot detailed process shown “SPI Boot Procedure”

Note: Any irregularity in boot steps, SPI_boot will disable SSI controller and jump to MSC1 boot.

Figure 34-13 M200 SPI Boot Procedure

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.

SPI-NOR BOOT:

When SPI NOR flash by General-Purpose I/O Port-A 18, 19,20, 23 pin. In SPI NOR flash address 0x0~0xF, this space will store 16 bytes that agreed SPI boot flag.

The SPI boot flag information table and procedure of the M200 SPI NOR boot is shown as follow:

Table 34-11 SPI NOR flash boot flag informations（in spl signature）

addr	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
detail	0x04	0x03	0x02	0x55	0xaa	0x55	0xaa	SSI_GR
addr	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
detail	Len (LSB)	Len	Len	Len (MSB)	Check (LSB)	Check	Check	Check (MSB)

Explain:

0x00~0x03 :Must store 0x04,0x03,0x02

These part is used for cpu to decide the SPI address length

0X00~0X06 :Must store 0x55,0xAA,0x55,0xAA.

Magic code，device use this code decided whether is SPI-NOR flash boot

0x07 :The value of SSI Clock Generator Register.

0x08~0x0B :The length of copy data.

0x0C~0x0F :Copy data checksum.

SPI-NAND BOOT:

When SPI NAND flash by General-Purpose I/O Port-A 18, 19,20, 23 pin. Like spi nor flash, in spi NAND flash address 0x0~0xF, this space will store 16 bytes that agreed SPI_boot flag.

The SPI_boot flag information table and procedure of the M200 SPI NAND boot is shown as follow:

Table 34-12 SPI NAND flash boot flag informations（in spl signature）

addr	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
detail	0x04	0x03	0x02	x	x	x	x	SSI_GR
addr	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
detail	Len (LSB)	Len	Len	Len (MSB)	Check (LSB)	Check	Check	Check (MSB)

Explain:

0x00~0x02 :Must store 0x04,0x03,0x02

These part is used for cpu to decide the spi address length

0X03~0X06 :Must not be 0x55,0xAA,0x55,0xAA.

Magic code，device use this code decided whether is spi-nand flash boot

0x07 :The value of SSI Clock Generator Register.

0x08~0x0B :The length of copy data.

0x0C~0x0F :Copy data checksum.

Copyright © 2005-2014 Ingenic Semiconductor Co., Ltd. All rights reserved.