JZ4775

Mobile Application Processor

Programming Manual

Release Date: Apr. 10, 2015

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

JZ4775 Mobile Application Processor

Programming Manual

Copyright © 2005-2015 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 Dist., 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 3
1.2 Features 3
1.2.1 CPU 3
1.2.2 VPU 4
1.2.3 GPU 4
1.2.4 Display/Camera/Audio 4
1.2.5 Memory Interface 7
1.2.6 System Functions 8
1.2.7 Peripherals 10
1.2.8 Boot 12
1.3 Characteristic 12

Section 2 Core Functions

2 CPU 15

2.1 Block Diagram 16
2.2 Extra Features of the CPU core 17
2.3 Instruction Cycles 17
2.4 TCSM 20
2.4.1 TCSM Occupied Available Physical Address Range 20
2.5 PMON 21
2.5.1 Fundamental 22
2.6 Partial Kernel Mode 22

3 VPU 23

3.1 Block Diagram 23
3.2 Features of VPU 24
3.3 Internal physical address base definition 25
3.4 AUX 26
3.4.1 Register Definition 26
3.5 TCSM/SRAM 29

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.5.1 TCSM/SRAM space usage 29
3.6 Video Acceleration Block 29
4 EXtreme 2D 30
4.1 Overview 30
4.1.1 Feature 30
4.2 Registers Descriptions 31
4.2.1 Global control Register 32
4.2.2 Global Status Register 33
4.2.3 Global trigger Register 33
4.2.4 TLB address register 34
4.2.5 Description chain address register 34
4.2.6 Watch dog counter register 35
4.2.7 Destination Layers global control 35
4.2.8 Destination picture base address register 35
4.2.9 Destination picture control and stride register 36
4.2.10 Destination picture Geometric size register 36
4.2.11 Destination picture mask value register 37
4.2.12 Destination picture format register 37
4.2.13 Layer0 control Register 37
4.2.14 Layer0 input Y plane Address Register 39
4.2.15 Layer0 input U plane Address Register 39
4.2.16 Layer0 input V plane Address Register 39
4.2.17 Layer0 source frame Geometric Size Register 40
4.2.18 Layer0 plane Stride Register 40
4.2.19 Layer0 Output Geometric Size Register 40
4.2.20 Layer0 Output start pos offset Register 41
4.2.21 Layer0 scale ratio Register 41
4.2.22 Layer0 background ARGB Register 42
4.3 Software stack 42

Section 3 Display/Camera/Audio

5 LCD Controller 45
5.1 Overview 45
5.2 Pin Description 46
5.3 Block Diagram 47
5.4 LCD Display Timing 49
5.5 OSD Graphic 50
5.5.1 Color Key 50
5.6 Register Description 52
5.6.1 Configure Register (LCDCFG) 53
5.6.2 Control Register (LCDCTRL) 55

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.6.3 Status Register (LCDSTATE) 56
5.6.4 OSD Configure Register (LCDOSDC) 57
5.6.5 OSD Control Register (LCDOSDCTRL) 58
5.6.6 OSD State Register (LCDOSDS) 59
5.6.7 Background0 Color Register (LCDBGC0) 59
5.6.8 Background1 Color Register (LCDBGC1) 60
5.6.9 Foreground Color Key Register 0 (LCDKEY0) 60
5.6.10 Foreground Color Key Register 1 (LCDKEY1) 61
5.6.11 ALPHA Register (LCDALPHA) 61
5.6.12 RGB Control (LCDRGBC) 62
5.6.13 Virtual Area Setting (LCDVAT) 63
5.6.14 Display Area Horizontal Start/End Point (LCDDAH) 63
5.6.15 Display Area Vertical Start/End Point (LCDDAV) 63
5.6.16 Foreground 0 XY Position Register (LCDXYP0) 64
5.6.17 Foreground 1 XY Position Register (LCDXYP1) 64
5.6.18 Foreground 0 Size Register (LCDSIZE0) 65
5.6.19 Foreground 1 Size Register (LCDSIZE1) 65
5.6.20 Vertical Synchronize Register (LCDVSYNC) 65
5.6.21 Horizontal Synchronize Register (LCDHSYNC) 66
5.6.22 PS Signal Setting (LCDPS) 66
5.6.23 CLS Signal Setting (LCDCLS) 66
5.6.24 SPL Signal Setting (LCDSPL) 67
5.6.25 REV Signal Setting (LCDREV) 67
5.6.26 Interrupt ID Register (LCDIID) 68
5.6.27 Descriptor Address Registers (LCDDAx) 68
5.6.28 Source Address Registers (LCDSA) 69
5.6.29 Frame ID Registers (LCDFIDx) 69
5.6.30 DMA Command Registers (LCDCMDx) 70
5.6.31 DMA OFFSIZE Registers (LCDOFFSx) 71
5.6.32 DMA Page Width Registers (LCDPWx) 71
5.6.33 DMA Commend Counter Registers (LCDCNUMx) 72
5.6.34 DMA Commend Counter Registers (LCDCPOSx) 72
5.6.35 Foreground x Size in Descriptor (LCDDESSIZEx) 73
5.6.36 Priority level threshold configure Register (LCDPCFG) 73
5.7 LCD Controller Pin Mapping 74
5.7.1 TFT Pin Mapping 75
5.7.2 Data mapping to GPIO function. 76
5.8 Display Timing 76
5.8.1 General 16-bit and 18-bit TFT Timing 76
5.8.2 8-bit Serial TFT Timing 77
5.8.3 Special TFT Timing 78
5.8.4 Delta RGB panel timing 80
5.8.5 RGB Dummy mode timing 80

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.9 Format of Frame Buffer 81
5.9.1 16bpp 81
5.9.2 18bpp 81
5.9.3 24bpp 82
5.9.4 16bpp with alpha 82
5.9.5 18bpp with alpha 82
5.9.6 24bpp with alpha 82
5.9.7 24bpp compressed 82
5.10 Format of Data Pin Utilization 83
5.10.1 18-bit Parallel TFT 83
5.10.2 16-bit Parallel TFT 83
5.10.3 8-bit Serial TFT (24bpp) 83
5.11 LCD Controller Operation 83
5.11.1 Set LCD Controller AHB Clock and Pixel Clock 83
5.11.2 Enabling the Controller 84
5.11.3 Disabling the Controller 84
5.11.4 Resetting the Controller 85
5.11.5 Frame Buffer 85
5.11.6 OSD Operation 85
5.11.7 Descriptor Operation 86
6 Smart LCD Controller 87
6.1 Overview 87
6.2 Structure 87
6.3 Pin Description 88
6.4 Register Description 88
6.4.1 SLCD Configure Register (MCFG) 89
6.4.2 SLCD Control Register (MCTRL) 90
6.4.3 SLCD Status Register (MSTATE) 91
6.4.4 SLCD Data Register (MDATA) 91
6.5 System Memory Format 92
6.5.1 Data format 92
6.5.2 Command Format 92
6.6 Transfer Mode 93
6.6.1 DMA Transfer Mode 93
6.6.2 Register Transfer Mode 94
6.7 Timing 94
6.7.1 Parallel Timing 94
6.7.2 Serial Timing 95
6.8 Operation Guide 95
6.8.1 DMA Operation 95
6.8.2 Register Operation 96
7 EPD Controller 97

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.1 Overview 97
7.2 EPDC Pin Mappings 97
7.3 EPD Controller Registers 98
7.4 Registers Description 100
7.4.1 EPDC Control Registers 100
7.4.2 EPDC Configuration Registers 100
7.4.3 EPDC Status Register 101
7.4.4 EPDC ISR Control Register 102
7.4.5 EPDC DISPLAY0 Register 102
7.4.6 EPDC DISPLAY1 Register 104
7.4.7 EPDC SIZE Setting Register 104
7.4.8 EPDC Virtual Display Area Setting Register 105
7.4.9 EPDC Vertical Display Area Setting Register 105
7.4.10 EPDC Horizontal Display Area Setting Register 105
7.4.11 EPDC Vertical Synchronous Start Pulse Setting 106
7.4.12 EPDC Horizontal Synchronous Start Pulse Setting 106
7.4.13 EPDC Gate Driver Clock Setting Register 106
7.4.14 EPDC Gate Output Enable Setting Register 107
7.4.15 EPDC Gate Driver Start Pulse Setting 107
7.4.16 EPDC Source Driver Output Enable Setting Register 108
7.4.17 EPDC Source Driver Start Pulse Setting Register 108
7.4.18 EPDC Power Management Registers 0 108
7.4.19 EPDC Power Management Registers 1 109
7.4.20 EPDC Power Management Registers 2 109
7.4.21 EPDC Power Management Registers 3 109
7.4.22 EPDC Power Management Registers 4 110
7.4.23 EPDC LUT Base address Registers 110
7.4.24 EPDC LUT Size Registers 111
7.4.25 EPDC Current Buffer Base address Registers 111
7.4.26 EPDC Current Buffer Size Registers 112
7.4.27 EPDC Working Buffer0 Base address Registers 112
7.4.28 EPDC Working Buffer0 Size Registers 112
7.4.29 EPDC Working Buffer1 Base address Registers 113
7.4.30 EPDC Working Buffer0 Size Registers 113
7.4.31 EPDC VCOM/Border Registers 0~15 113
7.4.32 EPDC PRIORITY Registers 114
7.5 Application Guide 115
7.5.1 Data Buffer Requirement 115
7.5.2 Power On/Off Sequence 115
7.5.3 Display Timing Setting 116
7.5.4 Auto Stop 118
7.5.5 Partial/Parallel Update 119
7.5.6 Update VCOM0~15 119

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.5.7 Border Display 119
7.5.8 Update flow 119

8 EPD Color Engine 121

8.1 Overview 121
8.2 Features 121
8.3 Register Descriptions 121
8.3.1 Register Mapping 121
8.3.2 EPDCE Control Register (CTRLR) 123
8.3.3 Input Frame Size Register (IFSR) 123
8.3.4 Input Frame Offset Register (IFOFR) 124
8.3.5 Input Frame Address Register (IFAR) 124
8.3.6 Input Frame Window Size Register (IFWSR) 124
8.3.7 Output Frame Size Register (OFSR) 125
8.3.8 Output Frame Control Register (OFCR) 125
8.3.9 Output Frame Address 0 Register (OFA0R) 126
8.3.10 Output Frame Address 1 Register (OFA1R) 127
8.3.11 Interrupt Status Register (ISR) 127
8.3.12 Interrupt Mask Register (IMR) 128
8.3.13 Interrupt Clear Register (ICR) 128
8.3.14 Interrupt Input Frame ID Register (IIFIDR) 129
8.3.15 Interrupt Output Frame ID Register (IOFIDR) 129
8.3.16 DMA Control Register (DMACR) 129
8.3.17 DMA Status Register (DMASR) 130
8.3.18 Input DMA Descriptor Address Register (IDMADESR) 130
8.3.19 Output DMA Descriptor Address Register (ODMADESR) 131
8.3.20 Input DMA Command Register (IDMACMDR) 131
8.3.21 Output DMA Command 0 Register (ODMACMD0R) 132
8.3.22 Output DMA Command 1 Register (ODMACMD1R) 132
8.3.23 Input Frame ID Register (IFIDR) 133
8.3.24 Output Frame ID Register (OFIDR) 133
8.3.25 CSC Control Register (CSCCR) 133
8.3.26 3x3 Color Filter Control Register (CFCR) 134
8.3.27 3x3 Color Filter Red Coefficients Register (CFRCER0) 135
8.3.28 3x3 Color Filter Red Coefficients Register (CFRCER1) 135
8.3.29 3x3 Color Filter Red Coefficients Register (CFRCER2) 136
8.3.30 3x3 Color Filter Green Coefficients Register (CFGCER0) 136
8.3.31 3x3 Color Filter Green Coefficients Register (CFGCER1) 137
8.3.32 3x3 Color Filter Green Coefficients Register (CFGCER2) 137
8.3.33 3x3 Color Filter Blue Coefficients Register (CFBCER0) 138
8.3.34 3x3 Color Filter Blue Coefficients Register (CFBCER1) 138
8.3.35 3x3 Color Filter Blue Coefficients Register (CFBCER2) 139
8.3.36 Color Linearization Control Register (VEECR) 139

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.37 Color Correction Control Register (HUECR) 139
8.3.38 Color Correction Coefficients Register (HUECER) 140
8.3.39 Color Saturation Control Register (CSCR) 140
8.3.40 Color Saturation Coefficients Register (CSCER) 141
8.3.41 Dither Control Register (DTCR) 141
8.3.42 Color Remap Control Register (CRCR) 142
8.3.43 AXI Read Identifier Register (XRIDR) 142
8.3.44 AXI Write Identifier Register (XWIDR) 142
8.4 CSC from RGB888 to YUV444 143
8.4.1 601 Format Wide Conversion 143
8.4.2 601 Format Narrow Conversion 143
8.4.3 709 Format Wide Conversion 143
8.4.4 709 Format Narrow Conversion 144
8.5 CSC from YUV444 to RGB888 144
8.5.1 601 Format Wide Conversion 144
8.5.2 601 Format Narrow Conversion 144
8.5.3 709 Format Wide Conversion 144
8.5.4 709 Format Narrow Conversion 145
8.6 3x3 Color Filter 145
8.7 Color Linearization(VEE) 145
8.8 Color Correction (HUE) 146
8.9 Color Saturation 146
8.10 Dither 146
8.10.1 YUV.Y Dither 146
8.11 Color Remap 147
8.11.1 Translate Mode 147
8.11.2 Output Format of Color Remap 147
8.11.3 LUT of Color Remap 148
8.12 Common Input DMA 148
8.12.1 Descriptor of Common Input DMA 148
8.12.2 Software Flow to use Common Input DMA without Interrupt 149
8.12.3 Software Flow to use Common Input DMA with Interrupt 149
8.13 Common Output DMA 150
8.13.1 Descriptor of Common Output DMA 150
8.13.2 Software Flow to use Common Input DMA without Interrupt 151
8.13.3 Software Flow to use Common Input DMA with Interrupt 152
9 Camera Interface Module 154
9.1 Overview 154
9.1.1 Features 154
9.1.2 Pin Description 154
9.2 CIM Special Register 155
9.2.1 CIM Register Map 155

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2.2 CIM Configuration Register (CIMCFG) 155
9.2.3 CIM Control Register (CIMCR) 158
9.2.4 CIM Control Register 2 (CIMCR2) 159
9.2.5 CIM Status Register (CIMST) 161
9.2.6 CIM Interrupt Mask Register (CIMIMR) 162
9.2.7 CIM Interrupt ID Register (CIMIID) 163
9.2.8 CIM Descriptor Address (CIMDA) 164
9.2.9 CIM Frame buffer Address Register (CIMFA) 164
9.2.10 CIM Frame ID Register (CIMFID) 165
9.2.11 CIM DMA Command Register (CIMCMD) 165
9.2.12 CIM Window Size (CIMWSIZE) 166
9.2.13 CIM Window Offset (CIMWOFFSET) 167
9.2.14 CIM Frame Size Register (CIMFS) 167
9.2.15 CIM Y Frame buffer Address Register (CIMYFA) 168
9.2.16 CIM Y DMA Command Register (CIMYCMD) 168
9.2.17 CIM Cb Frame buffer Address Register (CIMCBFA) 169
9.2.18 CIM Cb DMA Command Register (CIMCBCMD) 169
9.2.19 CIM Cr Frame buffer Address Register (CIMCRFA) 170
9.2.20 CIM Cr DMA Command Register (CIMCRCMD) 170
9.2.21 CIM TLB Control Register (CIMTC) 171
9.2.22 CIM TLB Index Register (CIMTINX) 171
9.2.23 CIM TLB Content Register (CIMTCNT) 172
9.3 CIM Data Sampling Modes 172
9.3.1 Gated Clock Mode 172
9.3.2 ITU656 Interlace Mode 173
9.3.3 ITU656 Progressive Mode 174
9.4 DMA Descriptors 175
9.4.1 4-Word Descriptor 175
9.4.2 8-Word Descriptor 175
9.5 Interrupt Generation 176
9.6 Software Operation 177
9.6.1 Enable CIM with DMA 177
9.6.2 Enable CIM without DMA 177
9.6.3 Disable CIM 177

10 AC97/I2S/SPDIF Controller 178

10.1 Overview 178
10.1.1 Block Diagram 179
10.1.2 Features 179
10.1.3 Interface Diagram 180
10.1.4 Signal Descriptions 182
10.2 Register Descriptions 184
10.2.1 AIC Configuration Register (AICFR) 185

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.2.2 AIC Common Control Register (AICCR) 187
10.2.3 AIC AC-link Control Register 1 (ACCR1) 190
10.2.4 AIC AC-link Control Register 2 (ACCR2) 191
10.2.5 AIC I2S/MSB-justified Control Register (I2SCR) 193
10.2.6 AIC Controller FIFO Status Register (AICSR) 194
10.2.7 AIC AC-link Status Register (ACSR) 195
10.2.8 AIC I2S/MSB-justified Status Register (I2SSR) 197
10.2.9 AIC AC97 CODEC Command Address & Data Register (ACCAR, ACCDR) 197
10.2.10 AIC AC97 CODEC Status Address & Data Register (ACSAR, ACSDR) 198
10.2.11 AIC I2S/MSB-justified Clock Divider Register (I2SDIV) 199
10.2.12 AIC FIFO Data Port Register (AICDR) 200
10.2.13 SPDIF Enable Register (SPENA) 200
10.2.14 SPDIF Control Register (SPCTRL) 201
10.2.15 SPDIF State Register (SPSTATE) 202
10.2.16 SPDIF Configure 1 Register (SPCFG1) 202
10.2.17 SPDIF Configure 2 Register (SPCFG2) 203
10.2.18 SPDIF FIFO Register (SPFIFO) 205
10.3 Serial Interface Protocol 206
10.3.1 AC-link serial data format 206
10.3.2 I2S and MSB-justified serial audio format 206
10.3.3 Audio sample data placement in SDATA_IN/SDATA_OUT 209
10.3.4 SPDIF Protocol 210
10.4 AC97/I2S Operation 211
10.4.1 Initialization 211
10.4.2 AC ’97 CODEC Power Down 212
10.4.3 Cold and Warm AC ’97 CODEC Reset 212
10.4.4 External CODEC Registers Access Operation 214
10.4.5 Audio Replay 214
10.4.6 Audio Record 215
10.4.7 FIFOs operation 216
10.4.8 Data Flow Control 218
10.4.9 Audio Samples format 219
10.4.10 Serial Audio Clocks and Sampling Frequencies 221
10.4.11 Interrupts 225
10.5 SPDIF Guide 225
10.5.1 Set SPDIF clock frequency 225
10.5.2 PCM audio mode operation (Reference IEC60958) 225
10.5.3 Non-PCM mode operation (Reference IEC61937) 226
10.5.4 Disable operation 226
11 PCM Interface 227
11.1 Overview 227
11.2 Pin Description 227

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.3 Block Diagram 228
11.4 Register Description 228
11.4.1 PCM Control Register (PCMCTL) 229
11.4.2 PCM Configuration Register (PCMCFG) 230
11.4.3 PCM FIFO DATA PORT REGISTER (PCMDP) 231
11.4.4 PCM INTERRUPT CONTROL REGISTER (PCMINTC) 231
11.4.5 PCM INTERRUPT STATUS REGISTER (PCMINTS) 232
11.4.6 PCM CLOCK DIVIDE REGISTER (PCMDIV) 234
11.5 PCM Interface Timing 234
11.5.1 Short Frame SYN 235
11.5.2 Long Frame SYN 235
11.5.3 Multi-Slot Operation 236
11.6 PCM Operation 236
11.6.1 PCM Initialization 236
11.6.2 Audio Replay 237
11.6.3 Audio Record 238
11.6.4 FIFOs operation 238
11.6.5 Data Flow Control 239
11.6.6 PCM Serial Clocks and Sampling Frequencies 240
11.6.7 Interrupts 240

12 Internal CODEC Interface 241

12.1 Overview 241
12.1.1 Features 241
12.1.2 Signal Descriptions 241
12.1.3 Block Diagram 243
12.2 Mapped Register Descriptions 244
12.2.1 CODEC internal register access control (RGADW) 244
12.2.2 CODEC internal register data output (RGDATA) 245
12.3 Operation 246
12.3.1 Access to internal registers of the embedded CODEC 246
12.3.2 CODEC controlling and typical operations 246
12.3.3 Power saving 247
12.3.4 Pop noise and the reduction of it 247
12.4 Timing parameters 248
12.5 AC & DC parameters 249
12.6 CODEC internal Registers 250
12.6.1 CODEC internal registers 251
12.7 Programmable gains 271
12.7.1 Programmable boost gain: GIM 271
12.7.2 Programmable input gain amplifier: GID 272
12.7.3 Programmable digital attenuation: GOD 272
12.7.4 Programmable attenuation: GO 273

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.7.5 Programmable Bypass path attenuation: GI 274
12.7.6 Programmable digital mixer gain: GIMIX and GOMIX 274
12.7.7 Hard Clipping Preventer 275
12.7.8 Detailed description of the digital mixer 276
12.7.9 Gain refresh strategy 277
12.8 Configuration of the headphone output stage 277
12.9 Out-of-band noise filtering 278
12.10 Output short-circuit protection (headphone output) 278
12.10.1 Indication of the short circuit detection 278
12.10.2 Reset of short circuit detection 278
12.10.3 Capacitor-coupled headphone connection 278
12.11 Sampling frequency: FREQ 279
12.12 Programmable data word length 279
12.13 Ramping system note 280
12.14 AGC system guide 280
12.14.1 AGC operating mode 281
12.15 Digital Mixer description 282
12.16 Digital microphone interface 283
12.16.1 Timing Diagram 284
12.16.2 Timings 284
12.16.3 Noise template (TBC) 285
12.17 CODEC Operating modes 285
12.17.1 Power-On mode and Power-Off mode 286
12.17.2 RESET mode 286
12.17.3 STANDBY mode 287
12.17.4 SLEEP mode 287
12.17.5 Soft Mute mode 288
12.17.6 Power-Down mode and ACTIVE mode 289
12.17.7 Working modes summary 289
12.18 SYS_CLK turn-off and turn-on 290
12.19 Requirements on outputs and inputs selection and power-down modes 290
12.20 Anti-pop operation sequences 291
12.20.1 Initialization and configuration 291
12.20.2 Start up sequence (DAC) 291
12.20.3 Shutdown sequence (DAC) 293
12.20.4 Start up sequence (Line input) 295
12.20.5 Shutdown sequence (Line input) 295
12.21 Circuits design suggestions 296
12.21.1 Avoid quiet ground common currents 296
12.21.2 Headphone connection (Capacitor-coupled) 297
12.21.3 Microphone connection 298
12.21.4 Description of the connections to the jack 300
12.21.5 PCB considerations 301

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.22 Analog characteristics 303
12.22.1 Line input to audio ADC path 303
12.22.2 Microphone input to audio ADC path 303
12.22.3 Audio DAC to headphone output path 304
12.22.4 Audio DAC to mono line output path 305
12.22.5 Line input to headphone output path (analog bypass) 305
12.22.6 Microphone input to headphone output path (analog sidetone) 306
12.22.7 Micbias and reference 306

Section 4 Memory Interface

13 DDR Controller 308
13.1 Overview 308
13.1.1 Supported DDR SDRAM Types 308
13.1.2 Block Diagram 308
13.2 Register Description 309
13.2.1 DSTATUS 311
13.2.2 DCFG 312
13.2.3 DCTRL 315
13.2.4 DLMR 317
13.2.5 DTIMING1,2,3,4,5,6 (DDR Timing Configure Register) 319
13.2.6 DREFCNT (DDR Auto-Refresh Counter) 323
13.2.7 DMMAP0,1 (DDR Memory Map Configure Register) 324
13.2.8 DDLP (DDR DFI low power handshake control register) 324
13.2.9 DREMAP1,2,3,4,5 (DDR Address Remapping Register 1,2,3,4,5) 325
13.2.10 WCMDCTRL1 (Performance wcmd reorder & grouping) 326
13.2.11 RCMDCTRL0 (Performance rcmd request control) 328
13.2.12 RCMDCTRL1 (Performance rcmd request control) 329
13.2.13 WDATTHD0 (performance wcmd request control) 329
13.2.14 WDATTHD1 (performance wcmd request control) 330
13.2.15 IPORTPRI (performance priority control) 330
13.2.16 CHxQOS0,1,2,3,4,5 (performance QoS control) 332
13.2.17 CPM_DRCG 333
13.3 Functional Description 333
13.3.1 DDRC and DDR2 Memory Initialization Sequence 333
13.4 Change Clock Frequency 334
13.4.1 Manually SELF-REFRESH Mode 334
13.4.2 CPM driven SELF-REFRESH Mode 334
13.4.3 DLL bypass mode 335
13.5 Data Endian 335
13.6 DDR Connection Diagrams 335
13.6.1 Connection to one 512Mb x16 DDR2 devic 335
13.6.2 Connection to two 512Mb x16 DDR2 devices 336

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14 External NAND Memory Controller 337
14.1 Overview 337
14.2 Pin Description 337
14.3 Physical Address Space Map 338
14.4 Static Memory Interface 341
14.4.1 Register Description 341
14.4.2 Example of Connection 346
14.4.3 Basic Interface 347
14.4.4 Burst ROM Interface 350
14.5 NAND Flash Interface 351
14.5.1 Register Description 351
14.5.2 NAND Flash Boot Loader 367
14.5.3 NAND Flash Operation 368
14.5.4 Example of Toggle NAND Flash Access 369
15 BCH Controller 371
15.1 Overview 371
15.2 Register Description 373
15.2.1 BCH Control Register (BHCR) 373
15.2.2 BCH Control Set Register (BHCSR) 374
15.2.3 BCH Control Clear Register (BHCCR) 375
15.2.4 BCH ENC/DEC Count Register (BHCNT) 375
15.2.5 BCH Data Register (BHDR) 376
15.2.6 BCH Parity Register (BHPARn, n=0, 1, 2, 3, 4, 5…26, 27) 376
15.2.7 BCH Error Report Register (BHERRn, n=0,1,2,3,4,5,6,7,…,61,62,63) 377
15.2.8 BCH Interrupt Status Register (BHINT) 378
15.2.9 BCH Interrupt Enable Set Register (BHINTES) 380
15.2.10 BCH Interrupt Enable Clear Register (BHINTEC) 380
15.2.11 BCH Interrupt Enable Register (BHINTE) 381
15.2.12 BCH User TAG OUTPUT Register (BHTO) 382
15.2.13 BCH Erase Analyzer Register (BHEA) 382
15.3 BCH Operation 383
15.3.1 Encoding Sequence 383
15.3.2 Decoding Sequence 384

Section 5 System Functions

16 Clock Reset and Power Controller 388
16.1 Overview 388
16.1.1 CGU Block Diagram 388
16.1.2 CGU Registers 391
16.1.3 PLL Operation 420

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.1.4 Main Clock Division Change Sequence 421
16.2 Power Manager 422
16.2.1 Low-Power Modes and Function 422
16.2.2 Register Description 423
16.2.3 Doze Mode 429
16.2.4 IDLE Mode 430
16.2.5 SLEEP Mode 430
16.2.6 Power Down Mode 430
16.3 Reset Control Module 431
16.3.1 Register Description 431
16.3.2 Power On Reset 432
16.3.3 WDT Reset 432
17 Timer/Counter Unit 433
17.1 Overview 433
17.2 Pin Description 433
17.3 Register Description 433
17.3.1 Timer Control Register (TCSR) 435
17.3.2 Timer Data FULL Register (TDFR) 437
17.3.3 Timer Data HALF Register (TDHR) 437
17.3.4 Timer Counter (TCNT) 438
17.3.5 Timer Counter Enable Register (TER) 438
17.3.6 Timer Counter Enable Set Register (TESR) 439
17.3.7 Timer Counter Enable Clear Register (TECR) 440
17.3.8 Timer Flag Register (TFR) 441
17.3.9 Timer Flag Set Register (TFSR) 442
17.3.10 Timer Flag Clear Register (TFCR) 443
17.3.11 Timer Mast Register (TMR) 444
17.3.12 Timer Mask Set Register (TMSR) 444
17.3.13 Timer Mask Clear Register (TMCR) 445
17.3.14 Timer Stop Register (TSR) 446
17.3.15 Timer Stop Set Register (TSSR) 447
17.3.16 Timer Stop Clear Register (TSCR) 448
17.3.17 Timer Status Register (TSTR) 449
17.3.18 Timer Status Set Register (TSTSR) 450
17.3.19 Timer Status Clear Register (TSTCR) 450
17.3.20 Timer control mode Register (TCUMOD) 451
17.3.21 Timer fifo write data (TFWD) 452
17.3.22 Timer fifo state Register (TFIFOSR) 452
17.4 Operation 453
17.4.1 Basic Operation in TCU1 Mode 453
17.4.2 Disable and Shutdown Operation in TCU1 Mode 453
17.4.3 Basic Operation in TCU2 Mode 453

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.4.4 Disable and Shutdown Operation in TCU2 Mode 454
17.4.5 Read Counter in TCU2 Mode 454
17.4.6 Pulse Width Modulator (PWM) 454
17.4.7 Trackball Input Waveform Detect 455
17.4.8 Basic Operation in FIFO Mode 1 455
17.4.9 Basic Operation in FIFO Mode 2 456

18 Operating System Timer 458

18.1 Overview 458
18.2 Register Description 458
18.2.1 Operating System Control Register (OSTCSR) 458
18.2.2 Operating System Timer Data Register (OSTDR) 460
18.2.3 Operating System Timer Counter (OSTCNTH, OSTCNTL) 460
18.2.4 Operating System Timer Counter high 32 bits buffer (OSTCNTHBUF) 461
18.3 Operation 461
18.3.1 Basic Operation 461
18.3.2 Disable and Shutdown Operation 461

19 Interrupt Controller 462

19.1 Overview 462
19.2 Register Description 462
19.2.1 Interrupt Controller Source Register (ICSR0) 463
19.2.2 Interrupt Controller Source Register (ICSR1) 463
19.2.3 Interrupt Controller Mask Register (ICMR0) 463
19.2.4 Interrupt Controller Mask Register (ICMR1) 464
19.2.5 Interrupt Controller Mask Set Register (ICMSR0) 464
19.2.6 Interrupt Controller Mask Set Register (ICMSR1) 465
19.2.7 Interrupt Controller Mask Clear Register (ICMCR0) 465
19.2.8 Interrupt Controller Mask Clear Register (ICMCR1) 465
19.2.9 Interrupt Controller Pending Register (ICPR0) 466
19.2.10 Interrupt Controller Pending Register (ICPR1) 466
19.2.11 Interrupt Source Register0 for PDMA (DSR0) 467
19.2.12 Interrupt Mask Register0 for PDMA (DMR0) 467
19.2.13 Interrupt Pending Register0 for PDMA (DPR0) 468
19.2.14 Interrupt Source Register1 to PDMA (DSR1) 468
19.2.15 Interrupt Mask Register1 for PDMA (DMR1) 468
19.2.16 Interrupt Pending Register1 for PDMA (DPR1) 469
19.3 Software Considerations 469

20 Watchdog Timer 471

20.1 Overview 471
20.2 Register Description 471
20.2.1 Watchdog Control Register (TCSR) 471

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.2.2 Watchdog Enable Register (TCER) 472
20.2.3 Watchdog Timer Data Register (TDR) 473
20.2.4 Watchdog Timer Counter (TCNT) 473
20.3 Watchdog Timer Function 474
21 PDMA Controller 475
21.1 Features 475
21.2 Block Diagram 475
21.3 Memory Mapped Register Descriptions 476
21.3.1 DMA Channel Registers 476
21.3.2 Global Control Registers 476
21.4 DMA Channel Register Definition 477
21.4.1 DMA Source Address (DSAn, n = 0 ~ 31) 477
21.4.2 DMA Target Address (DTAn, n = 0 ~ 31) 477
21.4.3 DMA Transfer Count (DTCn, n = 0 ~ 31) 477
21.4.4 DMA Request Types (DRTn, n = 0 ~ 31) 478
21.4.5 DMA Channel Control/Status (DCSn, n = 0 ~ 31) 479
21.4.6 DMA Channel Command (DCMn, n = 0 ~ 31) 481
21.4.7 DMA Descriptor Address (DDAn, n = 0 ~ 31) 485
21.4.8 DMA Stride Difference (DSDn, n = 0 ~ 31) 485
21.5 DMA Global Register Definition 486
21.5.1 DMA Control 486
21.5.2 DMA Interrupt Pending (DIRQP) 487
21.5.3 DMA Doorbell (DDB) 487
21.5.4 DMA Doorbell Set (DDS) 488
21.5.5 DMA Channel Programmable (DMACP) 488
21.5.6 DMA Soft IRQ Pending 488
21.5.7 DMA Soft IRQ Mask 489
21.5.8 DMA Channel IRQ Pending to MCU 489
21.5.9 DMA Channel IRQ to MCU Mask 489
21.5.10 Programmable Channel Bound With INTC_IRQ 490
21.5.11 Special Channel 0 and Channel 1 490
21.6 MCU 490
21.6.1 MCU Control & Status 490
21.6.2 MCU Normal MailBox 491
21.6.3 MCU Security MailBox 492
21.6.4 MCU Interrupt 492
21.6.5 Multiple Bank Tightly Coupled Sharing Memory 492
21.6.6 Security ROM & Security RAM 493
21.6.7 CP0 Registers of MCU 493
21.6.8 Normal Exceptions Accepted by MCU 493
21.6.9 How to Boot MCU Up 494
21.6.10 SecurityCall Accepted by MCU 494

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.6.11 Interruptable SMD Mode 495
21.7 DMA manipulation 496
21.7.1 Descriptor Transfer Mode 496
21.7.2 No-Descriptor Transfer Mode 499
21.8 DMA Requests 499
21.8.1 Auto Request 500
21.8.2 On-Chip Peripheral Request 500
21.9 How to Use Programmable DMA Channel 500

22 SAR A/D Controller 501

22.1 Overview 501
22.2 Pin Description 501
22.3 Register Description 502
22.3.1 ADC Enable Register (ADENA) 502
22.3.2 ADC Configure Register (ADCFG) 503
22.3.3 ADC Control Register (ADCTRL) 505
22.3.4 ADC Status Register (ADSTATE) 506
22.3.5 ADC Same Point Time Register (ADSAME) 507
22.3.6 ADC Wait Pen Down Time Register (ADWAIT) 507
22.3.7 ADC Touch Screen Data Register (ADTCH) 507
22.3.8 ADC VBAT Data Register (ADVDAT) 510
22.3.9 ADC AUX Data Register (ADADAT) 511
22.3.10 ADC Clock Divide Register (ADCLK) 511
22.3.11 ADC Command Register (ADCMD) 512
22.3.12 ADC Stable Register (ADCSTB) 513
22.4 SAR A/D Controller Guide 513
22.4.1 Power Down Mode 513
22.4.2 A Sample Touch Screen Operation 513
22.4.3 SLEEP mode Sample Operation 515
22.4.4 VBAT Sample Operation 515
22.4.5 AUX Sample Operation 515
22.4.6 Disable Touch Screen 515
22.4.7 Multi-touch Operation 516
22.4.8 Use Software Command Operation 516
22.4.9 Use 5-wire touch panel Operation 516
22.4.10 Use External Touch Screen Controller Operation 517
22.4.11 Use TSC to support keypad 517

23 Real Time Clock 522

23.1 Overview 522
23.1.1 Features 522
23.1.2 Signal Descriptions 522
23.2 Register Description 523

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.2.1 RTC Control Register (RTCCR) 524
23.2.2 RTC Second Register (RTCSR) 526
23.2.3 RTC Second Alarm Register (RTCSAR) 526
23.2.4 RTC Regulator Register (RTCGR) 526
23.2.5 Hibernate Control Register (HCR) 527
23.2.6 HIBERNATE mode Wakeup Filter Counter Register (HWFCR) 528
23.2.7 Hibernate Reset Counter Register (HRCR) 528
23.2.8 HIBERNATE Wakeup Control Register (HWCR) 529
23.2.9 HIBERNATE Wakeup Status Register (HWRSR) 530
23.2.10 Hibernate Scratch Pattern Register (HSPR) 531
23.2.11 Write Enable Pattern Register (WENR) 531
23.2.12 CLK32K Pin control register (CKPCR) 532
23.2.13 Power on control register (PWRONCR) 533
23.3 Time Regulation 534
23.4 Operate Mode 534
23.4.1 Normal Mode 534
23.4.2 HIBERNATE Mode 535
23.5 Clock select 536

24 EFUSE Slave Interface (EFUSE) 538

24.1 Overview 538
24.2 Register Description 538
24.2.1 EFUSE Control Register (EFSCTL) 539
24.2.2 EFUSE Data Register (EFUSEn) 539
24.3 Flow 540
24.3.1 Write EFUSE Flow 540
24.3.2 Read EFUSE Flow 541

Section 6 Peripherals

25 General-Purpose I/O Ports 543

25.1 Overview 543
25.1.1 GPIO Port A Summary 543
25.1.2 GPIO Port B Summary 544
25.1.3 GPIO Port C Summary 545
25.1.4 GPIO Port D Summary 546
25.1.5 GPIO Port E Summary 546
25.1.6 GPIO Port F Summary 547
25.1.7 GPIO Port G Summary 548
25.2 Registers Description 549
25.2.1 PORT PIN Level Registers (PxPIN) 552
25.2.2 PORT Interrupt Registers (PxINT) 553
25.2.3 PORT Interrupt Set Registers (PxINTS) 553

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

25.2.4 PORT Interrupt Clear Registers (PxINTC) 554
25.2.5 PORT Mask Registers (PxMSK) 554
25.2.6 PORT Mask Set Registers (PxMSKS) 554
25.2.7 PORT Mask Clear Registers (PxMSKC) 555
25.2.8 PORT PAT1/Direction Registers (PxPAT1) 555
25.2.9 PORT PAT1/Direction Set Registers (PxPAT1S) 556
25.2.10 PORT PAT1/Direction Clear Registers (PxPAT1C) 556
25.2.11 PORT PAT0/Data Registers (PxPAT0) 557
25.2.12 PORT PAT0/Data Set Registers (PxPAT0S) 557
25.2.13 PORT PAT0/Data Clear Registers (PxPAT0C) 558
25.2.14 PORT FLAG Registers (PxFLG) 558
25.2.15 PORT FLAG Clear Registers (PxFLGC) 559
25.2.16 PORT PULL Disable Registers (PxPE) 559
25.2.17 PORT PULL Set Registers (PxPES) 560
25.2.18 PORT PULL Clear Registers (PxPEC) 560
25.3 Program Guide 560
25.3.1 Port Function Guide 560

26 SMB Controller 562

26.1 Overview 562
26.1.1 Features 562
26.1.2 Pin Description 562
26.2 Registers 563
26.2.1 Registers Memory Map 563
26.2.2 Registers and Fields Description 564
26.3 Operating Flow 587
26.3.1 SMB Behavior 588
26.3.2 Master Mode Operation 588
26.3.3 Slave Mode Operation 590
26.3.4 Disabling SMB 593

27 Synchronous Serial Interface 595

27.1 Overview 595
27.2 Pin Description 595
27.3 Register Description 596
27.3.1 Register Mapping 596
27.3.2 SSI Data Register (SSIDR) 596
27.3.3 SSI Control Register0 (SSICR0) 597
27.3.4 SSI Control Register1 (SSICR1) 599
27.3.5 SSI Status Register (SSISR) 602
27.3.6 SSI Interval Time Control Register (SSIITR) 603
27.3.7 SSI Interval Character-per-frame Control Register (SSIICR) 604
27.3.8 SSI Clock Generator Register (SSIGR) 604

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.3.9 SSI Receive Counter Register (SSIRCNT) 605
27.4 Functional Description 605
27.5 Data Formats 606
27.5.1 Motorola's SPI Format Details 606
27.5.2 TI's SSP Format Details 610
27.5.3 National Microwire Format Details 611
27.6 Interrupt Operation 613
28 UART Interface 614
28.1 Overview 614
28.1.1 Features 614
28.1.2 Pin Description 614
28.2 Register Descriptions 615
28.2.1 Register Mapping 615
28.2.2 UART Receive Buffer Register (URBR) 615
28.2.3 UART Transmit Hold Register (UTHR) 616
28.2.4 UART Divisor Latch Low/High Register (UDLLR / UDLHR) 616
28.2.5 UART Interrupt Enable Register (UIER) 617
28.2.6 UART Interrupt Identification Register (UIIR) 618
28.2.7 UART FIFO Control Register (UFCR) 619
28.2.8 UART Line Control Register (ULCR) 620
28.2.9 UART Line Status Register (ULSR) 621
28.2.10 UART Modem Control Register (UMCR) 623
28.2.11 UART Modem Status Register (UMSR) 624
28.2.12 UART Scratchpad Register 625
28.2.13 Infrared Selection Register (ISR) 625
28.2.14 UART M Register (UMR) 626
28.2.15 UART Add Cycle Register (UACR) 626
28.2.16 UART RXFIFO Counter Register (URCR) 627
28.2.17 UART TXFIFO Counter Register (UTCR) 627
28.3 Operation 627
28.3.1 UART Configuration 627
28.3.2 Data Transmission 628
28.3.3 Data Reception 628
28.3.4 Receive Error Handling 629
28.3.5 Modem Transfer 629
28.3.6 DMA Transfer 629
28.3.7 Slow IrDA Asynchronous Interface 629
28.3.8 For any frequency clock to use the UART 630
29 MMC/SD CE-ATA Controller 633
29.1 Overview 633
29.2 Features 633

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.3 Pins Description 633
29.4 Block Diagram 634
29.5 MMC/SD Controller Signal I/O Description 634
29.6 Register Description 635
29.6.1 MSC Control Register (MSC_CTRL) 636
29.6.2 MSC Control 2 Register (MSC_CTRL2) 638
29.6.3 MSC Status Register (MSC_STAT) 639
29.6.4 MSC Clock Rate Register (MSC_CLKRT) 641
29.6.5 MSC Command and Data Control Register (MSC_CMDAT) 641
29.6.6 MSC Response Time Out Register (MSC_RESTO) 644
29.6.7 MSC Read Time Out Register (MSC_RDTO) 645
29.6.8 MSC Block Size Register (MSC_BLKLEN) 645
29.6.9 MSC Block Count Register (MSC_NOB) 645
29.6.10 MSC Successfully-transferred Blocks Count Register (MSC_SNOB) 646
29.6.11 MSC Interrupt Mask Register (MSC_IMASK) 646
29.6.12 MSC Interrupt Flag Register (MSC_IFLG) 648
29.6.13 MSC Command Index Register (MSC_CMD) 651
29.6.14 MSC Command Argument Register (MSC_ARG) 651
29.6.15 MSC Response FIFO Register (MSC_RES) 651
29.6.16 MSC Receive FIFO Port Register (MSC_RXFIFO) 652
29.6.17 MSC Transmit FIFO Port Register (MSC_TXFIFO) 652
29.6.18 MSC Low Power Mode Register (MSC_LPM) 653
29.6.19 MSC DMA Control Register (MSC_DMAC) 653
29.6.20 MSC DMA Descriptor Address Register (MSC_DMANDA) 654
29.6.21 MSC DMA Data Address Register (MSC_DMADA) 655
29.6.22 MSC DMA Data Length Register (MSC_DMALEN) 655
29.6.23 MSC DMA Command Register (MSC_DMACMD) 656
29.6.24 MSC RTFIFO Data Counter Register(MSC_RTCNT) 657
29.7 MMC/SD Functional Description 657
29.7.1 MSC Reset 657
MSC Card Reset 658
29.7.2 Voltage Validation 658
29.7.3 Card Registry 658
29.7.4 Card Access 659
29.7.5 Protection Management 661
29.7.6 Card Status 664
29.7.7 SD Status 668
29.7.8 SDIO 668
29.7.9 Clock Control 670
29.7.10 Application Specified Command Handling 670
29.8 MMC/SD Controller Operation 671
29.8.1 Data FIFOs 671
29.8.2 DMA and Program I/O 672

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.8.3 Start and Stop clock 674
29.8.4 Software Reset 674
29.8.5 Voltage Validation and Card Registry 675
29.8.6 Single Data Block Write 677
29.8.7 Single Block Read 677
29.8.8 Multiple Block Write 678
29.8.9 Multiple Block Read 679
29.8.10 Stream Write (MMC) 680
29.8.11 Stream Read (MMC) 680
29.8.12 Erase, Select/Deselect and Stop 681
29.8.13 SDIO Suspend/Resume 682
29.8.14 SDIO ReadWait 682
29.8.15 Operation and Interrupt 682

30 OTG Controller 685

30.1 Overview 685
30.2 Pin Description 685
30.3 Control and Status Overview 685
30.4 CSR Memory Map 686
30.4.1 Register Maps 687
30.4.2 Global CSR Map 687
30.4.3 Host Mode CSR Map 688
30.4.4 Device Mode CSR Map 689
30.4.5 Data FIFO (DFIFO) Access Register Map 690
30.4.6 Interrupt Hierarchy 691
30.5 Register Descriptions 692
30.5.1 Application Access to the CSRs 692
30.5.2 Overview of Commonly Used Register Bits 694
30.5.3 Global Registers 698
30.5.4 Host Mode Registers 736
30.5.5 Device Mode Registers 754

Section 7 Boot

31 XBurst Boot ROM Specification 789

31.1 Boot Select 789
31.2 Boot Procedure 789
31.3 NAND Boot Specification 791
31.4 USB Boot Specification 795
31.5 MSC0 Boot Specification 799
31.6 eMMC Boot Specification 800
31.7 msc1 boot Specification 802
31.8 Spi boot Specification 802

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

TABLES

Table 3-1 VPU Features 24
Table 3-2 VPU Internal physical address base definition 25
Table 3-3 TCSM/SRAM space usage 29
Table 4-1 register list 31
Table 5-1 LCD Controller Pins Description 46
Table 5-2 LCD Controller Registers Description 52
Table 6-1 SLCD Pins Description 88
Table 7-1 EPDC Pin Mapping 97
Table 7-2 EPD Controller Registers 98
Table 9-1 Camera Interface Pins Description 154
Table 9-2 CIM Registers 155
Table 9-3 The modes and the corresponding signals used 172
Table 10-1 AIC Pins Description 182
Table 10-2 AIC Registers Description 184
Table 10-3 Sample data bit relate to SDATA_IN/SDATA_OUT bit 209
Table 10-4 Cold AC ’97 CODEC Reset Timing parameters 213
Table 10-5 Warm AC ’97 CODEC Reset Timing Parameters 213
Table 10-6 Audio Sampling rate, BIT_CLK and SYS_CLK frequencies 222
Table 10-7 BIT_CLK divider setting 223
Table 10-8 Approximate common multiple of SYS_CLK for all sample rates 223
Table 10-9 CPM/AIC clock divider setting for various sampling rate if PLL = 270.64MHz 224
Table 10-10 PLL parameters and audio sample errors for EXCLK=12MHz 224
Table 11-1 PCM Interface Pins Description 227
Table 11-2 PCM Registers Description 228
Table 12-1 CODEC signal IO pin description 241
Table 12-2 Internal CODEC Mapped Registers Description (AIC Registers) 244
Table 13-1 DDRC Register 309
Table 14-1 NEMC Pin Description 337
Table 14-2 Physical Address Space Map 340
Table 14-3 Default Configuration of NEMC Chip Select Signals 340
Table 14-4 Static Memory Interface Registers 341
Table 14-5 NAND Flash Interface Registers 351
Table15-1 BCH Registers 371
Table 16-1 CGU Registers Configuration 391
Table 16-2 Power/Reset Management Controller Registers Configuration 423
Table 17-1 PWM Pins Description 433
Table 19-1 INTC Register 462
Table 21-1 DMA Channel Registers (n=0~31) 476
Table 21-2 Global Control Registers 476
Table 21-3 Transfer Request Types 478
Table 21-4 Available RDIL 482

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 21-5 TCSM space 493
Table 21-6 Descriptor Structure 497
Table 22-1 SADC Pin Description 501
Table 22-2 SADC Register Description 502
Table 23-1 Registers for real time clock 523
Table 23-2 Registers for hibernating mode 523
Table 23-3 Clock select registers 536
Table 24-1 EFUSE Register Description 538
Table 25-1 GPIO Port A summary 543
Table 25-2 GPIO Port B summary 544
Table 25-3 GPIO Port C summary 545
Table 25-4 GPIO Port D summary 546
Table 25-5 GPIO Port E summary 546
Table 25-6 GPIO Port F summary 547
Table 25-7 GPIO Port G summary 548
Table 25-8 GPIO Registers 549
Table 26-1 SMB Pin Description 562
Table 26-2 Registers Memory Map-Address Base 563
Table 26-3 Registers Memory Map-Address Offset 563
Table 27-1 SSI Controller Pins Description 595
Table 27-2 SSI Serial Port Registers 596
Table 27-3 SSI Interrupts 613
Table 28-1 UART Pins Description 614
Table 28-2 UART Registers Description 615
Table 28-3 UART Interrupt Identification Register Description 619
Table 29-1 Command Token Format 635
Table 29-2 MMC/SD Data Token Format 635
Table 29-3 MMC/SD Controller Registers Map 635
Table 29-4 Command Data Block Structure 662
Table 29-5 Card Status Description 665
Table 29-6 SD Status Structure 668
Table 29-7 How to stop multiple block write 678
Table 29-8 How to stop multiple block read 679
Table 29-9 The mapping between Commands and Steps 682
Table 30-1 OTG Pins Description 685
Table 30-2 Core Global CSR Map (000h-3FFh) 687
Table 30-3 Host Mode CSR Map (400h-7FFh) 688
Table 30-4 Device Mode CSR Map (800h-BFFh) 689
Table 30-5 Data FIFO (DFIFO) Access Register Map 690
Table 30-6 List of Commonly Used Register Bits 694
Table 30-7 Control and Status Register: GOTGCTL 698
Table 30-8 Interrupt Register: GOTGINT 702
Table 30-9 AHB Configuration Register: GAHBCFG 703

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-10 USB Configuration Register: GUSBCFG 706
Table 30-11 Reset Register: GRSTCTL 712
Table 30-12 Interrupt Register: GINTSTS 715
Table 30-13 Interrupt Mask Register: GINTMSK 721
Table 30-14 Host Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP 724
Table 30-15 Device Mode Receive Status Debug Read/Status Read and Pop Registers: GRXSTSR/GRXSTSP 724
Table 30-16 Receive FIFO Size Register: GRXFSIZ 725
Table 30-17 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Host Mode and Device Shared FIFO Mode) 725
Table 30-18 Non-Periodic Transmit FIFO Size Register: GNPTXFSIZ (Device Dedicated FIFO Mode) 726
Table 30-19 Non-Periodic Transmit FIFO/Queue Status Register: GNPTXSTS 726
Table 30-20 User HW Config1 Register: GHWCFG1 727
Table 30-21 User HW Config2 Register: GHWCFG2 728
Table 30-22 User HW Config3 Register: GHWCFG3 730
Table 30-23 User HW Config4 Register: GHWCFG4 732
Table 30-24 Global DFIFO Software Config Register: GDFIFOCFG 734
Table 30-25 Host Periodic Transmit FIFO Size Register: HPTXFSIZ 734
Table 30-26 Device Periodic Transmit FIFO-n Register: DPTXFSIZn 735
Table 30-27 Device In Endpoint Transmit FIFO Size Register: (DIEPTXFn) 736
Table 30-28 Host Configuration Register: HCFG 737
Table 30-29 Host Frame Interval Register: HFIR 739
Table 30-30 Host Frame Number/Frame Time Remaining Register: HFNUM 739
Table 30-31 Host Periodic Transmit FIFO/Queue Status Register: HPTXSTS 740
Table 30-32 Host All Channels Interrupt Register: HAINT 741
Table 30-33 Host All Channels Interrupt Mask Register: HAINTMSK 741
Table 30-34 Host Port Control and Status Register: HPRT 742
Table 30-35 Host Channel-n Characteristics Register: HCCHARn 745
Table 30-36 Host Channel-n Split Control Register: HCSPLTn 747
Table 30-37 Host Channel-n Interrupt Register: HCINTn 748
Table 30-38 Host Channel-n Interrupt Mask Register: HCINTMSKn 749
Table 30-39 Host Channel-n Transfer Size Register: HCTSIZn 750
Table 30-40 Host Channel-n DMA Address Register: HCDMAn 752
Table 30-41 Host Channel-n DMA Buffer Address Register: HCDMABn 754
Table 30-42 Host Frame List Base Address Register: HFLBAddr 754
Table 30-43 Device Configuration Register: DCFG 755
Table 30-44 evice Control Register: DCTL 757
Table 30-45 Minimum Duration for Soft Disconnect 761
Table 30-46 Device Status Register: DSTS 761
Table 30-47 Device IN Endpoint Common Interrupt Mask Register: DIEPMSK 762
Table 30-48 Device OUT Endpoint Common Interrupt Mask Register: DOEPMSK 763

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-49 Device All Endpoints Interrupt Register: DAINT 764
Table 30-50 Device Endpoints Interrupt Mask Register: DAINTMSK 764
Table 30-51 Device IN Token Sequence Learning Queue Read Register 1: DTKNQR1 765
Table 30-52 Device IN Token Sequence Learning Queue Register 2: DTKNQR2 765
Table 30-53 Device IN Token Sequence Learning Queue Register 3: DTKNQR3 766
Table 30-54 Device IN Token Sequence Learning Queue Register 4: DTKNQR4 766
Table 30-55 Device VBUS Discharge Time Register: DVBUSDIS 767
Table 30-56 Device VBUS Pulsing Time Register (DVBUSPULSE) 767
Table 30-57 Device Threshold Control Register (DTHRCTL) 767
Table 30-58 Device IN Endpoint FIFO Empty Interrupt Mask Register: DIEPEMPMSK 769
Table 30-59 Device Each Endpoint Interrupt Register: DEACHINT 769
Table 30-60 Device Each Endpoint Interrupt Register Mask: DEACHINTMSK 770
Table 30-61 Device Each In Endpoint-n Interrupt Register: DIEPEACHMSKn 770
Table 30-62 Device Each Out Endpoint-n Interrupt Register: DOEPEACHMSKn 771
Table 30-63 Device Control IN Endpoint 0 Control Register: DIEPCTL0 772
Table 30-64 Device OUT Endpoint 0 Control Register: DOEPCTL0 774
Table 30-65 Device Endpoint-n Control Register: DIEPCTLn/DOEPCTLn 775
Table 30-66 Device Endpoint-n Interrupt Register: DIEPINTn/DOEPINTn 780
Table 30-67 Device IN Endpoint 0 Transfer Size Register: DIEPTSIZ0 783
Table 30-68 Device OUT Endpoint 0 Transfer Size Register: DOEPTSIZ0 783
Table 30-69 Device Endpoint-n Transfer Size Register: DIEPTSIZn/DOEPTSIZn 784
Table 30-70 Device Endpoint-n DMA Address Register: DIEPDMAn/DOEPDMAn 785
Table 30-71 Device Endpoint-n DMA Buffer Address Register: DIEPDMABn/DOEPDMABn 786
Table 30-72 Device IN Endpoint Transmit FIFO Status Register: DTXFSTSn 786
Table 31-1 Boot Configuration of JZ4775 789
Table 31-2 The definition of 5 flags in NAND flash 791
Table 31-3 Transfer Types Used by the Boot Program 795
Table 31-4 Vendor Request 0 Setup Command Data Structure 798
Table 31-5 Vendor Request 1 Setup Command Data Structure 798
Table 31-6 Vendor Request 2 Setup Command Data Structure 798
Table 31-7 Vendor Request 3 Setup Command Data Structure 798
Table 31-8 Vendor Request 4 Setup Command Data Structure 799
Table 31-9 Vendor Request 5 Setup Command Data Structure 799
Table 31-10 SPI nor flash boot flag informations 802

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

FIGURES

Figure 1-1 JZ4775 Diagram 3
Figure 2-1 Structure of CPU core 16
Figure 3-1 VPU Block Diagram 23
Figure 5-1 Block Diagram when use OSD mode 47
Figure 5-2 Block Diagram of TFT mode (not use OSD) 48
Figure 5-3 Block Diagram of HDMI interface 48
Figure 5-4 Display Parameters 49
Figure 5-5 OSD Graphic 50
Figure 5-6 General 16-bit and 18-bit TFT LCD Timing 77
Figure 5-7 8-bit serial TFT LCD Timing (24bpp) 78
Figure 5-8 Special TFT LCD Timing 1 79
Figure 5-9 Special TFT LCD Timing 2 79
Figure 5-10 Delta RGB timing 80
Figure 5-11 RGB Dummy timing 81
Figure 8-1 Individual CFA Component Translation Mode 147
Figure 8-2 Pixel Array Translation Mode 147
Figure 9-1 Typical BT.656 Vertical Blanking Intervals for 625/50 Video Systems 174
Figure 9-2 ITU656 Progressive Mode 175
Figure 10-1 AIC Block Diagram 179
Figure 10-2 Interface to an External AC'97 CODEC Diagram 180
Figure 10-3 Interface to an External Master Mode I2S/MSB-Justified CODEC Diagram 181
Figure 10-4 Interface to an External Slave Mode I2S/MSB-Justified CODEC Diagram 181
Figure 10-5 Interface to a HDMI Transmitter via SPDIF Diagram 181
Figure 10-6 Interface to an internal Master Mode I2S CODEC Diagram 182
Figure 10-7 AC-link audio frame format 206
Figure 10-8 AC-link tag phase, slot 0 format 206
Figure 10-9 AC-link data phases, slot 1 ~ slot 12 format 206
Figure 10-10 I2S data format (A: LR mode) 207
Figure 10-11 I2S data format (B: RL mode) 207
Figure 10-12 MSB-justified data format (C: LR mode) 208
Figure 10-13 MSB-justified data format (D: RL mode) 208
Figure 10-14 Block format 210
Figure 10-15 Sub-frame format in PCM mode 210
Figure 10-16 Sub-frame format in non-PCM mode 210
Figure 10-17 Cold AC '97 CODEC Reset Timing 213
Figure 10-18 Warm AC '97 CODEC Reset Timing 213
Figure 10-19 Transmitting/Receiving FIFO access via APB Bus 217
Figure 10-20 One channel (Left) and Two channels (right) mode (16 bits packed mode) 219
Figure 10-21 Four channels (Left) and Six channels (right) mode (16 bits packed mode) 219
Figure 10-22 Eight channels mode (16 bits packed mode) 220
Figure 10-23 One channel (Left) and Two channels (right) mode 220

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 10-24 Four channels (Left) and Six channels (right) mode 220
Figure 10-25 Eight channels mode 221
Figure 10-26 SYS_CLK, BIT_CLK and SYNC generation scheme 222
Figure 11-1 Short Frame SYN Timing (Shown with 16bit Sample) 235
Figure 11-2 Short Frame SYN Timing (Shown with 16bit Sample) 235
Figure 11-3 Long Frame SYN Timing (Shown with 16bit Sample) 235
Figure 11-4 Long Frame SYN Timing (Shown with 16bit Sample) 236
Figure 11-5 Multi-Slot Frame SYN Timing (Shown with two Slots and 8bit Sample) 236
Figure 11-6 Transmitting/Receiving FIFO access via APB Bus 239
Figure 11-7 PCMCLK and PCMSYN generation scheme 240
Figure 12-1 CODEC block diagram 243
Figure 12-2 Internal CODEC works with AIC 243
Figure 12-3 AGC adjusting waves 281
Figure 12-4 AGC adjust areas 282
Figure 12-5 Digital microphone interface connection 283
Figure 12-6 Digital microphone timing diagram at MCLK = 12 MHz 284
Figure 12-7 Digital microphone modulation noise reference spectrum 285
Figure 12-8 CODEC Power Diagram 286
Figure 12-9 Gain up and gain down sequence 288
Figure 12-10 Start up sequence 292
Figure 12-11 Shutdown sequence 294
Figure 12-12 Capacitor-coupled connection 297
Figure 12-13 Capacitor-less connection 298
Figure 12-14 Ground distributing 301
Figure 12-15 the bottom corner of chip PCB Layer 302
Figure 13-1 DDRC block diagram 309
Figure 14-1 Physical Address Space Map 339
Figure 14-2 Example of 16-Bit Data Width SRAM Connection 346
Figure 14-3 Example of 8-Bit Data Width SRAM Connection 347
Figure 14-4 Basic Timing of Normal Memory Read 348
Figure 14-5 Basic Timing of Normal Memory Write 348
Figure 14-6 Normal Memory Read Timing With Wait (Software Wait Only) 349
Figure 14-7 Normal Memory Write Timing With Wait (Software Wait Only) 349
Figure 14-8 Normal Memory Read Timing With Wait (Wait Cycle Insertion by WAIT# pin) 350
Figure 14-9 Burst ROM Read Timing (Software Wait Only) 351
Figure 14-10 Toggle NAND Page Write/Read Operation 355
Figure 14-11 Example of DQS Delay Line Auto-Detect-Operation 363
Figure 14-12 Basic Timing of Toggle NAND Write 365
Figure 14-13 Basic Timing of Toggle NAND Read 366
Figure 14-14 Basic Timing of Toggle NAND Page Write 366
Figure 14-15 Basic Timing of Toggle NAND Page Read 367
Figure 14-16 Structure of NAND Flash Boot Loader 367
Figure 14-17 Static Bank 1 Partition When NAND Flash is Used (an example) 368

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 14-18 Example of 8-bit NAND Flash Connection 369
Figure 14-19 Toggle NAND Page Read Operation 369
Figure 14-20 Toggle NAND Page Program Operation 369
Figure 14-21 Toggle NAND Page Program Operation (Basic) 370
Figure 14-22 Program Operation with Random Data Input 370
Figure 15-1 Block Diagram for BCH encoding 384
Figure 15-2 Block Diagram for BCH decoding 385
Figure 15-3 BCH decoding data flow 386
Figure 16-1 Block Diagram of PLL 420
Figure 21-1 Block Diagram of PDMA 475
Figure 21-2 Descriptor Transfer Flow 498
Figure 21-3 Example for Stride Transfer Mode 499
Figure 22-1 6x5 keypad circuit 518
Figure 22-2 Wait for pen-down (C=1100) circuit 519
Figure 22-3 Measure X-position (C=0010) circuit 520
Figure 22-4 Measure Y-position (C=0011) circuit 520
Figure 23-1 Core Power On directly 535
Figure 23-2 Core Power On-Off-On 535
Figure 23-3 RTC clock selection path 536
Figure 27-1 SPI Single Character Transfer Format (PHA = 0) 607
Figure 27-2 SPI Single Character Transfer Format (PHA = 1) 607
Figure 27-3 SPI Back-to-Back Transfer Format 608
Figure 27-4 SPI Frame Interval Mode Transfer Format (ITFRM = 0, LFST = 0) 609
Figure 27-5 SPI Frame Interval Mode Transfer Format (ITFRM = 1, LFST = 1) 610
Figure 27-6 TI's SSP Single Transfer Format 610
Figure 27-7 TI's SSP Back-to-back Transfer Format 611
Figure 27-8 National Microwire Format 1 Single Transfer 612
Figure 27-9 National Microwire Format 1 Back-to-back Transfer 612
Figure 27-10 National Microwire Format 2 Read Timing 612
Figure 27-11 National Microwire Format 2 Write Timing 613
Figure 29-1 MMC/SD Controller Block Diagram 634
Figure 30-1 OTG CSR Memory Map 686
Figure 30-2 Interrupt Hierarchy 691
Figure 30-3 Core Interrupt Handler 692
Figure 31-1 Boot sequence diagram of JZ4775 790
Figure 31-2 the distribution and structure of the boot code in NAND 792
Figure 31-3 JZ4775 NAND Boot Procedure 794
Figure 31-4 USB Communication Flow 795
Figure 31-5 Typical Procedure of USB Boot 797
Figure 31-6 JZ4775 eMMC Boot Procedure 801
Figure 31-7 JZ4775 SPI Boot Procedure 803

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 1

OVERVIEW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1 Overview

JZ4775 is a mobile application processor targeting for multimedia rich and mobile devices like tablet computer, Ebook, mobile digital TV. This SOC introduces a kind of innovative architecture to fulfill both high performance mobile computing and high quality video decoding requirements addressed by mobile multimedia devices. JZ4775 provides high-speed CPU computing power and fluent 720p video replay.

The CPU (Central Processing Unit) core, equipped with 16kB instruction and 16kB data level 1 cache, and 256kB level 2 cache, operating at 1GHz, and full feature MMU function performs OS related tasks. At the heart of the CPU core is XBurst® processor engine. XBurst® is an industry leading microprocessor core which delivers superior high performance and best-in-class low power consumption. A hardware floating-point unit which compatible with IEEE754 is also included.

The VPU (Video Processing Unit) core is powered with another XBurst® processor engine. The SIMD instruction set implemented by XBurst® engine, in together with the on chip video accelerating engine and post processing unit, delivers high video performance. The maximum resolution of 720p in the formats of H.264, VC-1, MPEG-1/2, MPEG-4, RealVideo and VP8 are supported in decoding.
The memory interface supports a variety of memory types that allow flexible design requirements, including glueless connection to SLC NAND flash memory or up to 64-bit ECC MLC/TLC NAND flash memory and toggle NAND flash for cost sensitive applications. It provides the interface to DDR2, DDR3 and LPDDR memory chips with lower power consumption.

On-chip modules such as audio CODEC, multi-channel SAR-ADC, AC97/I2S controller and camera interface offer designers a rich suite of peripherals for multimedia application. The LCD controller support regular RGB, 1024x768 output, WLAN, Bluetooth and expansion options are supported through high-speed SPI and MMC/SD/SDIO host controllers. Other peripherals such as USB OTG and USB 1.1 host, UART and SPI as well as general system resources provide enough computing and connectivity capability for many applications.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1.1 Block Diagram

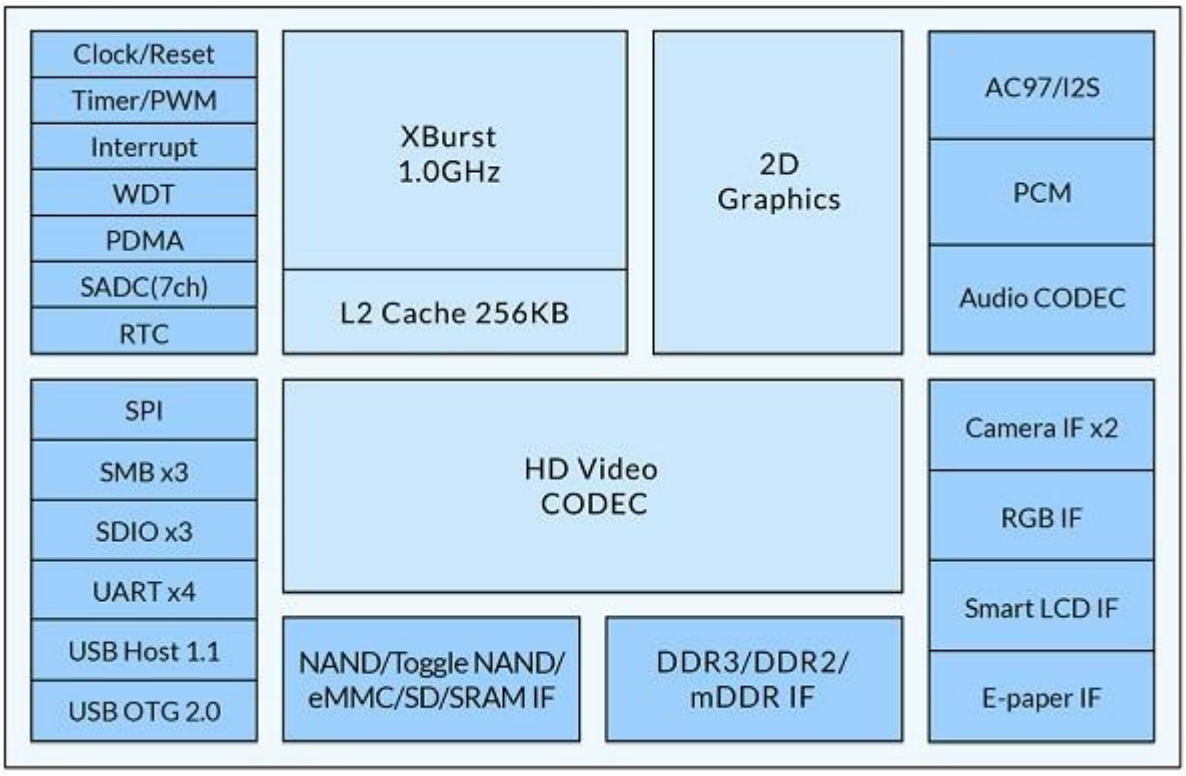


Figure 1-1 JZ4775 Diagram

1.2 Features

1.2.1 CPU

- 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, the maximum frequency is 1G
- MMU
  - 32-entry joint-TLB
  - 4 entry Instruction TLB
  - 4 entry data TLB
- L1 Cache
  - 16kB instruction cache
  - 16kB data cache
- Hardware debug support
- 16kB tight coupled memory

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- L2 Cache
  - 256kB unify cache

1.2.2 VPU

- MPEG-1/2 decoding up to 720P 30fps
- VC-1 decoding up to 720P 30fps
- H.264 decoding up to 720P 30fps
- VP8 decoding up to 720P 30fps
- MPEG-4 decoding up to 720P 30fps
- RV9 decoding up to 720P 30fps

1.2.3 GPU

- X2D
  - Location: AHB bus
  - Input format
    - Separate frame: YUV /YCbCr (4:2:0)
    - Packaged data: RGB888, RGB565, RGB555, NV12, NV21,TileYUV
  - Output data format
    - ARGB888，XRGB888, RGB555, RGB565
  - Color convention coefficient: configurable (CSC enable)
  - Minimum input image size (pixel): 4x4
  - Maximum input image size (pixel): 12288x12288 (12k x 12k)
  - Maximum output image size (pixel)
    - Width : up to 12288
    - Height: up to 12288
  - Image resizing
    - bi-cube zooming mode
  - Image Clockwise 90, 180, 270 rotation
  - Image horizontal and vertical mirror , same time with rotation
  - 5 layers OSD

1.2.4 Display/Camera/Audio

- LCD controller(compress must with IPU direct display)
  - Basic Features
    - Support panel(TFT, SLCD)
    - Display size up to 1024*600@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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- 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

- EPD Controller
  - 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
- EPD Color Engine
  - Input data format is RGB565
  - Maximum image direction is 4096x4096
  - Includes CSC between RGB888 and YUV444
  - CSC supports 601 or 709, Wide or Narrow mode
  - Includes 3x3 Color Filter modules for RGB.R, RGB.B, RGB.B and YUV.Y.
  - Includes Color Linearization(VEE) for YUV.Y using 256-grade LUT
  - Supports Color Correction(HUE) for YUV.UV, and the coefficients are configurable
  - Supports Color Saturation for YUV.UV, and the coefficients are configurable
  - Supports Dither for RGB.R, RGB.B, RGB.B and YUV.Y. The output format is 2/3/4-bit configurable.
  - Supports Color Remapping for RGB.R, RGB.B, RGB.B and YUV.Y. If for RGB, there are two methods can be selected between individual CFA component and pixel array. And, the output order is configurable.
  - The EPDCE has a AXI master interface and a AHB slave interface.
- Camera interface module
  - Input image size up to 2048x2048 pixels
  - Max. VGA for image preview
  - Max. VGA for video record
  - Integrated DMA
  - Supported data format: YCbCr 4:4:4, YCbCr 4:2:2 and other formats

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Output format
  - csc mode: YCbCr 4:2:2 or YCbCr 4:2:0
  - bypass mode: the input data format
- Output frame format
  - Packaged : for all data format
  - Separated: for YCbCr 4:4:4, YCbCr 4:2:2 and YCbCr 4:2:0
- Supports ITU656 (YCbCr 4:2:2) input
- Configurable CIM_VSYNC and CIM_HSYNC signals: active high/low
- Configurable CIM_PCLK: active edge rising/falling
- 256x33 image data receive FIFO (RXFIFO)
- PCLK max. 80MHz
- Configurable output order

- AC97/I2S/SPDIF 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
    - 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
    - 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

  - 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
- PCM interface
  - 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
- Internal CODEC Interface
  - 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, SNR: 95dB, THD: -65dB @17.6mW /16Ohm
  - DAC to stereo line output path @10kOhm: SNR: 95dB A-Weighted, THD: -80dB @FS-1dB
  - Line input to ADC path: SNR: 95dB 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
  - Digital MIC supported.
  - Support Capacitor-coupled and Capacitor-less mode headphone connection
  - Advance SNR of recode.
  - Updata AGC system.
  - Add digital amplitude limiter use for remove the short when sound is very largely.
  - Add DAC digital amplifier the gain up to 32dB.

1.2.5 Memory Interface

- DDR Controller
  - Support DDR2, DDR3, mobile DDR (LPDDR), memory，up to 800Mbps

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Support x16 and x32 external DDR data width
- Asynchornize to system bus and each 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 and order

- 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 8/16-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
  - 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
- BCH Controller
  - Support up to 64-bit ECC encoding and decoding for NAND
- The XBurst® processor system supports little endian only

1.2.6 System Functions

- Clock generation and power management
  - On-chip 24/26MHZ 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
  - SSI clock supports 50M clock
  - MSC clock supports 100M clock
  - Functional-unit clock gating
  - Shut down power supply for J1, VPU, L2CC, X2D
- Timer and counter unit with PWM output and/or input edge counter
  - Provide eight channels, four channels 0~3 can generate PWM, two of them have input

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
  - Every channel has PWM output
- 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
  - 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
  - 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 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
- SAR A/D Controller
  - 7 Channels
  - Resolution: 12-bit
  - Integral nonlinearity: ±1 LSB
  - Differential nonlinearity: ±0.5 LSB
  - Resolution/speed: up to 2Msps

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- Max Frequency: 200k
- Low power dissipation: 1.5mW(worst)
- Support 4-wire and 5-wire touch panel measurement (Through pin XP, XN, YP, YN and AUX2)
- Support multi-touch detect
- Support write control command by software
- Support voltage measurement (Through pin VBAT)
- Support two auxiliary input (Through pin AUX1, AUX2)
- Single-end and Differential Conversion Mode
- Auto X/Y, X/Y/Z1/Z2 and X/Y/Z1/Z2/X2/Y2 position measurement
- Support external touch screen controller
- Pin Description

- RTC (Real Time 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

1.2.7 Peripherals

- SMB Controller
  - 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
  - 3 independent SMB channels (SMB0, SMB1, SMB2)
- Synchronous serial interfaces (SSI0)
  - 3 protocols support: National's Microwire, TI's SSP, and Motorola's SPI

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- 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

- Four UARTs (UART0, UART1, UART2, UART3)
  - 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
- Three MMC/SD/SDIO controllers (MSC0, MSC1, MSC2)
  - 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
  - Maskable 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 1.1 host interface
  - Open Host Controller Interface OHCI-compatible and USB Revision 1.1-compatible
  - Full speed and low speed
  - Embedded USB 1.1 PHY

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- 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
  - 16 Endpoints:
  - Dedicate FIFO
  - Supports control, interrupt, ISO and bulk transfer
- GMAC controller
  - 10/100/1000 Mbps operation
  - Supports MII、RMII、GMII and RGMII PHY interfaces
  - Supports VLAN and CRC
  - Station Management Agent (SMA)
  - remote wake-up frame and magic packet frame processing
- OTP Slave Interface
  - Total 256 bits. Lower 128bits are read-able and write-able, Higher 128bits are read only
- General-Purpose I/O ports
  - 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

1.2.8 Boot

16kB Boot ROM memory

1.3 Characteristic

Item	Characteristic
Process Technology	65nm CMOS low power
Power supply voltage	General purpose I/O: 1.6~3.6V DDR I/O for DDR2: 1.8V± 0.1V DDR I/O for DDR3: 1.5V± 0.075V DDR I/O for DDR3L: 1.35V± 0.1V DDR I/O for LPDDR: 1.8V± 0.15V RTC I/O: 1.8V~3.6V EFUSE programming: 2.5V± 10%

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Analog power supply 1: 2.5V± 10% Analog power supply 2: 3.3V± 10% Core: 1.2 -0.1/+0.2 V
Package	BGA314 14mm x 14mm x 1.4mm, 0.65mm pitch
Operating frequency	1GHz

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 2

CORE FUNCTIONS

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 CPU

Enhanced features of CPU core include:

- Enhanced MXU implements XBurst SIMD instruction set release I and release II
- Full implementation of MIPS32 integer instruction release II
- TCSM, tightly coupled shared memory with physical address scope 0x132B0000 ~ 0x132BFFFF
- PMON, processor performance monitor
- FPU, floating point unit implemented to improve floating point number processing ability
- Unified level 2 cache that is transparent for programmer

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2.1 Block Diagram

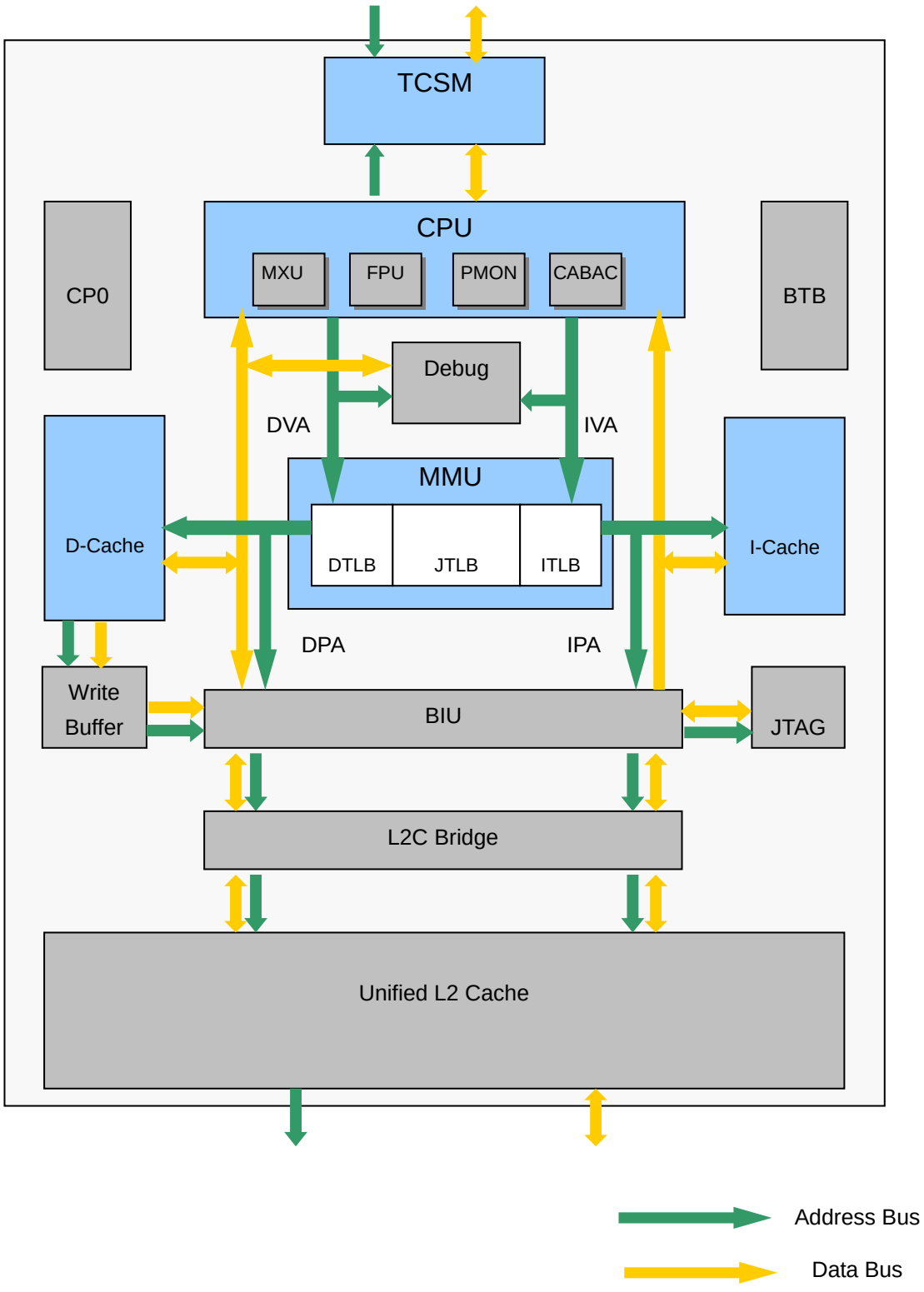


Figure 2-1 Structure of CPU core

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2.2 Extra Features of the CPU core

Item	Features
Media Extension Unit (MXU)	● XBurst SIMD instruction set release I and release II ● fully pipelined
Integer Unit with MIPS32 integer instruction release II	● non full pipelined implementation for most of MIP32 integer instruction release II, need 2 ~ 4 interlock cycles
Tightly Coupled Sharing Memory (TCSM)	● Size: 16K bytes ● Same clock frequency as L1 cache ● AHB slave interface ● Four banks support up to four simultaneous accesses
Floating Point Unit (FPU)	● Comply with IEEE754 standard ● Support single and double format ● not fully pipelined implementation
CABAC interface	● Part of bitstream processing cooperating CABAC in VPU ● Dedicated CP0 interface is CP0 register number 21, select0~7
Performance Monitor (PMON)	● Real-time monitor ● Dedicated CP0 interface
Unified Level 2 Cache	● Size: 256K bytes ● 4 way set association with LRU replacement ● Write from Level 1 data cache always write through to memory ● Programmer transparent, that is, those CACHE instructions managing L1 cache can manipulate L2 cache automatically
Processor ID	Value read from CP0.PRId is 0x2ed1024f

Please refer to documents XBurst-ISA and XBurst1_PM for ISA and programming relative details.

2.3 Instruction Cycles

Most instructions have one cycle repeat rate, that is, when the pipeline is fully filled, there is one instruction issued per clock cycle. However, some particular instructions require extra cycles. Following table lists cycle consumption of all instructions belonging to XBurst-ISA implemented.

1st Instruction	2nd Instruction	Cycles	Description
WAIT	**Anyone**	variable	WAIT instruction will be repeatedly executed until an interrupt arises.
MTC0 **TLBWI/TLBWR** **TLBP/TLBR**	**Anyone**	4	3 extra interlock cycles.
CACHE	**Anyone**	2	1 extra interlock cycles.
JMP/BC	**Anyone** **(delay slot)**	4/1	0 cycle penalty when BTB predicts taken and the branch is taken or BTB predicts

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			untaken and the branch is untaken or BTB miss and the branch is untaken. Otherwise, extra 3 cycles penalty.
BCL	**Anyone (delay slot)**	5/4/2/1	0 cycle penalty when BTB predicts taken and branch is taken, otherwise: 1 BTB miss, branch is taken, 3 cycles penalty. 2 BTB miss, branch is untaken, 1 cycle penalty. 3 BTB predict taken, branch is untaken, 4 cycles penalty. 4 BTB predict untaken, branch is taken, 3 cycles penalty.
MULT/MULTU MADD/MADDU MSUB/MSUBU	**MULT/MULTU MADD/MADDU MSUB/MSUBU**	4	3 extra interlock cycles due to MDU operating hazard.
	MUL/DIV/DIVU	4	3 extra interlock cycles due to MDU operating hazard.
	MFHI/MFLO MTHI/MTLO	4	3 extra interlock cycles due to MDU operating hazard.
	Any other	1	No data dependency or hazards exist.
MUL	**MULT/MULTU MADD/MADDU MSUB/MSUBU**	4	3 extra interlock cycles due to MDU operating hazard.
	MUL/DIV/DIVU	4	3 extra interlock cycles due to MDU operating hazard.
	MFHI/MFLO MTHI/MTLO	4	3 extra interlock cycles due to MDU operating hazard.
	Any other	4/3/2/1	If the second instruction has RAW data dependency, 3 extra interlock cycles; similarly, 2 extra for the third RAW one and 1 extra for the forth RAW one, otherwise, 0 cycle penalty.
DIV/DIVU	**MULT/MULTU MADD/MADDU MSUB/MSUBU MUL/DIV/DIVU**	4~35	3~34 extra interlock cycles determined by characteristic value of divider and dividend.
	MFHI/MFLO	2~34	1~33 interlock cycles determined by characteristic value of divider and dividend.
	Any other	1	No data dependency or hazards exist.
MFHI/MFLO/MFC0	**Anyone**	4/3/2/1	If the second instruction has RAW data dependency, 3 extra interlock cycles, similarly, 2 extra for the third RAW one

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			and 1 extra for the forth RAW one, otherwise, 0 cycle penalty.
LW/LL **LWL/LWR** **LB/LBHU** **LH/HU** **LXW** **LXH/LXHU** **LXB/LXBU**	**Anyone**	4/3/2/1	If the second instruction has RAW data dependency, 3 extra interlock cycles, similarly, 2 extra for the third RAW one and 1 extra for the forth RAW one, otherwise, 0 cycle penalty.
D16MUL/D16MULF **D16MAC/D16MACF** **D16MULE/D16MACE**	**SIMD instruction**	3/2/1	If the second SIMD instruction has RAW data dependency, 2 extra interlock cycles, similarly, 1 extra for the third RAW one, otherwise, 0 cycle penalty.
	Any other	1	No data dependency or hazards exist.
D32ACC/Q16ACC **Q8SAD** **S32MAX/S32MIN** **D16MAX/D16MIN** **D32ACCM/D32ASUM** **Q16ACCM/D16ASUM**	**SIMD instruction**	2/1	If the second SIMD instruction has RAW data dependency, 1 extra interlock cycle, otherwise, 0 cycle penalty.
	Any other	1	No data dependency or hazards exist.
S32LDD/S32LDDV **S32LDI/S32LDIV** **S32LDDR/S32LDDVR** **S32LDIR/S32LDIVR** **S16LDD/S16DI** **S8LDD/S8LDI**	**SIMD instruction**	2/1	If the second SIMD instruction has RAW data dependency, 1extra interlock cycle, otherwise, 0 cycle penalty.
	Any other	1	No data dependency or hazards exist.
S32I2M	**SIMD instruction**	2/1	If the second SIMD instruction has RAW data dependency, 1extra interlock cycle, otherwise, 0 cycle penalty.
	Any other	1	No data dependency or hazards exist.
S32M2I	**Anyone**	4/3/2/1	If the second instruction has RAW data dependency, 3 extra interlock cycles, similarly, 2 extra for the third RAW one and 1 extra for the forth RAW one, otherwise, 0 cycle penalty.
S32EXTR **S32EXTRV**	**SIMD instruction**	2/1	If the second SIMD instruction has RAW data dependency, 1extra interlock cycle, otherwise, 0 cycle penalty.
	Any other	1	No data dependency or hazards exist.
Others	**Anyone**	1	--

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: JMP denotes J and JR instructions; BC denotes branch conditionally instructions; BCL denotes
branch conditionally and likely instructions.

2.4 TCSM

TCSM (tightly-coupled shared memory) is a dedicated on-chip SRAM. It serves as an on-chip scratchpad memory, moreover, it acts as a high-speed SRAM for CPU. Through the TCSM, CPU and VPU's AHB masters such as DBlock can exchange data quickly and efficiently. TCSM in the CPU core has following features:

- 16K bytes
- The same clock frequency as L1 cache
- Physical address scope from 0x132B,0000 to 0x132B,FFFF
- Four banks support up to four simultaneous accesses if no bank conflicts occur

Moreover, like the **dseg** section separated from K3 section, another **tcsm** section with 16MB capacity range from 0xF400,0000 to 0xF4FF,FFFF is separated too. This virtual address section is uncacheable and unmappable and can only be accessed by CPU core in kernel mode.

Please note the fact that the capacity of TCSM is only 16K bytes, which denotes that available virtual address range is from 0xF400,0000 to 0xF400,3FFF and available physical address range is from 0x132B,0000 to 0x132B,3FFF.

2.4.1 TCSM Occupied Available Physical Address Range

Physical Address range 0x132B,0000 ~ 0x132B,FFFF are reserved for TCSM. Physical address range 0x132B,0000 ~ 0x132B,3FFF are available and others are reserved, and corresponding address partition for the four banks are as following:

bank0: 0xF4000000~0xF4000FFF (virtual); 0x132B,0000~0x132B,0FFF (physical)
bank1: 0xF4001000~0xF4001FFF (virtual); 0x132B,1000~0x132B,1FFF (physical)
bank2: 0xF4002000~0xF4002FFF (virtual); 0x132B,2000~0x132B,2FFF (physical)
bank3: 0xF4003000~0xF4003FFF (virtual); 0x132B,3000~0x132B,3FFF (physical)

Therefore, arranging instructions and data into different banks can achieve best access performance. Similarly, using ping-pong buffers located in the separate banks for efficient data exchange between CPU core and other VPU's AHB masters is a better choice.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2.5 PMON

PMON is a simple performance monitor. In JZ4770, PMON can make real-time monitoring for following hardware events.

- I-cache miss times, D-cache miss times
- Total issued instructions, Discarded instructions
- Pipeline freeze cycles, CPU clock cycles
- TLB exceptions caused by instruction fetch, TLB exceptions caused by data load/store

Moreover, in JZ4770, PMON can be configured to work in expected mode.

- Normal mode (when PMON is enabled, always work until it is disabled)
- User mode (when PMON is enabled, only work in user mode and paused in kernel mode)
- Kernel mode (when PMON is enabled, only work in kernel mode and paused in user mode)
- PC mode (when PMON is enabled, only work when PC locates in preset address range)

A dedicated software interface is devised to manipulate PMON in kernel mode, that is, CP0 Config4 ~ Config7 registers are extended for PMON. Refer to chapter of CP0 in the document XBurst1_PM for detail. However, since new function and bit fields has been expanded for Config7 register, following definition of Config7 is the precise description for JZ4770.

	Config16																													select 7	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC_HI																PEVENT				MODE			PME	Reserved	PK	Reserved			ALLOC	BTBV	BTBE
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	R/W
31:16	PC_HI	If PMON work in PC range mode, the valid range is PC_HI,0000 ~ PC_HI,FFFC.	RW
15:12	PEVENT	Event pair encoding. 0000: count of pipeline freeze cycles, count of cpu clock cycles 0001: times count of icache-miss, times count of dcache-miss 0010: count of discarded instructions, count of issued instructions 0011: count of TLB exceptions caused by fetching instruction, count of TLB exceptions caused by data load/store 0100 – 1111: reserved	RW
11:9	MODE	000: null mode, work unconditionally 001: work in user mode 010: work in kernel mode 011: work in specific PC range	RW
8	PME	PMON enable bit. 0: disable; 1: enable.	RW
7	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6	PK	Partial kernel mode. 0: forbid; 1: permit. Refer to later description.	RW
5:3	Reserved	Writing has no effect, read as zero.	R
2	ALLOC	Allocate hint of PREF instruction. 0: enabled (default); 1: disabled.	RW
1	BTBV	BTB invalid. Writing 1 to this bit to invalidates BTB.	W
0	BTBE	BTB enable. 0: enabled (default); 1: disabled.	RW

2.5.1 Fundamental

When PMON is enabled (set value 1 to config7.bit8), one preset event pair determined by config7.bit15~bit12 will be continuously monitored until PMON is disabled (set value 0 to config7.bit8).

Finally, loading values of CP0.config4~CP0.config6 can get monitored result.

2.6 Partial Kernel Mode

Setting 1 to config7.bit6 can permit applications in user mode possess some kernel mode oriented resources including TCSM, CABAC I/F, CACHE instructions. This is a shortcut for those performance sensitive applications such as video codec. However, OS must make serious control for config7.bit6 and those dedicated resources to forbid this partial-kernel-mode permission for those malicious applications.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3 VPU

Video Processing Unit (VPU) core in this chip is dedicated for video decoding. VPU embeds an XBurst® CPU core (named AUX in VPU) and application specified hardware accelerators for common video decompress algorithms, which includes Stream Parser, Motion Compensation, Inverse Quant, 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-1/2 decoding up to 720P 30fps
- VC-1 decoding up to 720P 30fps
- H.264 decoding up to 720P 30fps
- VP8 decoding up to 720P 30fps
- MPEG-4 decoding up to 720P 30fps
- RV9 decoding up to 720P 30fps

3.1 Block Diagram

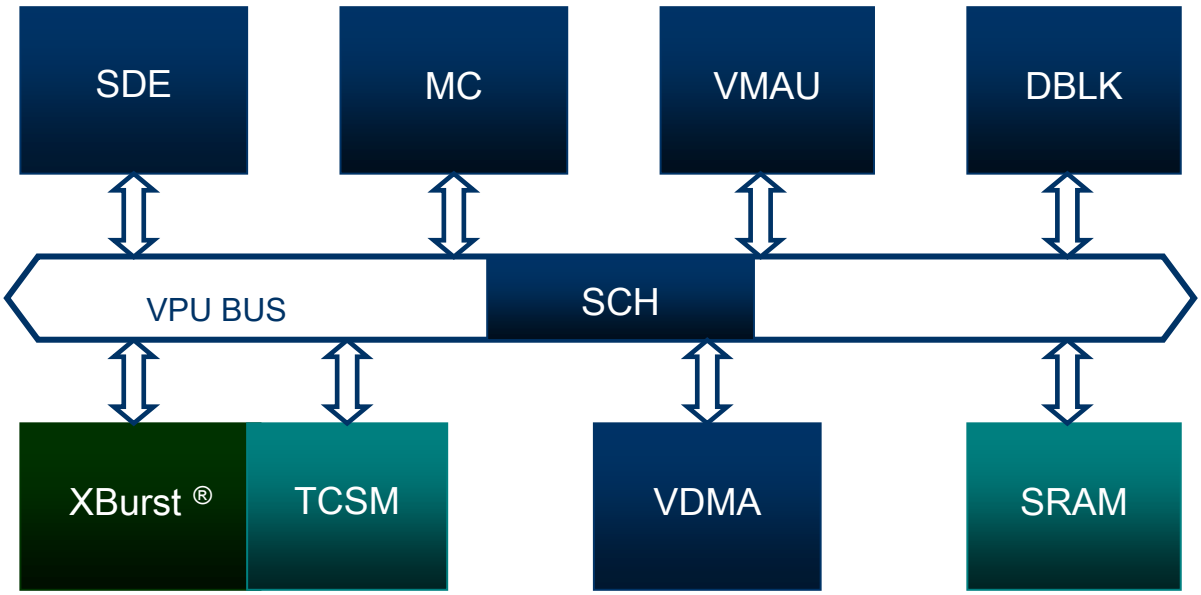


Figure 3-1 VPU Block Diagram

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.2 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: 16K bytes – AHB slave interface supports external DMA access **NOTE:** TCSM is coupled with VPU XBurst® core internally.
Scratch RAM (SRAM)	● Size: 8K 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 ● H.264 Context adaptive variable length (CAVLC) decoding support, PCM not supported ● VC-1 bitstream decoding support ● MPEG2 bitstream decoding support ● VP8 bitstream decoding support
Motion (MC)	Motion compensation 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 ● Interlaced mode support ● Intensity compensation support ● Weighted prediction support ● Automatic rotation support for rotated referenced pictures ● Automatic expanding support for outside frame's reference

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Recover (VMAU)	Recover is a Matrix Arithmetic Unit in VPU (named as VMAU), it serves for pixel's recovery and reconstruction during vedio decoding. ● Descriptor based task fetching ● Configurable format intra prediction support ● Configurable format Inverse quant support ● Configurable format IDCT support ● Residual add for pixel's recovery
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-based write buffer for DMA out
Scheduler (SCH)	Scheduler is a special unit which is used to manage internel functional engines' handshake. ● 4 programable channels ● 2 programable groups
Interrupt control center (INTC)	● Task end interrupt (frame/slice end) ● Bitstream error interrupt ● VDMA ACFG error interrupt

3.3 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
MC	0x1325_0000
DBLK	0x1327_0000
VMAU	0x1328_0000
SDE	0x1329_0000

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.4 AUX

3.4.1 Register Definition

3.4.1.1 Control and Status

	AUX_CTRL																															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
	SLEEP	Reserved																						BTB_INV	Reserved				MIRQ_EN	NMI_DIS	SW_NMI	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	0	0	0	0	0	1

Bits	Name	Description	R/W
31	SLEEP	AUX sleep status. 1: sleep; 0: no sleep.	R
30:9	Reserved	Writing has no effect, read as zero.	R
8	BTB_INV	Writing 1 can invalid BTB. Writing 0 has no effect, read as zero.	W
7:4	Reserved	Writing has no effect, read as zero.	R
3	MIRQ_EN	1: enable message IRQ. 0: disable.	RW
2	NMI_DIS	1: NMI only wakes up AUX from sleep status 0: NMI wakes up AUX and switch PC to 0xF4000000	RW
1	SW_NMI	Nonmaskable IRQ (NMI). Writing 1 to the field triggers a NMI pulse to AUX. Writing 0 has no effect, read as zero.	W
0	SW_RST	Software reset. 1: let AUX keep at reset status; 0: do not reset.	RW

NOTES:

1 When NMI or IRQ or RESET exception occurs, AUX resumes from PC 0xF4000000.

2 When AUX wakes up by an NMI meanwhile NMI_DIS is 1, AUX just resumes from the next PC of the WAIT instruction.

3.4.1.2 SPINLOCK

	AUX_SPINLK																															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
	Reserved																														LOCK	
RST	0	0	0	0	0	0	0	0	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
30:2	Reserved	Writing has no effect, read as zero.	R
1:0	LOCK	Lock status.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	AUX_SPIN1																													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																														SPIN1	
RST	0	0	0	0	0	0	0	0	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
30:2	Reserved	Writing has no effect, read as zero.	R
1:0	SPIN1	Reading SPIN1 triggers following special hardware operations. First, value of AUX_SPINLK will be checked, if the value equals zero, the value of SPIN1 will overwrite AUX_SPINLK immediately, otherwise, AUX_SPINLK keeps unchanged. Then reading AUX_SPINLK instead of SPIN1 supplies the final read result. Writing SPIN1 is a normal write operation.	RW

	AUX_SPIN2																													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
	Reserved																														SPIN2	
RST	0	0	0	0	0	0	0	0	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
30:2	Reserved	Writing has no effect, read as zero.	R
1:0	SPIN2	The operations for SPIN1 also fit SPIN2 except the role of SPIN1 should be replaced by SPIN2.	RW

	AUX_MIRQP																													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																															MIRQP
RST	0	0	0	0	0	0	0	0	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
31:1	Reserved	Writing has no effect, read as zero.	R
0	MIRQP	Pending status of MIRQ (message IRQ to CORE) which can only be set to 1 by HW and be reset to 0 by SW. This pending IRQ is routing to main CPU core.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	AUX_MSG																														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
	MESG																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	0	?	0	0

Bits	Name	Description	R/W
31:0	MESG	If AUX_CTRL.MIRQ_EN is value 1, writing the register raises an IRQ routing to the main CPU core meanwhile the AUX_MIRQP is set to 1 by HW automatically. The IRQ then keeps active until the register AUX_MIRQP is cleared to 0 by SW.	RW

	CORE_MIRQP																														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																															MIRQP
RST	0	0	0	0	0	0	0	0	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
31:1	Reserved	Writing has no effect, read as zero.	R
0	MIRQP	Pending status of MIRQ (message IRQ to AUX) which can only be set 1 by HW and be clear to 0 by SW. This pending IRQ is routing to AUX.	RW

	CORE_MSG																														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
	MESG																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	R/W
31:0	MESG	If AUX_CTRL.MIRQ_EN is value 1, writing the register raises an IRQ routing to the AUX. The IRQ then keeps active until the register CORE_MIRQP is cleared to 0 by SW.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3.5 TCSM/SRAM

TCSM0/TCSM1/SRAM serves as the VPU control flow and data flow's communication between XBurst® CPU core with specified algorithm hardware accelerators and different hardware accelerators as well.

3.5.1 TCSM/SRAM space usage

Table 3-3 TCSM/SRAM space usage

	XBurst® AUX	HW accelerator
TCSM	0xF400_0000 ~ 0xF400_3FFF	0x132C_0000 ~ 0x132C_3FFF
SRAM	0x132F_0000 ~ 0x132F_1FFF	0x132F_0000 ~ 0x132F_1FFF
NOTES: 1. TCSM severs as AUX's instruction and data memory and VPU inner HW accelerator data sharing space as well. 2. SRAM should not be accessed when it is in using by VPU internally.		

3.6 Video Acceleration Block

Please refer to relative programming manual documents.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 EXtreme 2D

4.1 Overview

X2D is the unit which can take charge of rotation, scale, mirror, csc, and 4 layer osd.

4.1.1 Feature

Location: AHB bus

Input format

- Separate frame: YUV /YCbCr (4:2:0)
- Packaged data: RGB888, RGB565, RGB555, NV12, NV21,TileYUV

Output data format

- ARGB888，XRGB888, RGB555, RGB565

Color convention coefficient: configurable (CSC enable)

Minimum input image size (pixel): 4x4

Maximum input image size (pixel): 12288x12288 (12k x 12k)

Maximum output image size (pixel)

- Width : up to 12288
- Height: up to 12288

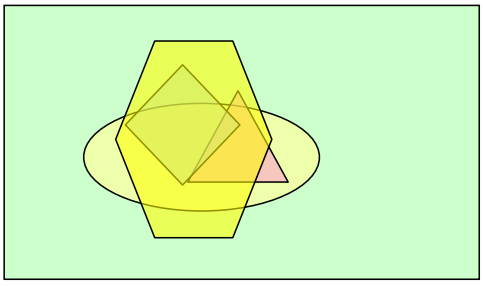
Image resizing

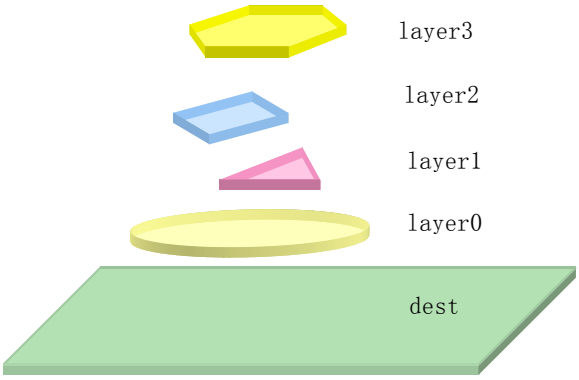
- bi-cube zooming mode

Image Clockwise 90, 180, 270 rotation

Image horizontal and vertical mirror , same time with rotation

5 layers OSD





Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4.2 Registers Descriptions

The physical address base for the address-mapped registers of X2D is **0x13030000.**

The following table will show all the register address.

Table 4-1 register list

NAME	Offset	Descript
GLB_CTRL	0xF000	global controller
GLB_STATUS	0xF004	Global status
GLB_TRIG	0xF008	Global trigger
TLB_BASE	0xF00C	Base address of TLB (physical address map) table
DHA	0xF010	Descript chain's base address
WDOG_CNT	0xF018	Watch dog's counter
LAY_GCTRL	0xE000	Destination layers global control
DST_BASE	0xE004	Destination picture's base address
DST_CTRL_STR	0xE008	Destination picture's stride
DST_GS	0xE00C	Destination picture's size(height and width)
DST_MSK_ARGB	0xE010	Destination picture's ARGB mask
DST_FMT	0xE014	Destination picture format
Chanel 0		
LAY0_CTRL	0x0000	Source format control register
LAY0_Y_ADDR	0x0004	Source Y (or RGB) base address
LAY0_U_ADDR	0x0008	Source U (or UV) base address
LAY0_V_ADDR	0x000C	Source V base address
LAY0_IN_FM_GS	0x0010	Input Geometric Size (height and width)
LAY0_STRIDE	0x0014	Source frame stride ((Y or RGB) and UV)
LAY0_OUT_GS	0x0018	Result frame size (height and width)
LAY0_OOSFT	0x001C	Left-up corner offset from Destination picture's left-up corner (y-offset and x-offset)
LAY0_RSZ_COEF	0x0020	Resize Coefficients(h-resize-coefficient and v-resize coefficient)
LAY0_BK_ARGB	0x0024	Background aRGB
Chanel 1		
LAY1_CTRL	0x1000	Source format control register
LAY1_Y_ADDR	0x1004	Source Y (or RGB) base address
LAY1_U_ADDR	0x1008	Source U (or UV) base address
LAY1_V_ADDR	0x100C	Source V base address
LAY1_IN_FM_GS	0x1010	Input Geometric Size (height and width)
LAY1_STRIDE	0x1014	Source frame stride ((Y or RGB) and UV)
LAY1_OUT_GS	0x1018	Result frame size (height and width)
LAY1_OOSFT	0x101C	Left-up corner offset from Destination picture's left-up corner (y-offset and x-offset)
LAY1_RSZ_COEF	0x1020	Resize Coefficients(h-resize-coefficient and v-resize coefficient)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

LAY1_BK_ARGB	**0x1024**	Background aRGB
Chanel 2		
LAY2_CTRL	**0x2000**	Source format control register
LAY2_Y_ADDR	**0x2004**	Source Y (or RGB) base address
LAY2_U_ADDR	**0x2008**	Source U (or UV) base address
LAY2_V_ADDR	**0x200C**	Source V base address
LAY2_IN_FM_GS	**0x2010**	Input Geometric Size (height and width)
LAY2_STRIDE	**0x2014**	Source frame stride ((Y or RGB) and UV)
LAY2_OUT_GS	**0x2018**	Result frame size (height and width)
LAY2_OOSFT	**0x201C**	Left-up corner offset from Destination picture's left-up corner (y-offset and x-offset)
LAY2_RSZ_COEF	**0x2020**	Resize Coefficients(h-resize-coefficient and v-resize coefficient)
LAY2_BK_ARGB	**0x2024**	Background aRGB
Chanel 3		
LAY3_CTRL	**0x3000**	Source format control register
LAY3_Y_ADDR	**0x3004**	Source Y (or RGB) base address
LAY3_U_ADDR	**0x3008**	Source U (or UV) base address
LAY3_V_ADDR	**0x300C**	Source V base address
LAY3_IN_FM_GS	**0x3010**	Input Geometric Size (height and width)
LAY3_STRIDE	**0x3014**	Source frame stride ((Y or RGB) and UV)
LAY3_OUT_GS	**0x3018**	Result frame size (height and width)
LAY3_OOSFT	**0x301C**	Left-up corner offset from Destination picture's left-up corner (y-offset and x-offset)
LAY3_RSZ_COEF	**0x3020**	Resize Coefficients(h-resize-coefficient and v-resize coefficient)
LAY3_BK_ARGB	**0x3024**	Background aRGB

4.2.1 Global control Register

	GLB_CTRL																														0xF000			
Bit	31	30	29	28	27	26	25	24	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														SHARPL		reserved		BANK_SEL		reserved			L3_TLB	L2_TLB	L1_TLB	L0_TLB	CMD_TLB_EN	DMA_MOD	WDOG_EN	TLB_EN	IRQ_EN
RST	0	0	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	1	0	0	0

Bits	Name	Description	R/W
31:14	Reserved	Writing has no effect, read as zero.	R
17:16	SHARPL	Bicube resize sharp level selector: 0: lowest sharp level	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		3: highest sharp level	
15:14	Reserved		
13:12	BANK_SEL	ddr bank size selector: 1: 1K 2: 2K 3: 4K 0: reserved	RW
11:9	Reserved	Writing has no effect, read as zero.	R
8	L3_TLB	Lay3 source frame tlb enable. 0: disable, 1:enable	RW
7	L2_TLB	Lay2 source frame tlb enable. 0: disable, 1:enable	RW
6	L1_TLB	Lay1 source frame tlb enable. 0: disable, 1:enable	RW
5	L0_TLB	Lay0 source frame tlb enable. 0: disable, 1:enable	RW
4	CMD_TLB_EN	Command chain tlb endable: 0: disable, 1:enable	RW
3	DMA_MOD	DMA model selector. 0: no dma mode (not support) 1: dma mode (must)	RW
2	WDOG_EN	Watch dog enable. 1: enable; 0: disable	RW
1	DST_TLB_EN	Destination TLB enable. 1: enable; 0: disable	RW
0	IRQ_EN	Interrupt enable 1:enable; 0:disable	RW

4.2.2 Global Status Register

	GLB_STATUS																															0xF004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	X2D_ID																reserved										WD_ERR	IRQ	GLB_STA			
RST	1	1	1	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	R/W
31:16	X2D_ID	Writing has no effect, read as 0xE003.	R
15:6	Reserved	Writing has no effect, read as zero.	R
5	WD_ERR	Watch dog error, it will happen when watch dog's time validate.	R
4	IRQ	Interrupt flag: 0: no interrupt generator 1: interrupt is happen.	R
3:0	GLB_STA	X2d global status. 0: free	R

4.2.3 Global trigger Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	GLB_TRIG																														0xF008
Bit	31	30	29	28	27	26	25	24	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																												IRQ_CLR	RST	STOP	START
RST	0	0	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	R/W
3	IRQ_CLR	Clean the interrupt flag	W
2	RST	Reset X2D but register	W
1	STOP	Stop X2D immediately. 1: stop	W
0	START	Start X2D. 1: start	W

4.2.4 TLB address register

	TLB_BASE																														0xF00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TLB_BASE																														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	TLB_BASE	TLE base address	RW

4.2.5 Description chain address register

	DHA																														0xF010
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DHA																														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	DHA	Description chain's address.(Must DW align)	RW

Note: DHA will act only when the GLB_CTRL. DMA_MOD is 1.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4.2.6 Watch dog counter register

	WDOG_CNT																														0xF018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDOG_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	R/W
31:0	WDOG_CNT	Watch dog counter	RW

4.2.7 Destination Layers global control

	LAY_GCTRL																														0xE000
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																DST_TILE	LMSK													Lay_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

Bits	Name	Description	R/W
16	DST_TILE	Destination result input and output Tile-mode selector. 1: Tile mode , 0: Plane mode	RW
2:0	Lay_num	Indicate the total layer numbers , it only action in DMA_MOD.	RW

4.2.8 Destination picture base address register

	DST_BASE																														0xE004
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DST_BASE																														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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:0	DHA	Description chain's address.	RW

4.2.9 Destination picture control and stride register

	DST_CTRL_STR																														0xE008
Bit	31	30	29	28	27	26	25	24	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								Msk_en	preM	Glba_en	back_en	DST_STR																			
RST	0	0	0	0	0	0	0	0	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	Alpha	Destination global alpha value	RW
23	Msk_en	Destination will just act as an pure color background. 0: invalid 1: valid, this mode need not fetch data from external memory	RW
22	preM	Pre-multi indictor: 0: no pre-multi to RGB 1: pre-multi to RGB	RW
21	Glba_en	Destination buffer using global alpha, it will valid when back_en = 1. 0: pixel alpha 1: global alpha	RW
20	Back_en	Act the destination buffer as background picture	RW
19:0	DST_STR	Destination buffer's stride	RW

4.2.10 Destination picture Geometric size register

	DST_GS																														0xE00C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DST_HEIGHT																DST_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	R/W
31:16	DST_HEIGHT	Destination picture's height	RW
15:0	DST_WIDTH	Destination picture's width	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4.2.11 Destination picture mask value register

	DST_MASK_ARGB																0xE010													
Bit	31	30	29	28	27	26	25	24	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								R								G								B							
RST	0	0	0	0	0	0	0	0	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	ARGB	The value of the pixel when destination layer is masker	RW

4.2.12 Destination picture format register

	DST_FMT																												0xE014			
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																								APOS	Reserved		RGB_FMT		Reserved		RGB_ORD	
RST	0	0	0	0	0	0	0	0	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
31:5			
8	APOS	Alpha position 1: low position [7:0] , RGBA 0: high position [31:24] , ARGB	RW
7:6	Reserved	Writing has no effect, read as zero.	
5:4	RGB_FMT	0: ARGB888 1: XARGB888 2:RGB565 3:RGB555	RW
3	Reserved	Writing has no effect, read as zero.	
2:0	RGB_ORD	0: RGB 1:BGR 2:GRB 3:BRG 4:RBG 5:GBR	RW

4.2.13 Layer0 control Register

	LAY0_CTRL																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
	OSD_MOD				APOS	RGBM							VMIR	HMIR	ROTA		GLB_ALPHA									Msk_en	CSCM	PREM	AL_MOD	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:28	OSD_MOD	3: clear 1: source 2: destination 0: source over 4: destination over 5: source in 6: destination in 7: source out 8: destination out 9: source atop A: destination atop B: XOR	RW
27	APOS	Alpha position (only active when IN_FMT==0, alpha RGB888) 0: ARGB (A:[bit31~bit24]) 1: RGBA (A:[bit7~bit0])	
26:24	RGBM	RGB mode selector 0: RGB 1:BGR 2:GRB 3:BRG 4:RBG 5:GBR	R
23:20			
19	VMIR	Vertical mirror. 0:disable 1:enable	
18	HMIR	Horizontal mirror. 0: disable 1:enable	
17: 16	ROTA	Clockwise rotation selector 00: no rotation 01: 90 rotation 10: 180 rotation 11:270 rotation	R
15:8	GLB_ALPHA	Global alpha value (0 ~ 255) , valid when AL_MOD = 1 , or IN_FMT not ARGB888	RW
7			
6	MSK_EN	Lay will act as mask layer which just contain the same color. 0: Invalid 1: Valid, in this mod, the x2d will not need to fetch data from external memory, and all the pixel will be the background value	
5	CSCM	Color convention mode selector: 0: YUV => RGB 1: YCrCb => RGBv	
4	PreM	The source pre-multi indicator: 0: No pre-multi to the RGB pixel 1: Pre-multi to the RGB pixel	
3	AL_MOD	Alpha mode selector. 1: global alpha 0:pixel alpha(Only when IN_FMT is ARGB888)	RW
2:0	IN_FMT	Indicates the source data format. 00: Alpha RGB888 01: RGB555 10: RGB565 11: YUV420 separate	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		100: Tile YUV420 101: NV12 110: NV21	

4.2.14 Layer0 input Y plane Address Register

	LAY0_YADDR																													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
	LAY0_YADDR																														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	LAY0_YADDR	In separated Frame case, it indicates the source Y (or R) data buffer's start address. In source package case, it indicates the start Address of the packaged Frame.	RW

4.2.15 Layer0 input U plane Address Register

	LAY0_UADDR																													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
	LAY0_UADDR																														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	LAY0_UADDR	The source U data buffer's start address of separated frame case.	RW

4.2.16 Layer0 input V plane Address Register

	LAY0_VADDR																													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
	LAY0_VADDR																														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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	R/W
31:0	LAY0_VADDR	The source V data buffer's start address of separated Frame case.	RW

4.2.17 Layer0 source frame Geometric Size Register

	LAY0_SGS																														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
	IN_FM_H																IN_FM_W															
RST	0	0	0	0	0	0	0	0	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_H	The width of the input frame (Unit : pixel).	RW
15:0	IN_FM_W	The height of the input frame (Unit : pixel).	RW

4.2.18 Layer0 plane Stride Register

	LAY0_STR																														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
	UV_STR																Y_S														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
31:16	UV_STR	The line stride of the source UV data in the external(DDR) memory (Unit: byte) when YUV420. In YUV Tile, UV_STR should be the 1/16 to the stride of UV frame (UV_STR = UV_STR/16).	R
15:0	Y_S	The line stride of the source Y(or RGB) data in the external(DDR) memory (Unit: byte). In YUV Tile, Y_S should be the 1/16 to the stride of Y frame (Y_S = Y_S/16).	RW

4.2.19 Layer0 Output Geometric Size Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LAY0_OGS																													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
	OUT_FM_H																OUT_FM_W															
RST	0	0	0	0	0	0	0	0	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	OUT_FM_H	The width of the output destination frame (Unit: pixel).	RW
15:0	OUT_FM_W	The height of the output destination frame (Unit: pixel).	RW

4.2.20 Layer0 Output start pos offset Register

	LAY0_OOSF																													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
	OUT_H_OFST																OUT_W_OFST															
RST	0	0	0	0	0	0	0	0	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	OUT_H_OFST	The horizontal offset in the output destination frame (Unit: pixel).	RW
15:0	OUT_W_OFST	The vertical offset in the output destination frame (Unit: pixel).	RW

4.2.21 Layer0 scale ratio Register

	LAY0_RSZ																													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
	VRSZ																HRSZ															
RST	0	0	0	0	0	0	0	0	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	VRSZ	The scale ratio of vertical	RW
15:0	HRSZ	The scale ratio of horizontal	RW

*XRSZ = (uint16_t)((m/n)*512), m raw size, n: result size

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4.2.22 Layer0 background ARGB Register

	LAY0_BK_ARGB																													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
	Alpha								YR								UG								VB							
RST	0	0	0	0	0	0	0	0	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	Alpha	Background alpha	RW
23:16	YR	Background YR value (Y or R)	RW
15:8	UG	Background UG value (U or G)	RW
7:0	VB	Background VB value (V or B)	RW

4.3 Software stack

```
Structure X2D_CTRL_CHN
{
Uint32_t   overlay_num;      // Total layer numbers and Destination-Tile-enable , see spec.23.2.7
Uint32_t   * dst_addr;       // Destination picture address
Uint32_t   dst_ctrl_str;     // Destination control information and destination stride , see spec.23.2.9
Uint32_t   dst_gs;           // Destination picture's height and width
Uint32_t   dst_bargb;         //Destination global argb
Uint32_t   dst_fmt;           //Destination data format and rgb_mode
X2D_OVERLAY_INFO layer_chain[4];          // Chain information of each 4 layer
}

Structure X2D_OVERLAY_INFO
{
 Uint32_t   lay_ctrl;         // Layer control information
 Uint32_t   Y_addr;           // Y or RGB source's address
 Uint32_t   U_addr;           // U source's address when YUV420
 Uint32_t   V_addr;           // V source's address when YUV420
 Uint16_t   lay_swidth;       // Y or RGB source's width (Unit : pixel)
 Uint16_t   lay_sheight;      // Y or RGB source's height (Unit : pixel)
 Uint16_t   lay_ystrid;       // Y or RGB source's stride
 Uint16_t   lay_uvstrid;      // UV source's stride
```


Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

```
                        // (Unit : byte , when in Tile-YUV ,see spec.23.2.17)
  Uint16_t   lay_owidth;       // Destination's width
  Uint16_t   lay_oheight;      // Destination's height
  Uint16_t   lay_ox_offset;    // Destination's x-offset from background's left-up corner.
  Uint16_t   lay_oy_offset;    // Destination's y-offset from background's left-up corner.
  Uint16_t   lay_horiz_rsz_coef;        // Current layer's horizontal resize coefficient
                                        // RSZ = ( uint16_t )((m/n)*512), m raw size, n: result size
  Uint16_t   lay_verit_rsz_coef;        // Current layer's vertical resize coefficient
  Uint32_t   lay_bk_argb;               // Current layer's background color or common alpha when
layer mask enable
}
```


Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 3

DISPLAY/CAMERA/AUDIO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5 LCD Controller

5.1 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 use of a dedicated DMA controller. And OSD 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 (LCDC0 or LCDC1)
  - Support Delta RGB panel
  - Support SLCD panel

- OSD Supports
  - Supports two 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.2 Pin Description

Table 5-1 LCD Controller Pins Description

Name	I/O	Description
Lcd_pclk	Input/Output	Display device pixel clock
Lcd_vsync	Input/Output	Display device vertical synchronize pulse
Lcd_hsync	Input/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

NOTE: The mode and timing of special pin Lcd_spl, Lcd_cls, Lcd_ps and Lcd_rev can be seen in **part 5.7 LCD Controller Pin Mapping.**

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.3 Block Diagram

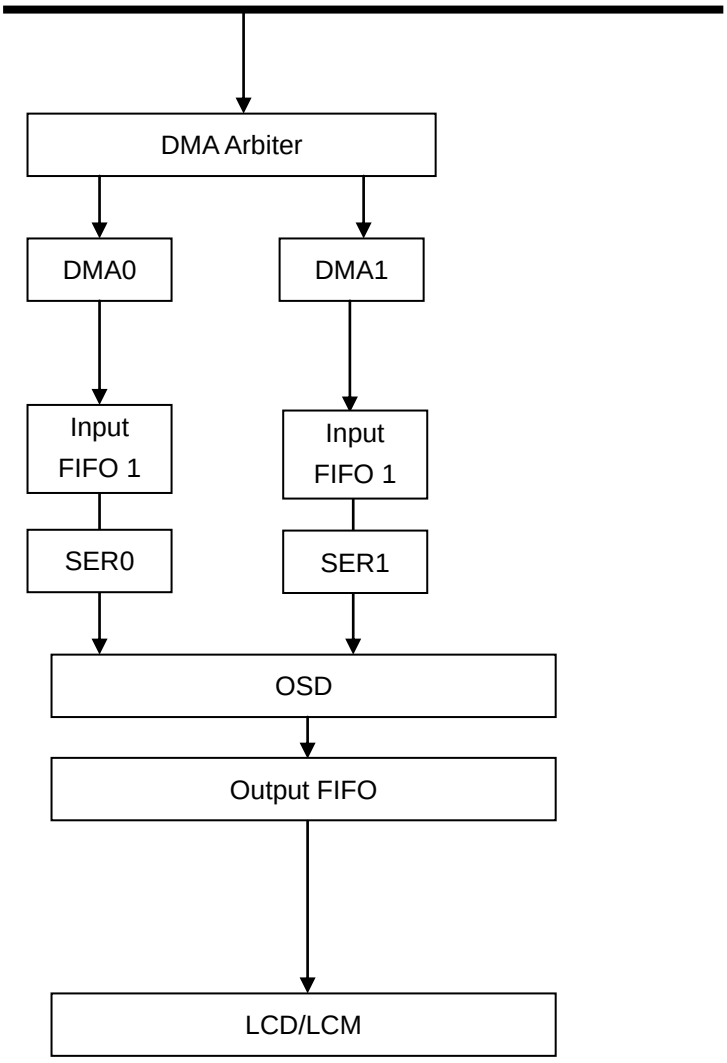


Figure 5-1 Block Diagram when use OSD mode

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

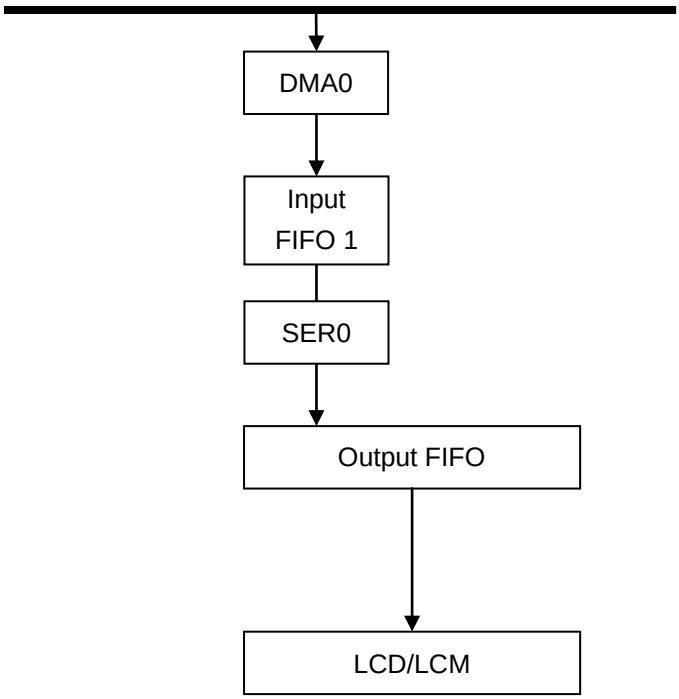


Figure 5-2 Block Diagram of TFT mode (not use OSD)

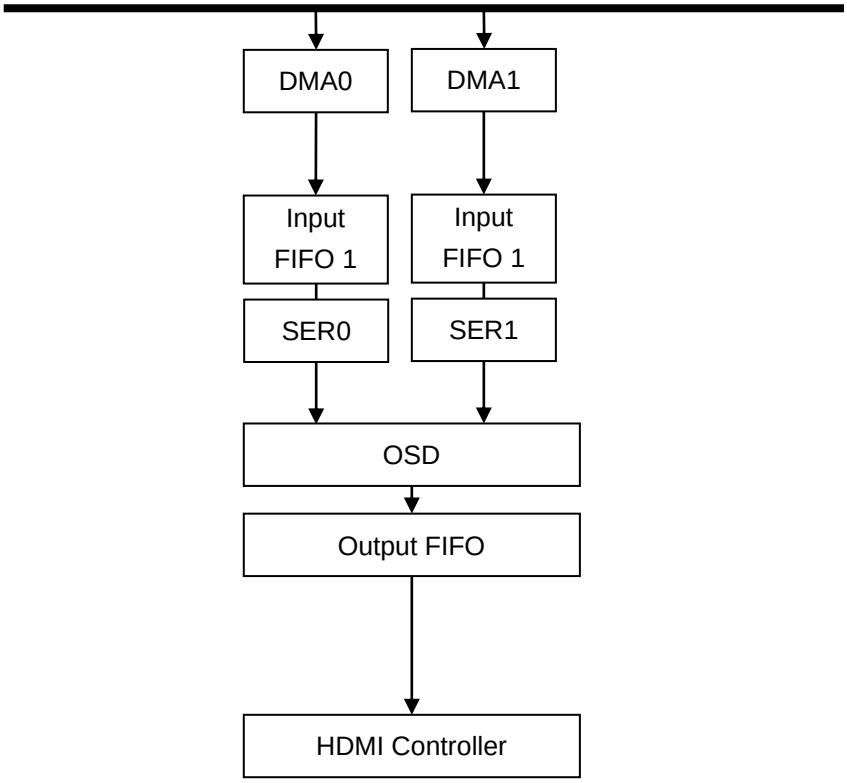


Figure 5-3 Block Diagram of HDMI interface

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.4 LCD Display Timing

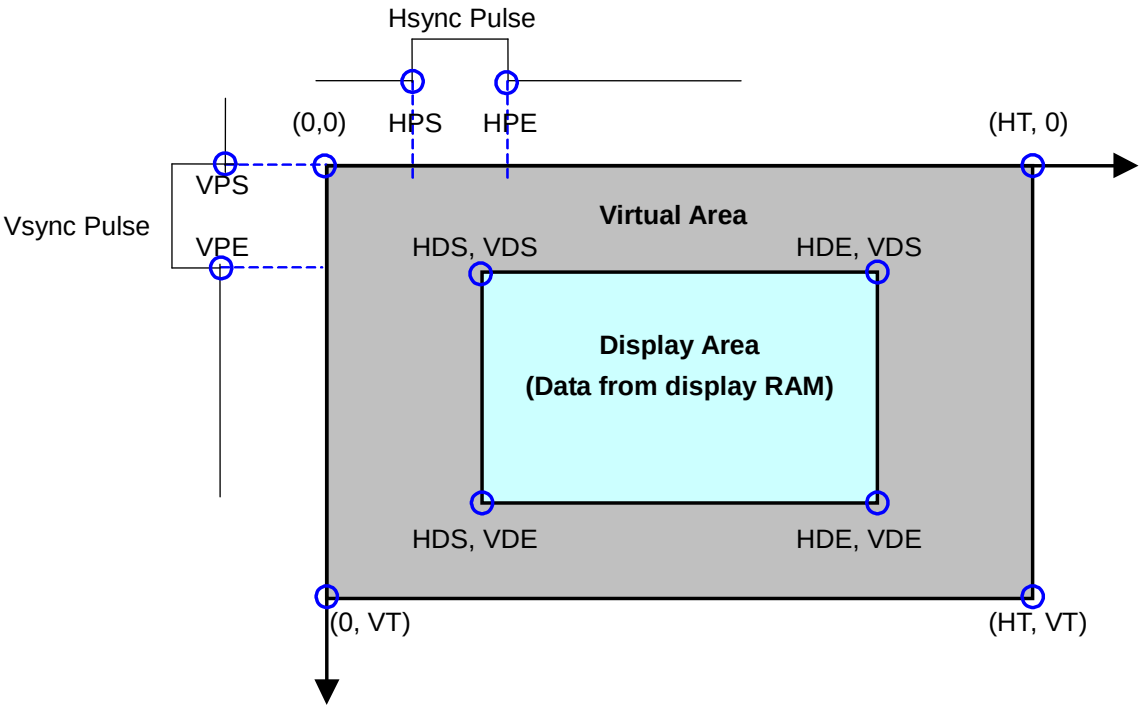


Figure 5-4 Display Parameters

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.5 OSD Graphic

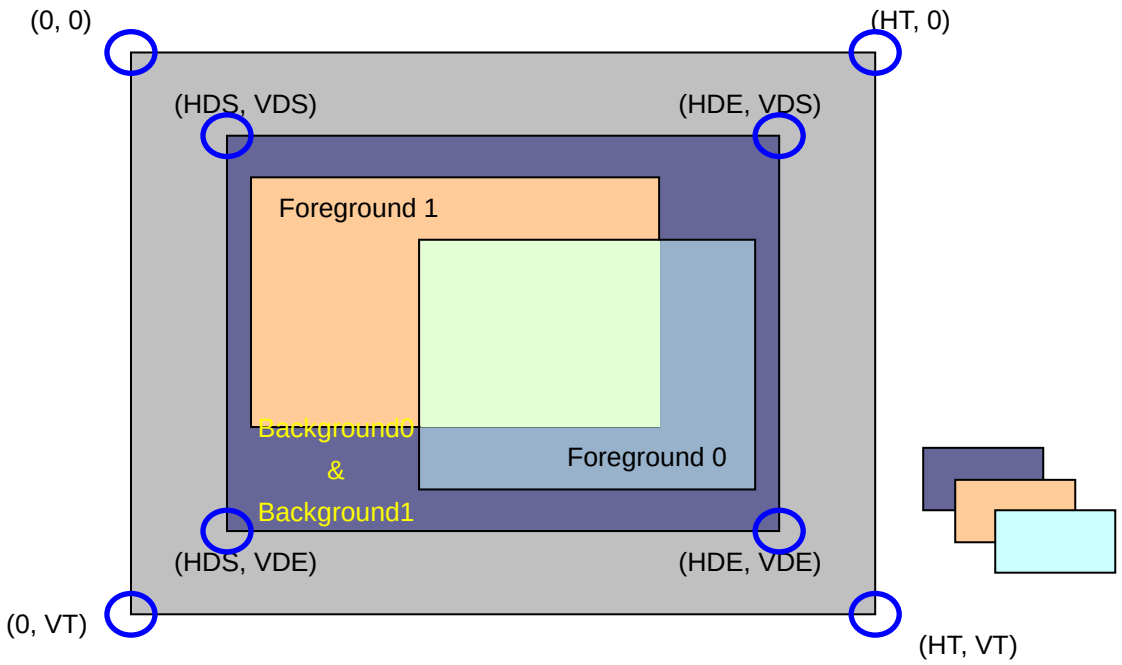


Figure 5-5 OSD Graphic

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.

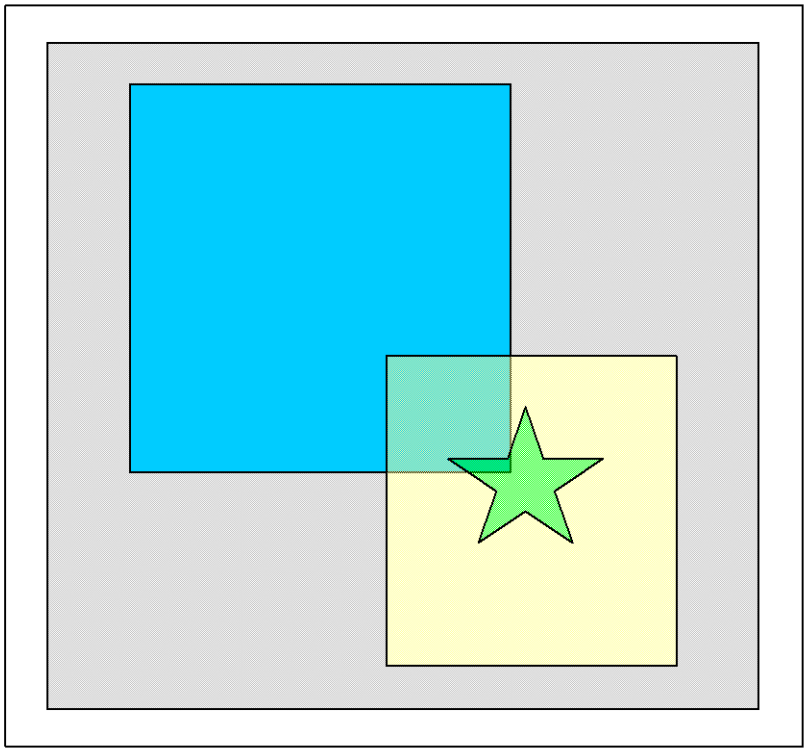
5.5.1 Color Key

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.

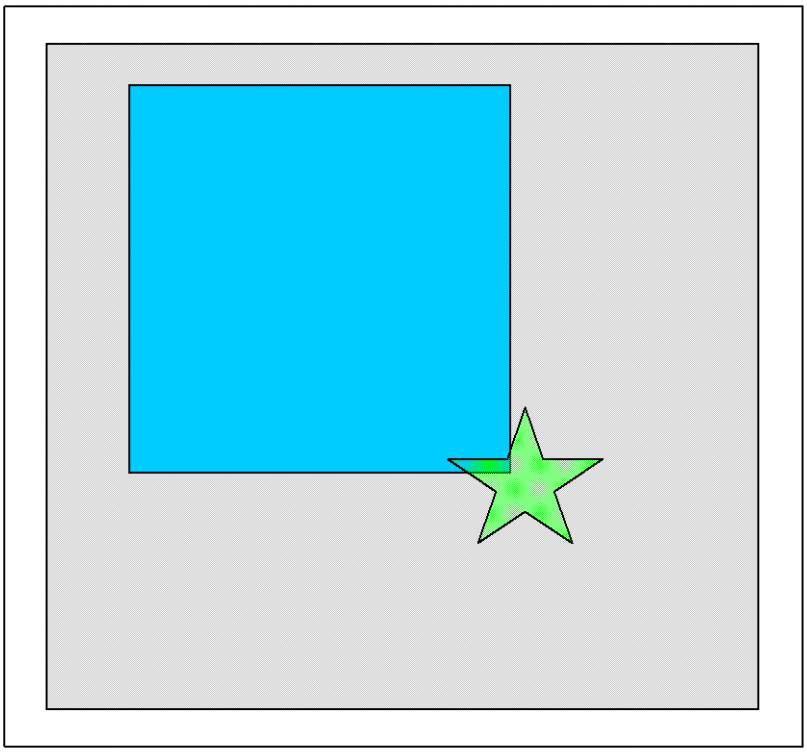
Color Key mode is meant to mask a chosen color and show others.

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

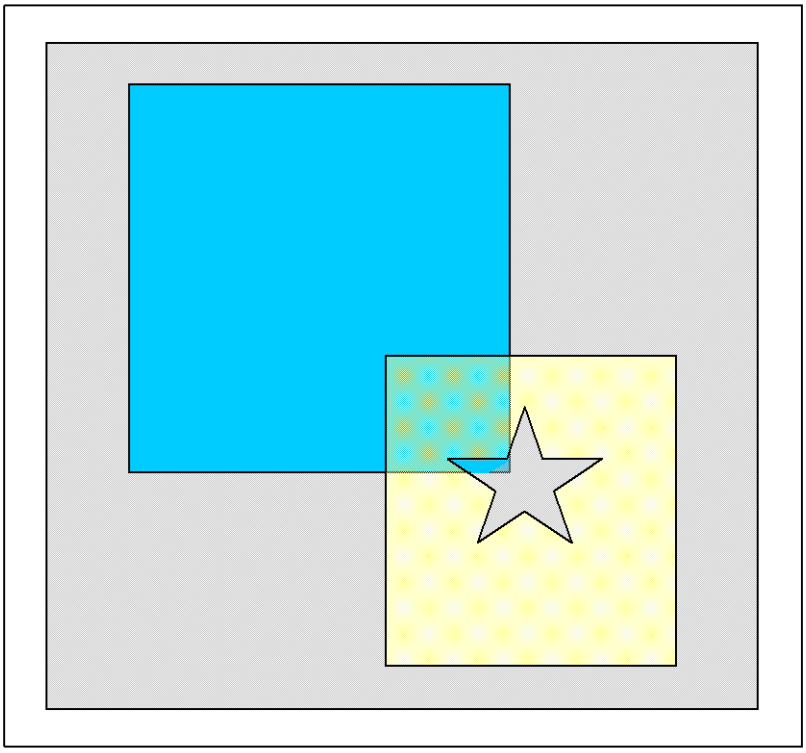


Not use color key function



Color key mode

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Mask color key mode

5.6 Register Description

0x13050000 is the base address of lcdc.

Table 5-2 LCD Controller Registers Description

Name	RW	Reset Value	Address offset	Access Size
LCDCFG	RW	0x00000000	0x0000	32
LCDCTRL	RW	0x00000000	0x0030	32
LCDSTATE	RW	0x00000000	0x0034	32
LCDOSDC	RW	0x0000	0x0100	32
LCDOSDCTRL	RW	0x0000	0x0104	32
LCDOSDS	RW	0x0000	0x0108	32
LCDBGC0	RW	0x00000000	0x010C	32
LCDBGC1	RW	0x00000000	0x02C4	32
LCDKEY0	RW	0x00000000	0x0110	32
LCDKEY1	RW	0x00000000	0x0114	32
LCDALPHA	RW	0x00	0x0118	32
LCDRGBC	RW	0x0000	0x0090	32
LCDVAT	RW	0x00000000	0x000C	32
LCDDAH	RW	0x00000000	0x0010	32
LCDDAV	RW	0x00000000	0x0014	32
LCDXYP0	RW	0x00000000	0x0120	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

LCDXYP1	RW	0x00000000	0x0124	32
LCDSIZE0	RW	0x00000000	0x0128	32
LCDSIZE1	RW	0x00000000	0x012C	32
LCDVSYNC	RW	0x00000000	0x0004	32
LCDHSYNC	RW	0x00000000	0x0008	32
LCDPS*1	RW	0x00000000	0x0018	32
LCDCLS*1	RW	0x00000000	0x001C	32
LCDSPL*1	RW	0x00000000	0x0020	32
LCDREV*1	RW	0x00000000	0x0024	32
LCDIID	R	0x00000000	0x0038	32
LCDDA0	RW	0x00000000	0x0040	32
LCDSA0	R	0x00000000	0x0044	32
LCDFID0	R	0x00000000	0x0048	32
LCDCMD0	R	0x00000000	0x004C	32
LCDOFFS0	R	0x00000000	0x0060	32
LCDPW0	R	0x00000000	0x0064	32
LCDCNUM0/LCDPOS0	R	0x00000000	0x0068	32
LCDDESSIZE0	R	0x00000000	0x006C	32
LCDDA1*2	RW	0x00000000	0x0050	32
LCDSA1*2	R	0x00000000	0x0054	32
LCDFID1*2	R	0x00000000	0x0058	32
LCDCMD1*2	R	0x00000000	0x005C	32
LCDOFFS1*2	R	0x00000000	0x0070	32
LCDPW1*2	R	0x00000000	0x0074	32
LCDCNUM1/LCDPOS1*2	R	0x00000000	0x0078	32
LCDDESSIZE1*2	R	0x00000000	0x007C	32
LCDPCFG	RW	0x00000000	0x02C0	32

NOTES:

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

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

5.6.1 Configure Register (LCDCFG)

	LCDCFG																															0x13050000
Bit	31	30	29	28	27	26	25	24	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			NEWDES			RECOVER		PSM	CLSM	SPLM	REVM	HSYNM	PCLKM	INVDAT	SYNDIR	PSP	CLSP	SPLP	REVP	HSP	PCP	DEP	VSP	18/16	24			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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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. **LCDPIN** / **PIN SELECT** 0 / LCD PIN 1 / SLCD PIN	RW
29		**KEEP THIS BIT TO 0.**	RW
28	NEWDES	indicate use new 8 words descriptor or not. Set this bit to 1	RW
27		**KEEP THIS BIT TO 0.**	RW
25	RECOVER	Auto recover when output FIFO under run. 0: disable; 1: enable.	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.	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
16	SYNDIR	V-Sync and H-Sync direction. 0: output; 1: input.	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.	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	18/16	18-bit TFT Panel or 16-bit TFT Panel. This bit will be available when MODE [3:2] is equal to 0 and 24[6] is equal to 0. 0: 16-bit TFT Panel 1: 18-bit TFT Panel	RW
6	24	Set this bit to 1 for 24-bit TFT Panel.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3:0	MODE	Display Device Mode Select/Output mode.	RW

	LCD Panel
0000	Generic 16-bit/18-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	LCM
1110	Reserved
1111	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 CS	SLCD RS	--	--	--	--	--	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.

5.6.2 Control Register (LCDCTRL)

	LCDCTRL																														0x13050030
Bit	31	30	29	28	27	26	25	24	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													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]	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>1: 16-bit data correspond with LCD_D[17:10], LCD_D[8:1]</td><td></td></tr>
<tr><td>30:28</td><td>BST</td><td>Burst Length Selection.<table><tr><th></th><th>Burst Length</th></tr><tr><td>000</td><td>4 word</td></tr><tr><td>001</td><td>8 word</td></tr><tr><td>010</td><td>16 word</td></tr><tr><td>011</td><td>32 word</td></tr><tr><td>100</td><td>64 word</td></tr></table></td><td>RW</td></tr>
<tr><td>27</td><td>OUTRGB</td><td>Bpp16 RGB output mode. 0: RGB565; 1: RGB555.
When 16-bit panel is used, this bit is valid</td><td>RW</td></tr>
<tr><td>26</td><td>OFUP</td><td>Output FIFO under run protection. 0: disable; 1: enable.</td><td>RW</td></tr>
<tr><td>23:16</td><td></td><td>keep these bits to 0.</td><td></td></tr>
<tr><td>15</td><td></td><td>keep this bit to 0.</td><td></td></tr>
<tr><td>13</td><td>EOFM</td><td>Mask end of frame interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>12</td><td>SOFM</td><td>Mask start of frame interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>11</td><td>OFUM</td><td>Mask out FIFO under run interrupt. 0: INT-disabled;1: INT-enabled.</td><td>RW</td></tr>
<tr><td>10</td><td>IFUM0</td><td>Mask in FIFO 0 under run interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>9</td><td>IFUM1</td><td>Mask in FIFO 1 under run interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>8</td><td>LDDM</td><td>Mask LCD disable done interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>7</td><td>QDM</td><td>Mask LCD quick disable done interrupt. 0: INT-disabled; 1: INT-enabled.</td><td>RW</td></tr>
<tr><td>6</td><td>BEDN</td><td>Endian selection. 0: same as system Endian; 1: reverse endian format.</td><td>RW</td></tr>
<tr><td>5</td><td>PEDN</td><td>Endian in byte. 0: msb first; 1: lsb first.</td><td>RW</td></tr>
<tr><td>4</td><td>DIS</td><td>Disable controller indicate bit. 0: enable; 1: in disabling or disabled.</td><td>RW</td></tr>
<tr><td>3</td><td>ENA</td><td>Enable controller. 0: disable; 1: enable.</td><td>W</td></tr>
<tr><td>2:0</td><td>BPP0</td><td>Bits Per Pixel of foreground0.<table><tr><th></th><th>Bits Per Pixel</th></tr><tr><td>000</td><td>Reserved</td></tr><tr><td>001</td><td>Reserved</td></tr><tr><td>010</td><td>Reserved</td></tr><tr><td>011</td><td>Reserved</td></tr><tr><td>100</td><td>15/16bpp</td></tr><tr><td>101</td><td>18bpp/24bpp</td></tr><tr><td>110</td><td>24bpp compressed</td></tr><tr><td>111</td><td>30bpp</td></tr></table></td><td>R</td></tr>
</table>

5.6.3 Status Register (LCDSTATE)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LCDSTATE																													0x13050034
Bit	31	30	29	28	27	26	25	24	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.	RW
6	Reserved	Writing has no effect, read as zero.	R
5	EOF	End of Frame indicate bit.	RW
4	SOF	Start of Frame indicate bit.	RW
3	OUF	Out FIFO under run.	RW
2	IFU0	In FIFO 0 under run.	RW
1	IFU1	In FIFO 1 under run.	RW
0	LDD	LCD disable. 0: not been normal disabled; 1: been normal disabled.	RW

5.6.4 OSD Configure Register (LCDOSDC)

	LCDOSDC																														0x13050100
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		SOFM1	EOFM1	Reserved	OSDIV	SOFM0	EOFM0	DENDM	Reserved				F1EN	F0EN	ALPHAEN	ALPHAMD0	OSDEN
RST									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	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 :	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		00 : 0 01 : 1 10 : alpha1 11 : 1-alpha1	
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
15	SOFM1	Start of frame interrupt mask for foreground 1.	RW
14	EOFM1	End of frame interrupt mask for foreground 1.	RW
13	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	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

5.6.5 OSD Control Register (LCDOSDCTRL)

	LCDOSDCTRL																0x13050104															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																											RGB0	RGB1		BPP1		
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	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.	R

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

5.6.6 OSD State Register (LCDOSDS)

	LCDOSDS																0x13050108															
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	R
7:0	Reserved	Writing has no effect, read as zero.	R

5.6.7 Background0 Color Register (LCDBGC0)

	LCDBGC0																																0x1305010C
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

5.6.8 Background1 Color Register (LCDBGC1)

	LCDBGC1																													0x130502C4
Bit	31	30	29	28	27	26	25	24	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

5.6.9 Foreground Color Key Register 0 (LCDKEY0)

	LCDKEY0																													0x13050110
Bit	31	30	29	28	27	26	25	24	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						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	Writing has no effect, read as zero.	R
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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.6.10 Foreground Color Key Register 1 (LCDKEY1)

	LCDKEY1																0x13050114													
Bit	31	30	29	28	27	26	25	24	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						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	Writing has no effect, read as zero.	R
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

5.6.11 ALPHA Register (LCDALPHA)

	LCDALPHA																0x13050118													
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	ALPHA 0[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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.6.12 RGB Control (LCDRGBC)

	LCDRGBC																0x13050090															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	RGBDM	DMM	Reserved						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
7	RGBFMT	RGB format enable : 0: disable; 1: enable.	RW
6:4	OddRGB	Odd line serial RGB data arrangement, useful for RGB serial mode only. *Please notice that you must set 000 when use 16/18parallel mode.	RW
3	Reserved	Writing has no effect, read as zero.	R
2:0	EvenRGB	Even line serial RGB data arrangement, useful for RGB serial mode only. *Please notice that you must set 000 when use 16/18parallel mode.	RW

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

	RGB mode
000	RGB
001	RBG
010	GRB
011	GBR
100	BRG

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			101	BGR		
			110	Reserved		
			111	Reserved		

5.6.13 Virtual Area Setting (LCDVAT)

	LCDVAT																														0x1305000C	
Bit	31	30	29	28	27	26	25	24	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

5.6.14 Display Area Horizontal Start/End Point (LCDDAH)

	LCDDAH																														0x13050010	
Bit	31	30	29	28	27	26	25	24	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
11:0	HDE	Horizontal display area end. (in dot clock)	RW

5.6.15 Display Area Vertical Start/End Point (LCDDAV)

	LCDDAV																														0x13050014	
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

5.6.16 Foreground 0 XY Position Register (LCDXYP0)

	LCDXYP0																														0x13050120	
Bit	31	30	29	28	27	26	25	24	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

5.6.17 Foreground 1 XY Position Register (LCDXYP1)

	LCDXYP1																														0x13050124	
Bit	31	30	29	28	27	26	25	24	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 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.6.18 Foreground 0 Size Register (LCDSIZE0)

	LCDSIZE0																0x13050128															
Bit	31	30	29	28	27	26	25	24	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

5.6.19 Foreground 1 Size Register (LCDSIZE1)

	LCDSIZE1																0x1305012c															
Bit	31	30	29	28	27	26	25	24	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

5.6.20 Vertical Synchronize Register (LCDVSYNC)

	LCDVSYNC																0x13050004															
Bit	31	30	29	28	27	26	25	24	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				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
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	VPS	V-Sync Pulse start position, fixed to 0. (in line clock)	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:12	Reserved	Writing has no effect, read as zero.	R
11:0	VPE	V-Sync Pulse end position. (in line clock)	RW

5.6.21 Horizontal Synchronize Register (LCDHSYNC)

	LCDHSYNC																0x13050008															
Bit	31	30	29	28	27	26	25	24	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
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	HPS	H-Sync pulse start position. (in dot clock)	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	HPE	H-Sync pulse end position. (in dot clock)	RW

5.6.22 PS Signal Setting (LCDPS)

	LCDPS																0x13050018															
Bit	31	30	29	28	27	26	25	24	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
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	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
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PSE	PS signal end position. (in dot clock) In STN mode, PSE defines N, which described in PSS.	RW

5.6.23 CLS Signal Setting (LCDCLS)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LCDCLS																															0x1305001C
Bit	31	30	29	28	27	26	25	24	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
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	CLSS	CLS signal start position. (in dot clock)	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	CLSE	CLS signal end position. (in dot clock)	RW

5.6.24 SPL Signal Setting (LCDSPL)

	LCDSPL																															0x13050020
Bit	31	30	29	28	27	26	25	24	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
31:28	Reserved	Writing has no effect, read as zero.	R
27:16	SPLS	SPL signal start position. (in dot clock)	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	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]).

5.6.25 REV Signal Setting (LCDREV)

	LCDREV																															0x13050024
Bit	31	30	29	28	27	26	25	24	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				REVS												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
31:28	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27:16	REVS	REV signal start position. (in dot clock)	RW
15:0	Reserved	Writing has no effect, read as zero.	R

5.6.26 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. For dual-panel displays, LCDIID is written only when both channels have reached a given state.

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																															0x13050038
Bit	31	30	29	28	27	26	25	24	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.	RW

5.6.27 Descriptor Address Registers (LCDDAx)

A frame descriptor is a 4-word block, aligned on 4-word (16-byte) 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.

WORD [3] contains the value for LCDCMDx.

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 4 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 16-byte boundary. Bits [3:0] of the address must be zero.

	LCDDA0, LCDDA1																0x13050040, 0x13050050															
Bit	31	30	29	28	27	26	25	24	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]: the buffer property. The value is same as LCDCMD	RW

5.6.28 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. 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																0x13050044, 0x13050054															
Bit	31	30	29	28	27	26	25	24	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

5.6.29 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LCDFID0, LCDFID1																0x13050048, 0x130500588															
Bit	31	30	29	28	27	26	25	24	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

5.6.30 DMA Command Registers (LCDCMDx)

	LCDCMD0, LCDCMD1																0x1305004C, 0x1305005C															
Bit	31	30	29	28	27	26	25	24	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			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

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		keep this bit to 0.	
26	FRM_EN	Indicat this frm is enable. 0:disable; 1:enable.	R
25	FIELD_S	Field select for interlace : 0:odd field or no interlace; 1: even field	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EL		
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

5.6.31 DMA OFFSIZE Registers (LCDOFFSx)

	LCDOFFS0, LCDOFFS1																0x13050060, 0x13050070															
Bit	31	30	29	28	27	26	25	24	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

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

5.6.32 DMA Page Width Registers (LCDPWx)

	LCDPW0, LCDPW1																0x13050064, 0x13050074															
Bit	31	30	29	28	27	26	25	24	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.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		* 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.	

5.6.33 DMA Commend Counter Registers (LCDCNUMx)

When LCDCMD.CMD = 1, **0x13050068, 0x13050078** is use as LCDCNUM0, 1 are not used now, set it to 0.

	LCDCNUM0, LCDCNUM1																0x13050068, 0x13050078													
Bit	31	30	29	28	27	26	25	24	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

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

5.6.34 DMA Commend Counter Registers (LCDCPOSx)

When LCDCMD.CMD = 0, **0x13050068, 0x13050078** is use as LCDPOS0, 1.

	LCDPOS0, LCDPOS1																0x13050068, 0x13050078													
Bit	31	30	29	28	27	26	25	24	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.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

5.6.35 Foreground x Size in Descriptor (LCDDESSIZEx)

When LCDCMD.CMD = 0, **0x1305006C, 0x1305007C** is use as LCDDESSIZE0, 1, to indicator the next frame foreground0, 1's size.

	LCDDESSIZE0, 1																**0x1305006C, 0x1305007C,0x130A006C, 0x130A007C**															
Bit	31	30	29	28	27	26	25	24	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

5.6.36 Priority level threshold configure Register (LCDPCFG)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	LCDPCFG																															0x130502C0
Bit	31	30	29	28	27	26	25	24	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

Bits	Name	Description	RW
31	Lcd_pri_md	Lcd priority mode. 0: use lcd dynamitic priority level; 1: use arbiter priority level.	RW
30:28	HP_BST	Highest priority Burst Length Selection.	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

	Burst Length
000	4 word
001	8 word
010	16 word
011	32 word
101	Contiue16
100	64 word
111	Disable

5.7 LCD Controller Pin Mapping

There are several mapping schemes for different LCD panels.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.7.1 TFT Pin Mapping

Pin	Generic 8-bit Serial TFT	Generic 18-bit Parallel TFT	Generic 16-bit Parallel TFT	24 bit Parallel
Lcd_pclk/ Slcd_clk	CLK	CLK	CLK	CLK
Lcd_vsync/Slcd_wr	VSYNC	VSYNC	VSYNC	VSYNC
Lcd_hsync/Slcd_dc	HSYNC	HSYNC	HSYNC	HSYNC
Lcd_de	DE	DE	DE	DE
Lcd_ps	-	-	-	
Lcd_cls	-	-	-	
Lcd_rev	-	-	-	
Lcd_spl	-	-	-	
Lcd_dat17	-	R5		R7
Lcd_dat16	-	R4		R6
Lcd_dat15	-	R3	R4	R5
Lcd_dat14	-	R2	R3	R4
Lcd_dat13	-	R1	R2	R3
Lcd_dat12	-	R0	R1	R2
Lcd_dat11	-	G5	R0	G7
Lcd_dat10	-	G4	G5	G6
Lcd_dat9	-	G3	G4	G5
Lcd_dat8	-	G2	G3	G4
Lcd_dat7	R7/G7/B7	G1	G2	G3
Lcd_dat6	R6/G6/B6	G0	G1	G2
Lcd_dat5	R5/G5/B5	B5	G0	B7
Lcd_dat4	R4/G4/B4	B4	B4	B6
Lcd_dat3	R3/G3/B3	B3	B3	B5
Lcd_dat2	R2/G2/B2	B2	B2	B4
Lcd_dat1	R1/G1/B1	B1	B1	B3
Lcd_dat0	R0/G0/B0	B0	B0	B2
Lcd_lo6_o[5]	0	0	0	R1
Lcd_lo6_o[4]	0	0	0	R0
Lcd_lo6_o[3]	0	0	0	G1
Lcd_lo6_o[2]	0	0	0	G0
Lcd_lo6_o[1]	0	0	0	B1
Lcd_lo6_o[0]	0	0	0	B0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.7.2 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_dat[23]	lcd_r1
Lcd_lo6_o[4] /Slcd_dat[22]	lcd_r0
Lcd_lo6_o[3] /Slcd_dat[21]	lcd_g1
Lcd_lo6_o[2] /Slcd_dat[20]	lcd_g0
Lcd_lo6_o[1] /Slcd_dat[19]	lcd_b1
Lcd_lo6_o[0] /Slcd_dat[18]	lcd_b0

5.8 Display Timing

5.8.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

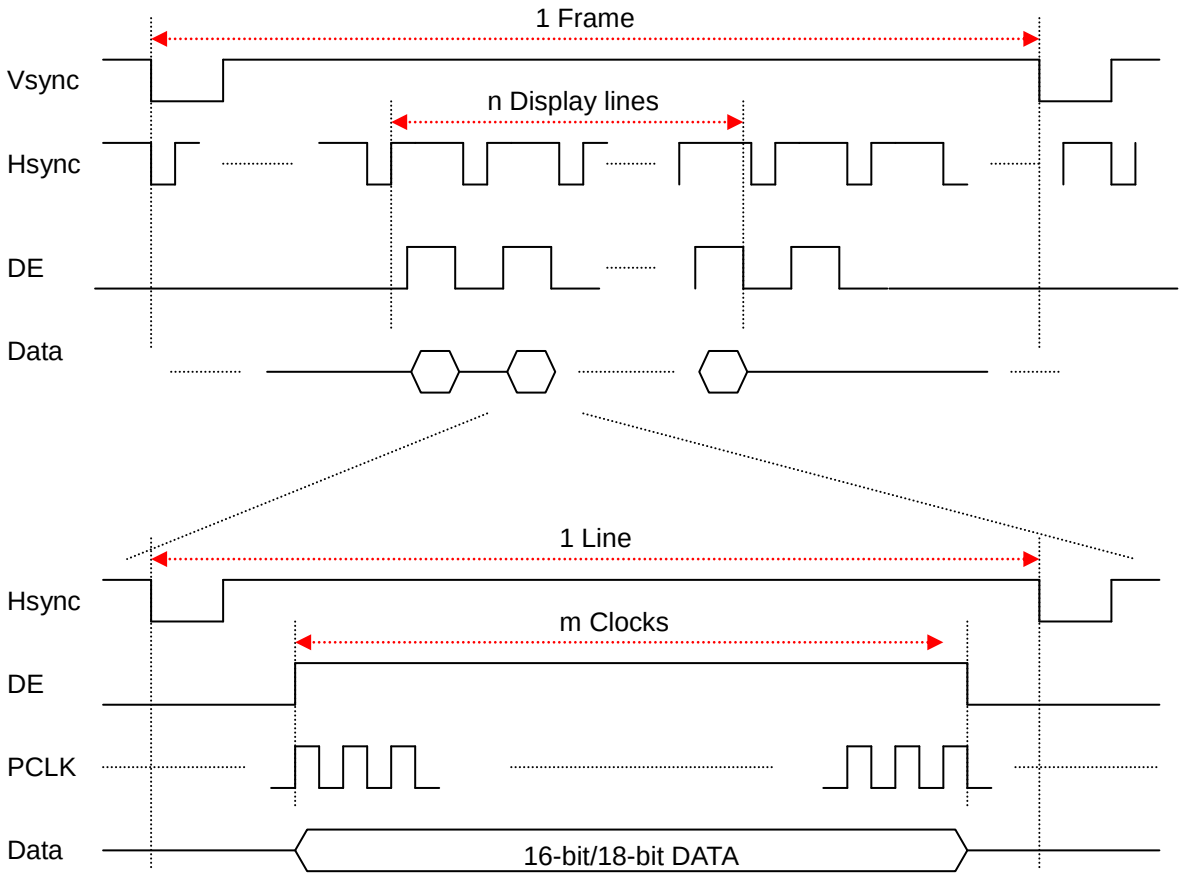


Figure 5-6 General 16-bit and 18-bit TFT LCD Timing

5.8.2 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

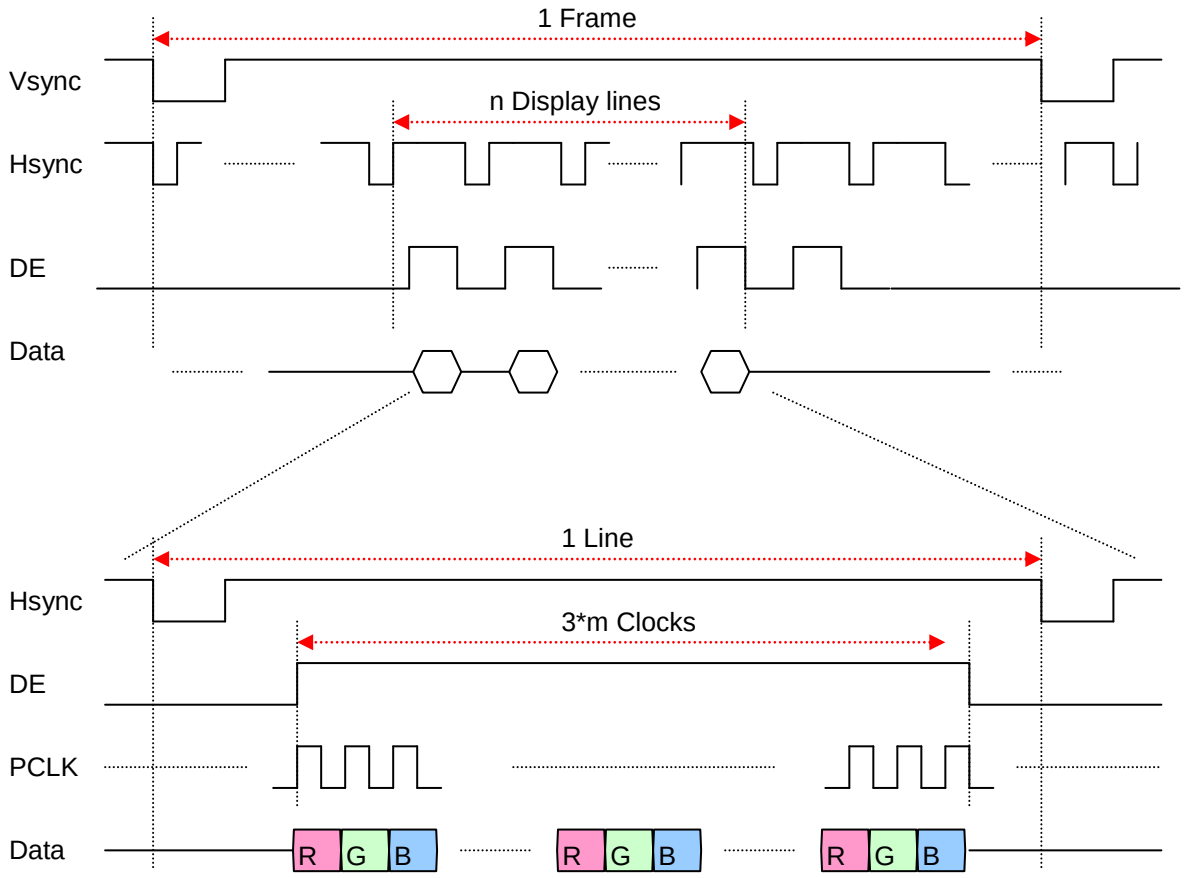


Figure 5-7 8-bit serial TFT LCD Timing (24bpp)

5.8.3 Special TFT Timing

Based on the general TFT LCD support, this controller also provides 4 special signals that can be programmed to general some special timing used for some panel. All 4 signals are worked in two modes: pulse mode and toggle mode. Signal “CLS” is fixed in pulse mode, and “REV” in toggle mode. The work mode of signals “SPL” and “PS” are defined in the special TFT LCD mode 1 to mode 3, either pulse mode or toggle mode. The position and polarity of these 4 signals can be programmed via registers. The Figures show the two modes as follows: (The toggle mode of signal “SPL” is different with the others signal. “SPL” does toggle after display line.)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

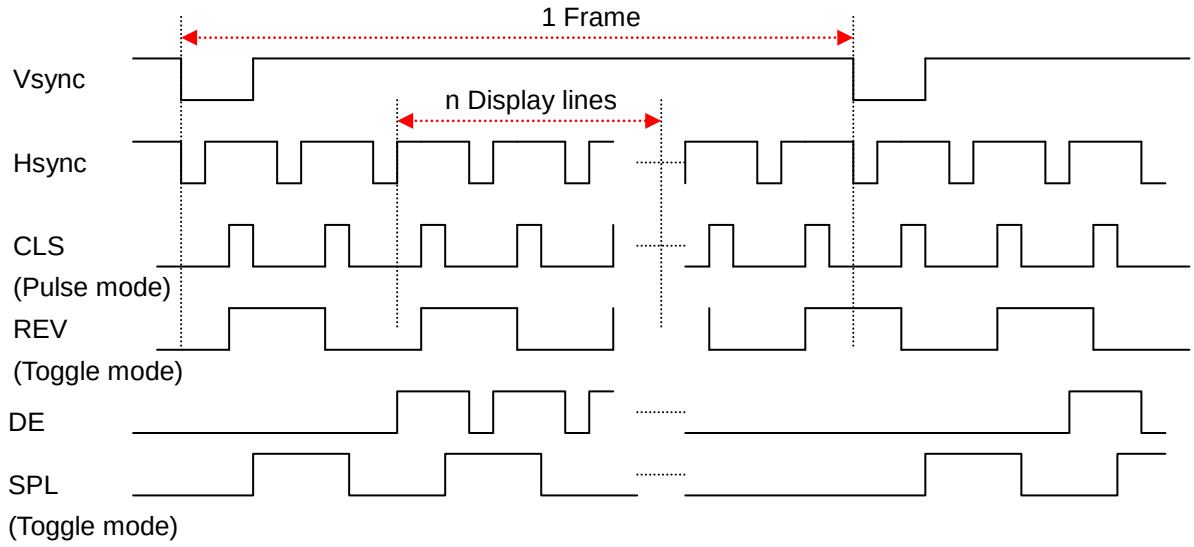


Figure 5-8 Special TFT LCD Timing 1

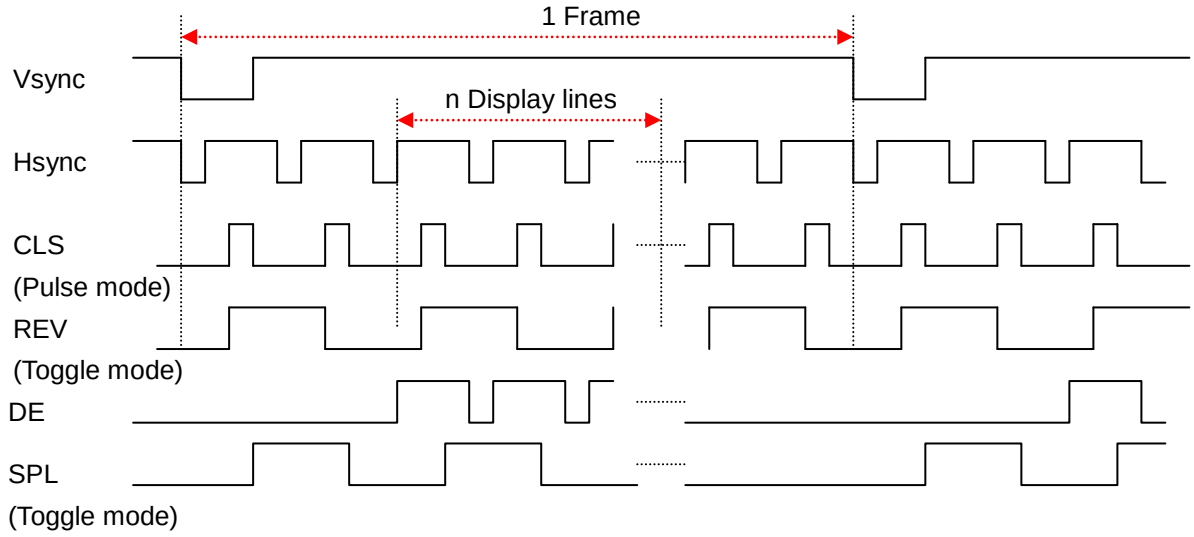


Figure 5-9 Special TFT LCD Timing 2

These two Figures show the timing of pulse mode and toggle mode, the pulse mode timing is same and the toggle mode timing is different. Timing 1 shows the condition when the total lines in 1 frame is odd (the number of display is even and the number of blank is odd), so the phase of REV inverse at the first line of each frame and the phase of SPL dose not inverse at the first line of each frame. Timing 2 shows the condition when the total lines in 1 frame is even (the number of display is even and the number of blank is even), so the phase of REV and SPL dose not inverse at the first line of each frame.

When LCDC is enabled ,there will be a null line to be add before transferring data to LCD panel. So the toggle mode exept SPL signal of special 3 TFT mode is when reset level is high,the first valid

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

edge will be rising edge. SPL signal of special 3 TFT mode is when reset level is high,the first valid edge will be falling edge.

5.8.4 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.

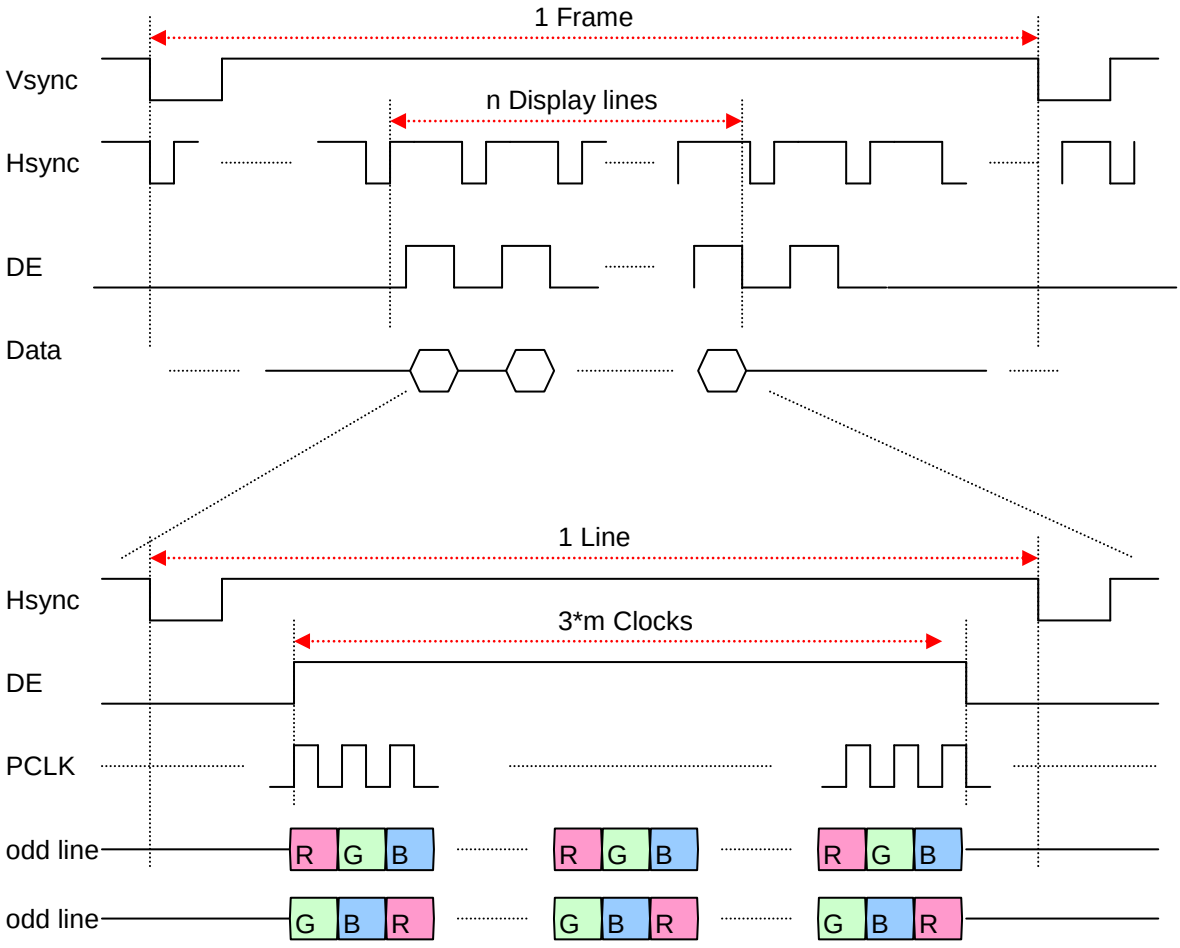
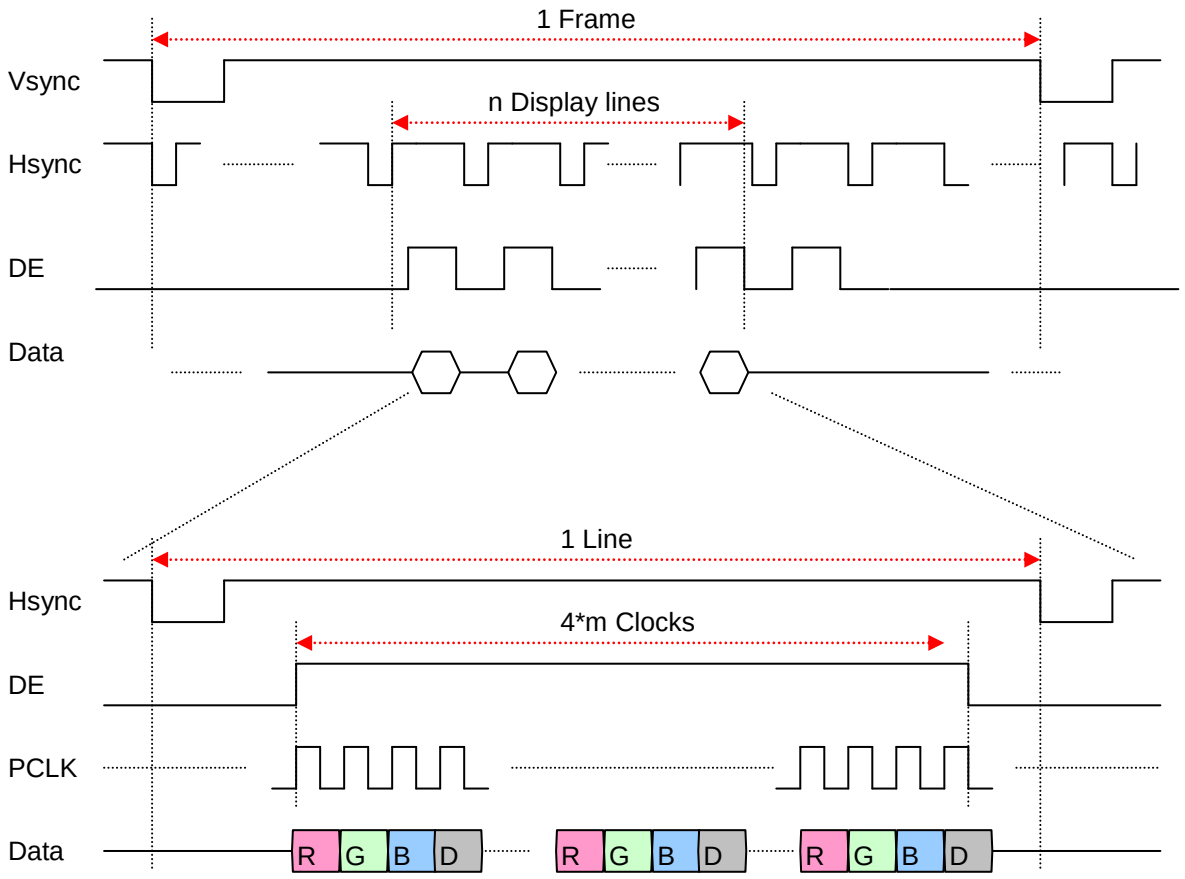


Figure 5-10 Delta RGB timing

5.8.5 RGB Dummy mode timing

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



***Dummy = 0**

Figure 5-11 RGB Dummy timing

5.9 Format of Frame Buffer

5.9.1 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

5.9.2 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5.9.3 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

5.9.4 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

5.9.5 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

5.9.6 24bpp 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	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

5.9.7 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]							

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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]							

5.10 Format of Data Pin Utilization

5.10.1 18-bit Parallel TFT

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

5.10.2 16-bit Parallel TFT

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

5.10.3 8-bit Serial TFT (24bpp)

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

5.11 LCD Controller Operation

5.11.1 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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:

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.

LCD panel determines the frequency of LCD pixel clock.

5.11.2 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.

5.11.3 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 PMC to shut down LCDC clock supply to save power consumption after disable LCDC. Please refer to PMC for detailed information.

5.11.4 Resetting the Controller

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

5.11.5 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

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.

5.11.6 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

* 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.

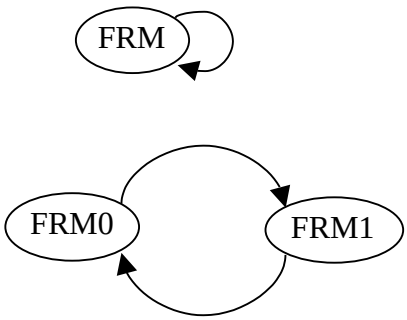
2 Reconfigure OSD.

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.

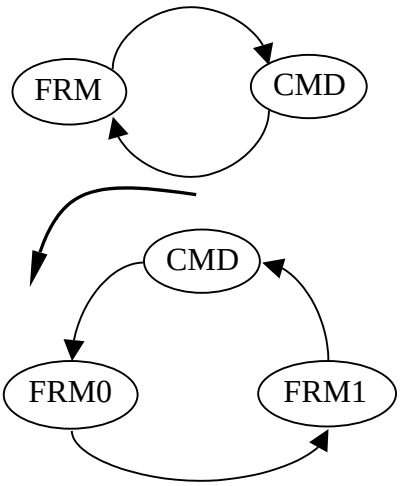
5.11.7 Descriptor Operation

1 TFT panel

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



2 SLCD



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6 Smart LCD Controller

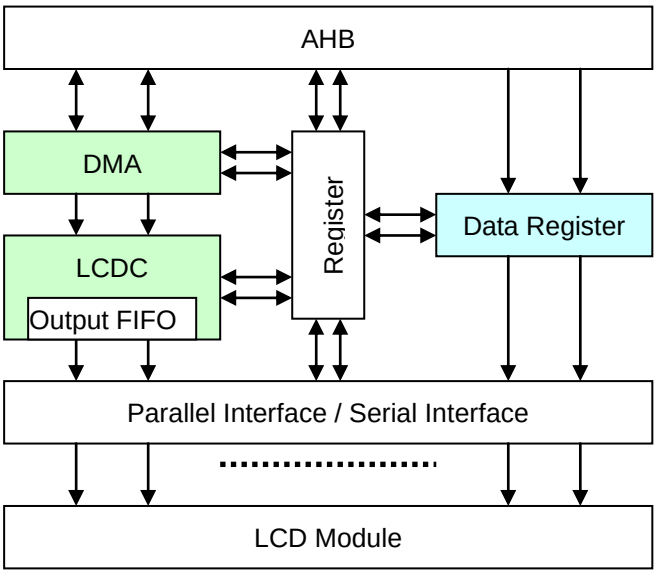
6.1 Overview

The Smart LCD Controller affords an interface to transfer data from the LCD controller to the LCD Module. 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

6.2 Structure



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.3 Pin Description

Table 6-1 SLCD Pins Description

Name	I/O	Description	Interface
SLCD_DC	O	Command/Data Select Signal. The polarity of the signal can be programmable.	Serial:DC Parallel: DC
SLCD_WR	O	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]	O	The data of SLCD. Relate to 1.7.2Data 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]	O	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.

6.4 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	SLCD Configure Register	RW	0x0000	0x130500A0	32
MCTRL	SLCD Control Register	RW	0x00	0x130500A4	32
MSTATE	SLCD Status Register	RW	0x00	0x130500A8	32
MDATA	SLCD Data Register	RW	0x00000000	0x130500AC	32

.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.4.1 SLCD Configure Register (MCFG)

The register MCFG is used to configure SLCD.

	MCFG																0x130500A0													
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved			DWIDTH			CWIDTH		Reserved			CSPLY	RSPLY	Reserved	CLKPLY	TYPE
RST																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:13	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 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 *Please notice that you can only use 24-bit parallel command when use 24-bit parallel data. (The command may not 24-bit but need put them as 24-bit in memory (one command use one word))	RW
7:5	Reserved	Writing has no effect, read as zero.	R
4	CSPLY	WR Polarity. (WR initial level will be different from WR Polarity) 0: Active Level is Low 1: Active Level is High	RW
3	RSPLY	DC Polarity. 0: Command DC = 0, Data DC = 1 1: Command DC = 1, Data DC = 0	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2	Reserved	Writing has no effect, read as zero.	R
1	CLKPLY	LCD_CLK Polarity. 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	--

6.4.2 SLCD Control Register (MCTRL)

MCTRL is SLCD Control Register.

	MCTRL																									0x130500A4							
Bit																										7	6	5	4	3	2	1	0
																										Reserved					DMASTART	DMAMODE	Reserved
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	DMAMODE	SLCD descriptor DMA mode select.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: DMA will continually transfer data follow descriptor chain 1: DMA will stop when one descriptor finished	
1	DMASTART	Only use when DMAMODE = 1, set 1 to restart DMA transfer.	RW
0	Reserved	Writing has no effect, read as zero.	R

6.4.3 SLCD Status Register (MSTATE)

The register of MSTATE is SLCD status register.

	MSTATE																								0x130500A8							
Bit																									7	6	5	4	3	2	1	0
	Reserved																														BUSY	
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	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	RW

6.4.4 SLCD Data Register (MDATA)

The register MDATA is used to send command or data to LCM. When RS=0, the low 24-bit is used as command. When RS=1, the low 24-bit is used as data.

	MDATA																														0x130500AC							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RS	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	RS	The DC bit of data register is used to decide the meanings of the low 24-bit. 0: data 1: command	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30:24	Reserved	Writing has no effect, read as zero.	R
23:0	DATA/CMD	Data or Command Register.	RW

6.5 System Memory Format

6.5.1 Data format

You can configure these registers according to LCDC module.

6.5.2 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

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

6.6 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.

6.6.1 DMA Transfer Mode

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

1 Command and Data

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 Configuration.
* LCDRGBC

2 Set Color.
* LCDBGC, LCDKEY0, LCDKEY1, LCDALHPA

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 Set Descriptors.

* LCDIID

* LCDDA0, LCDSA0, LCDFID0, LCDCMD0, LCDOFFS0, LCDPW0, LCDCNUM0/LCDPOS0, LCDDESSIZE0

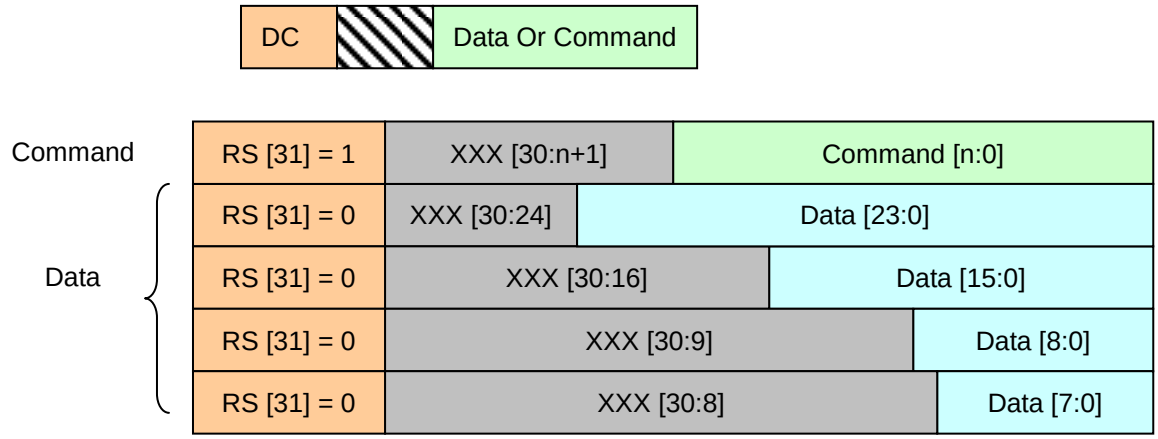
* LCDDA1, LCDSA1, LCDFID1, LCDCMD1, LCDOFFS1, LCDPW1, LCDCNUM1/LCDPOS1, LCDDESSIZE1

5 Enable slcd DMA.

6 Enable LCDC.

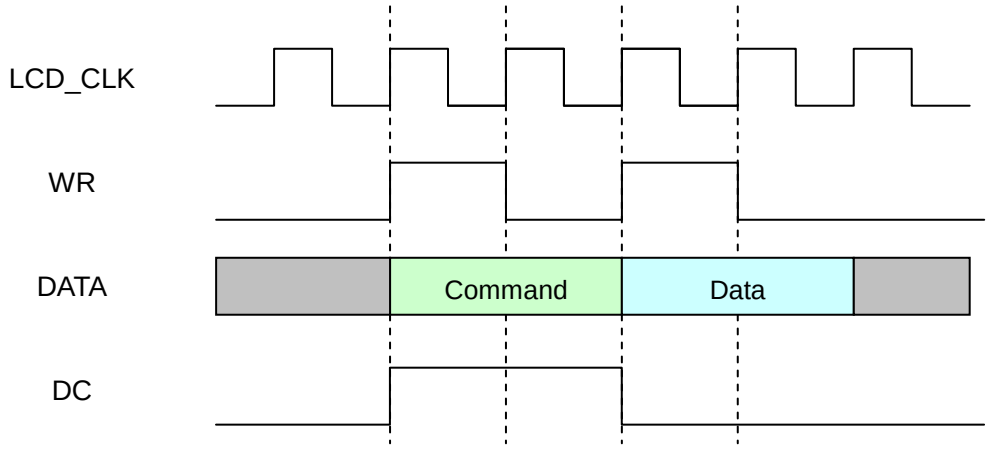
6.6.2 Register Transfer Mode

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 LCM. Command and data can be recognized by RS bit coming from data register. The format of data register transfer can be as follows:



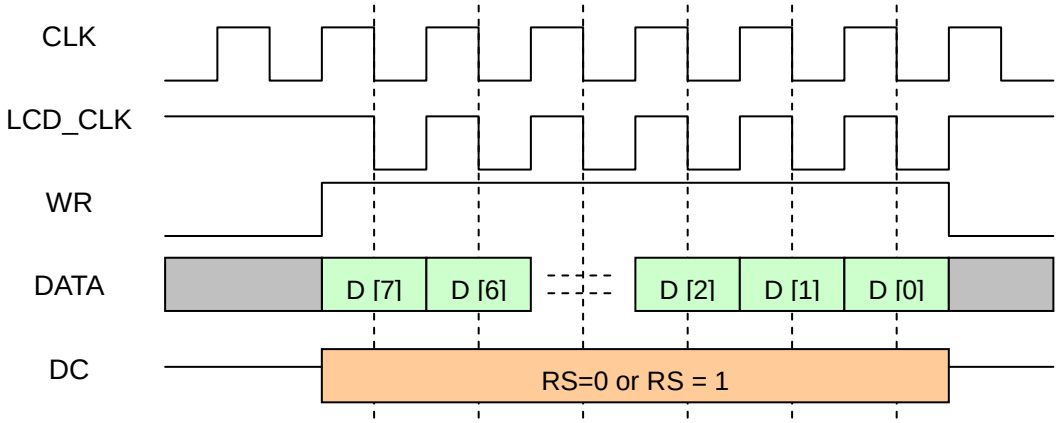
6.7 Timing

6.7.1 Parallel Timing



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6.7.2 Serial Timing



6.8 Operation Guide

6.8.1 DMA Operation

1 Start DMA transfer.

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.

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.

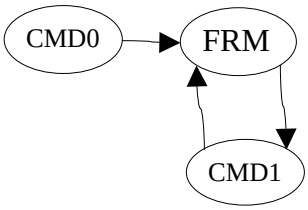
h Set LCDCCTRL.ENA to 1 to start LCDC internal DMA.

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):

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



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.

6.8.2 Register Operation

1 Set MCFG to configure SLCD.
2 Wait for MSTATE.BUSY == 0.
3 Set MDATA register.
4 Wait for MSTATE.BUSY == 0.
5 Set MDATA register.
6 Wait for MSTATE.BUSY == 0.
7 … …

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7 EPD Controller

7.1 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

7.2 EPDC Pin Mappings

Table 7-1 EPDC Pin Mapping

TYPE1	TYPE2	PIN
GDCLK	YCLK	PC19
GDRL	UD	PC00
GDSP	YDIOU	PC01
GDOE	YOE	PC02
SDCLK	XCLK	PC08
SDOE		PC09
SDLE	LD	PC18
SDRL	RL	PC03
SDCE [7]	YDIOD	PC21
SDCE [6]	VCOM[1]	PC20
SDCE [5]	VCOM[0]	PB15
SDCE [4]		PB14
SDCE [3]		PB13
SDCE [2]		PB12
SDCE [1]	XDIOR	PC11
SDCE [0]	XDIOL	PC10
SDDO [15]	DATA [15]	PC27
SDDO [14]	DATA [14]	PC26
SDDO [13]	DATA [13]	PC25
SDDO [12]	DATA [12]	PC24

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

SDDO [11]	DATA [11]	PC23
SDDO [10]	DATA [10]	PC22
SDDO [9]	DATA [9]	PC17
SDDO [8]	DATA [8]	PC16
SDDO [7]	DATA [7]	PC15
SDDO [6]	DATA [6]	PC14
SDDO [5]	DATA [5]	PC13
SDDO [4]	DATA [4]	PC12
SDDO [3]	DATA [3]	PC07
SDDO [2]	DATA [2]	PC06
SDDO [1]	DATA [1]	PC05
SDDO [0]	DATA [0]	PC04
PWRCOM		PB09
PWR7	PWR7	PF11
PWR6	PWR6	PF10
PWR5	PWR5	PF09
PWR4	PWR4	PF08
PWR3	PWR3	PB17
PWR2	PWR2	PB16
PWR1	PWR1	PB11
PWR0	PWR0	PB10
BD[3]	BD[3]	
BD[2]	BD[2]	
BD[1]	BD[1]	
BD[0]	BD[0]	

7.3 EPD Controller Registers

Table 7-2 EPD Controller Registers

Name	Address	Reset Value	Access Size	RW
EPDC_CTRL	0x130C0000	0x00000000	32	RW
EPDC_CFG	0x130C0004	0x00000000	32	RW
EPDC_STA	0x130C0008	0x00000000	32	RW
EPDC_ISRC	0x130C000C	0x00000000	32	RW
EPDC_DIS0	0x130C0010	0x00000000	32	RW
EPDC_DIS1	0x130C0014	0x00000000	32	RW
EPDC_SIZE	0x130C0018	0x00000000	32	RW
EPDC_VAT	0x130C0020	0x00000000	32	RW
EPDC_DAV	0x130C0024	0x00000000	32	RW
EPDC_DAH	0x130C0028	0x00000000	32	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

EPDC_VSYN	0x130C002C	0x00000000	32	RW
EPDC_HSYN	0x130C0030	0x00000000	32	RW
EPDC_GDCLK	0x130C0034	0x00000000	32	RW
EPDC_GDOE	0x130C0038	0x00000000	32	RW
EPDC_GDSP	0x130C003C	0x00000000	32	RW
EPDC_SDOE	0x130C0040	0x00000000	32	RW
EPDC_SDSP	0x130C0044	0x00000000	32	RW
EPDC_PMGR0	0x130C0050	0x00000000	32	RW
EPDC_PMGR1	0x130C0054	0x00000000	32	RW
EPDC_PMGR2	0x130C0058	0x00000000	32	RW
EPDC_PMGR3	0x130C005C	0x00000000	32	RW
EPDC_PMGR4	0x130C0060	0x00000000	32	RW
EPDC_LUTBF	0x130C0070	0x00000000	32	RW
EPDC_LUTSIZE	0x130C0074	0x00000000	32	RW
EPDC_CURBF	0x130C0080	0x00000000	32	RW
EPDC_CURSIZE	0x130C0084	0x00000000	32	RW
EPDC_WORK0BF	0x130C0090	0x00000000	32	RW
EPDC_WORK0SIZE	0x130C0094	0x00000000	32	RW
EPDC_WORK1BF	0x130C0098	0x00000000	32	RW
EPDC_WORK1SIZE	0x130C009C	0x00000000	32	RW
EPDC_VCOMBD0	0x130C0100	0x00000000	32	RW
EPDC_VCOMBD1	0x130C0104	0x00000000	32	RW
EPDC_VCOMBD2	0x130C0108	0x00000000	32	RW
EPDC_VCOMBD3	0x130C010C	0x00000000	32	RW
EPDC_VCOMBD4	0x130C0110	0x00000000	32	RW
EPDC_VCOMBD5	0x130C0114	0x00000000	32	RW
EPDC_VCOMBD6	0x130C0118	0x00000000	32	RW
EPDC_VCOMBD7	0x130C011C	0x00000000	32	RW
EPDC_VCOMBD8	0x130C0120	0x00000000	32	RW
EPDC_VCOMBD9	0x130C0124	0x00000000	32	RW
EPDC_VCOMBD10	0x130C0128	0x00000000	32	RW
EPDC_VCOMBD11	0x130C012C	0x00000000	32	RW
EPDC_VCOMBD12	0x130C0130	0x00000000	32	RW
EPDC_VCOMBD13	0x130C0134	0x00000000	32	RW
EPDC_VCOMBD14	0x130C0138	0x00000000	32	RW
EPDC_VCOMBD15	0x130C013C	0x00000000	32	RW
EPDC_PRI	0x130C0410	0x00000000	32	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4 Registers Description

7.4.1 EPDC Control Registers

	EPDC_CTRL																													0x130C0000
Bit	31	30	29	28	27	26	25	24	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.	RW
5	PWROFF	Start the power off sequence, cleared by HW when it finished.	RW
4	PWRON	Start the power on sequence, cleared by HW when it finished.	RW
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 disable when AUTO_STOP is 1.	RW
2	REF_STRT	Start refreshing process, cleared by HW when it stopped. This bit will be disable when AUTO_STOP is 1.	RW
1	LUT_STRT	Start LUT loading, cleared by HW when it finished.	RW
0	EPD_ENA	Enable EPD controller output.	RW

7.4.2 EPDC Configuration Registers

	EPDC_CFG																													0x130C0004
Bit	31	30	29	28	27	26	25	24	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 :	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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: 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 stopped ,the dma clock and core clock are gated. 0: When epdc stopped ,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

7.4.3 EPDC Status Register

	EPDC_STA																														
																															0x130C0008
Bit	31	30	29	28	27	26	25	24	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_STRT	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

7.4.4 EPDC ISR Control Register

	EPDC_ISRC																													0x130C000C
Bit	31	30	29	28	27	26	25	24	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

7.4.5 EPDC DISPLAY0 Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_DIS0																													0x130C0010
Bit	31	30	29	28	27	26	25	24	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	GDOE_MODE		GDUD	SDRL	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

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	GDUD	Up/Down pulse direction control and setting concatenating sequence for the Gate Driver.	RW
20	SDRL	Left/Right pulse direction control and setting concatenating sequence for the Source Driver.	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 YDIOD 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 1: 16-bit	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4.6 EPDC DISPLAY1 Register

	EPDC_DIS1																0x130C0014													
Bit	31	30	29	28	27	26	25	24	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

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 = 0, SDRL = 0: padding after the display data PDAT_SWAP = 0, SDRL = 1: padding before the display data PDAT_SWAP = 1, SDRL = 0: padding before the display data PDAT_SWAP = 1, SDRL = 1: padding after 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. For instance, if SDCE_NUM = 4, SDCE_STN = 0, controller output chips enable [SDCE0, SDCE1, SDCE2, SDCE3] in order. But if SDCE_REV = 1, outputs will be [SDCE3, SDCE2, 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

7.4.7 EPDC SIZE Setting Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_SIZE																															0x130C0018	
Bit	31	30	29	28	27	26	25	24	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

7.4.8 EPDC Virtual Display Area Setting Register

	EPDC_VAT																															0x130C0020	
Bit	31	30	29	28	27	26	25	24	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

7.4.9 EPDC Vertical Display Area Setting Register

	EPDC_DAV																															0x130C0024	
Bit	31	30	29	28	27	26	25	24	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

7.4.10 EPDC Horizontal Display Area Setting Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_DAH																															0x130C0028
Bit	31	30	29	28	27	26	25	24	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

7.4.11 EPDC Vertical Synchronous Start Pulse Setting

	EPDC_VSYN																															0x130C002C
Bit	31	30	29	28	27	26	25	24	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

7.4.12 EPDC Horizontal Synchronous Start Pulse Setting

	EPDC_HSYN																															0x130C0030
Bit	31	30	29	28	27	26	25	24	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

7.4.13 EPDC Gate Driver Clock Setting Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_GDCLK																															0x130C0034
Bit	31	30	29	28	27	26	25	24	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

7.4.14 EPDC Gate Output Enable Setting Register

	EPDC_GDOE																															0x130C0038
Bit	31	30	29	28	27	26	25	24	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

7.4.15 EPDC Gate Driver Start Pulse Setting

	EPDC_GDSP																															0x130C003C
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4.16 EPDC Source Driver Output Enable Setting Register

	EPD_SDOE																															0x130C0040
Bit	31	30	29	28	27	26	25	24	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

7.4.17 EPDC Source Driver Start Pulse Setting Register

	EPDC_SDSP																															0x130C0044
Bit	31	30	29	28	27	26	25	24	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

7.4.18 EPDC Power Management Registers 0

	EPDC_PMGR0																															0x130C0050
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

7.4.19 EPDC Power Management Registers 1

	EPDC_PMGR1																														0x130C0054	
Bit	31	30	29	28	27	26	25	24	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
27:16	PWR_DLY34	The delay time in lines between PWR [3] and PWR [4].	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

7.4.20 EPDC Power Management Registers 2

	EPDC_PMGR2																														0x130C0058	
Bit	31	30	29	28	27	26	25	24	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_DLY56												Reserved				PWR_DLY45											
RST	0	0	0	0	0	0	0	0	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_DLY56	The delay time in lines between PWR [5] and PWR [6].	RW
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PWR_DLY45	The delay time in lines between PWR [4] and PWR [5].	RW

7.4.21 EPDC Power Management Registers 3

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_PMGR3																															0x130C005C
Bit	31	30	29	28	27	26	25	24	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	VCOM [1:0] default value when idle.	RW
29	PWRCOM_POL	The polarity of PWRCOM.	RW
28	UNIPOL	This bit choose PWRCOM or VCOM [1:0] as common voltage control signals of Source Driver. 0: PWRCOM 1: VCOM [1:0]	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 PWR7~0.	R
15:12	Reserved	Writing has no effect, read as zero.	R
11:0	PWR_DLY67	The delay time in lines between PWR [6] and PWR [7].	RW

7.4.22 EPDC Power Management Registers 4

	EPDC_PMGR4																															0x130C0060
Bit	31	30	29	28	27	26	25	24	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
23:16	PWR_ENA	These bits enable PWR7~0 individually.	W
15:8	VCOMBD_STEP	VCOM/BORDER steps	RW
7:0	PWR_VAL	The PWR [x] pin value individually if it is not enabled.	RW

7.4.23 EPDC LUT Base address Registers

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EPDC_LUTBF																															
																	0x130C0070															
Bit	31	30	29	28	27	26	25	24	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

7.4.24 EPDC LUT Size Registers

	EPDC_LUTSIZE																															
																	0x130C0074															
Bit	31	30	29	28	27	26	25	24	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														EPDC_LUTSIZE															
RST	0	0	0	0	0	0	0	0	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	
12:0	EPDC_LUTSIZE	LUT buffer size, count in word.	RW

7.4.25 EPDC Current Buffer Base address Registers

	EPDC_CURBF																															
																	0x130C0080															
Bit	31	30	29	28	27	26	25	24	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

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4.26 EPDC Current Buffer Size Registers

	EPDC_CURSIZE																0x130C0084															
Bit	31	30	29	28	27	26	25	24	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

7.4.27 EPDC Working Buffer0 Base address Registers

	EPDC_WORK0BF																0x130C0090															
Bit	31	30	29	28	27	26	25	24	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

7.4.28 EPDC Working Buffer0 Size Registers

	EPDC_WORK0SIZE																0x130C0094															
Bit	31	30	29	28	27	26	25	24	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

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.4.29 EPDC Working Buffer1 Base address Registers

	EPDC_WORK1BF																0x130C0098															
Bit	31	30	29	28	27	26	25	24	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

7.4.30 EPDC Working Buffer0 Size Registers

	EPDC_WORK0SIZE																0x130C009C															
Bit	31	30	29	28	27	26	25	24	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

7.4.31 EPDC VCOM/Border Registers 0~15

	EPDC_VCOM0~ EPDC_VCOM15																0x130C0100~ 0x130C0140															
Bit	31	30	29	28	27	26	25	24	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/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

Bits	Name	Description	RW
31:0	VCOM/BD	The VCOM or border[1:0] of each frame up to 256-step EPDC_VCOMBD0: 16~1 EPDC_VCOMBD1: 32~17	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		EPDC_VCOMBD2: 48~33 EPDC_VCOMBD3: 64~49 EPDC_VCOMBD4: 80~65 EPDC_VCOMBD5: 96~81 EPDC_VCOMBD6: 112~97 EPDC_VCOMBD7: 128~113 EPDC_VCOMBD8: 144~129 EPDC_VCOMBD9: 160~145 EPDC_VCOMBD10: 176~161 EPDC_VCOMBD11: 192~177 EPDC_VCOMBD12: 212~193 EPDC_VCOMBD13: 228~213 EPDC_VCOMBD14: 240~229 EPDC_VCOMBD15: 256~241	

7.4.32 EPDC PRIORITY Registers

	EPDC_PRI																															0x130C0410
Bit	31	30	29	28	27	26	25	24	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
18:10	pri_th1	Threshold1: 0~511.	RW
8:0	pri_th0	Threshold0: 0~511	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7.5 Application Guide

7.5.1 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

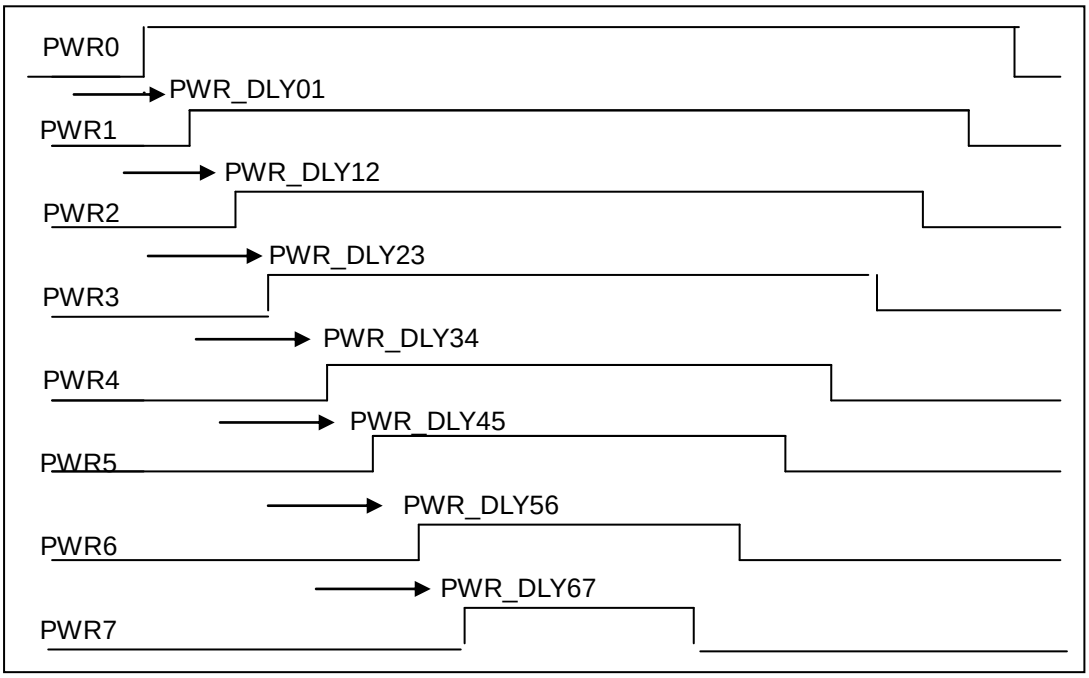
31:24	23:16	15:8	7:0
p3_pixcnt	p2_pixcnt	p1_pixcnt	p0_pixcnt

7.5.2 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Powers On/Off Sequence

7.5.3 Display Timing Setting

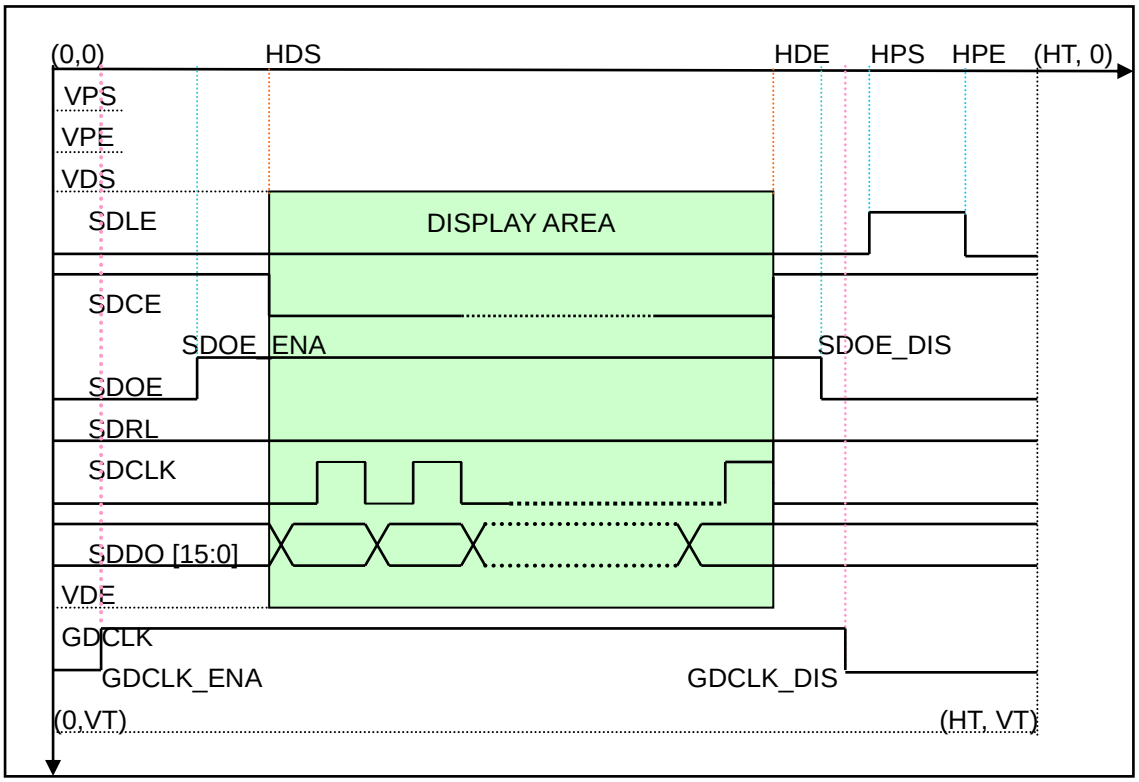
The controller is designed as part of LCD controller. The frequency of pixel clock of the LCD is always double of the Source Driver clock. Typically, the frame frequency is 50 Hz in EPD display. Calculate timing parameters as followings,

Frame Rate = Line Number * Line Period, and Line number is set as VT.

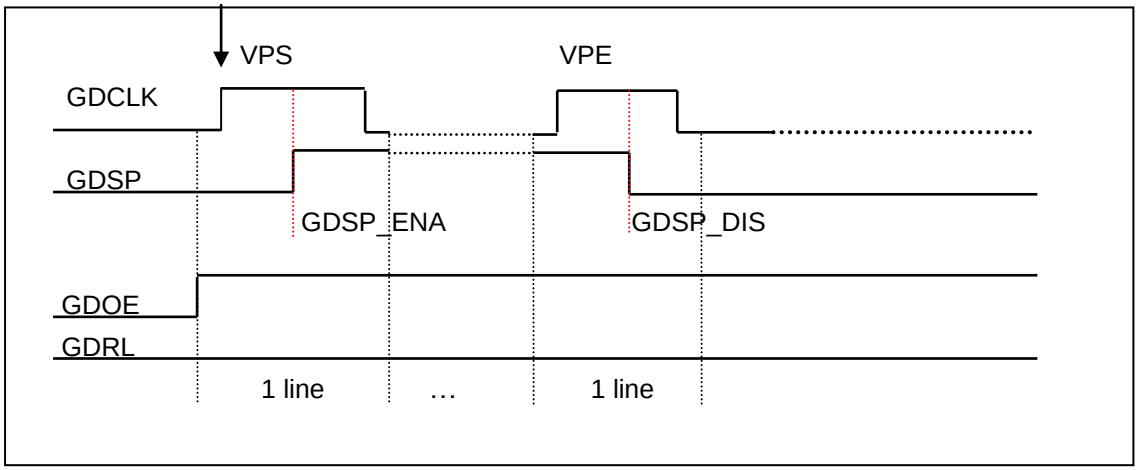
Line Period = HT * (SDCLK Clock Period), SDCLK Clock Period = 2 * Pixel Clock Period.

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

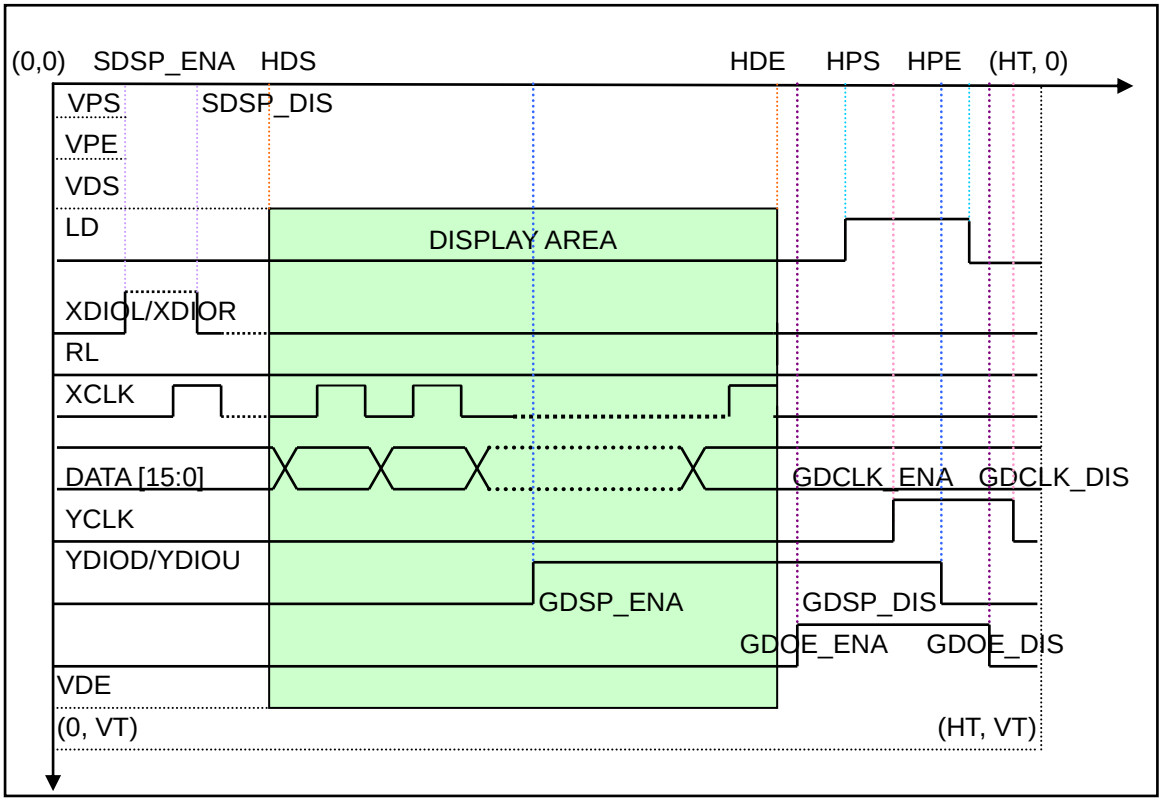


Source Drivers Reference Timing for TYPE1 EPD

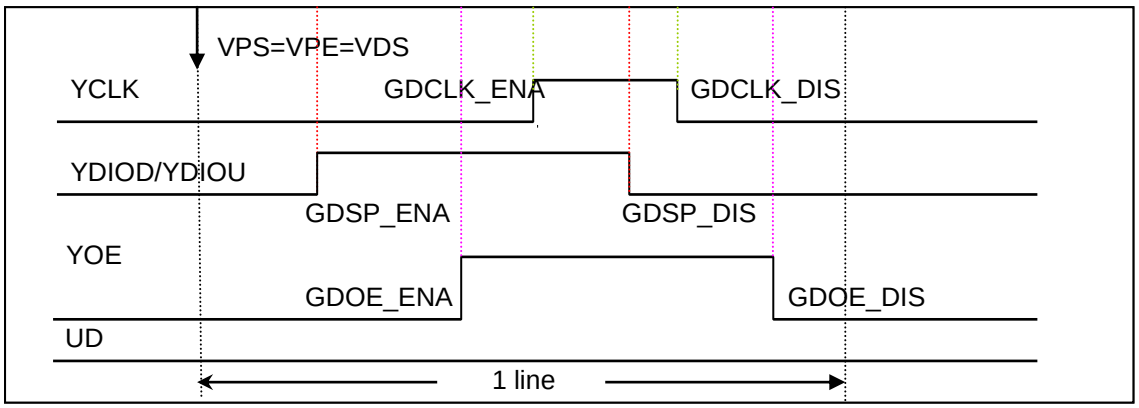


Gate Drivers Reference Timing for TYPE1 EPD

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Source Drivers Reference Timing for TYPE2 EPD



Gate Drivers Reference Timing for TYPE2 EPD

7.5.4 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 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

7.5.5 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.

7.5.6 Update VCOM0~15

For TYPE2 panel, before LUT updating and frame refreshing, EPDC_VCOMBD0~15 should be filled , EPDC_PMGR4 [VCOMBD_STEP] should be configured, and enable EPDC_PMGR3 [UNIPOL].

7.5.7 Border Display

1 Disable EPDC_PMGR3 [UNIPOL] and enable EPDC_PMGR3 [BDR_ENA].
2 Fill EPDC_VCOMBD0~15,configure EPDC_PMGR4 [VCOMBD_STEP].
3 Wait EPDC_STA [BDR_IDLE].
4 Set EPDC_CTRL [BDR_STRT].
5 Wait EPDC_STA [BDR_DONE].

7.5.8 Update flow

Auto-stop mode is recommended.

AUTO-STOP:

1 Prepare current buffer, working buffer and lut.
2 Initialize working buffer to all zero.
3 Configure EPDC registers and set 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_CFG [IBPP].
9 Change EPDC_CURBF to update data.
10 EPDC will stop when all pixels done.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8 EPD Color Engine

8.1 Overview

The EPD Color Engine (EPDCE) is used to do color post processing. The main functions include Color Linearization (VEE), Color Correction (HUE), Color Saturation, 3x3 Color Filter, Dither and Color Filter Remapping.

8.2 Features

- Input data format is RGB565
- Maximum image direction is 4096x4096
- Includes CSC between RGB888 and YUV444
- CSC supports 601 or 709, Wide or Narrow mode
- Includes 3x3 Color Filter modules for RGB.R, RGB.B, RGB.B and YUV.Y.
- Includes Color Linearization(VEE) for YUV.Y using 256-grade LUT
- Supports Color Correction(HUE) for YUV.UV, and the coefficients are configurable
- Supports Color Saturation for YUV.UV, and the coefficients are configurable
- Supports Dither for RGB.R, RGB.B, RGB.B and YUV.Y. The output format is 2/3/4-bit configurable.
- Supports Color Remapping for RGB.R, RGB.B, RGB.B and YUV.Y. If for RGB, there are two methods can be selected between individual CFA component and pixel array. And, the output order is configurable.
- The EPDCE has a AXI master interface and a AHB slave interface.

8.3 Register Descriptions

8.3.1 Register Mapping

Name	RW	Reset Value	Address offset	Access Size
CTRLR	RW	0x0000_0000	0x0000	32-bit
IFSR	RW	0x0000_0000	0x0004	32-bit
IFOFR	RW	0x0000_0000	0x0008	32-bit
IFAR	R	0x????_????	0x000C	32-bit
OFSR	RW	0x0000_0000	0x0010	32-bit
OFCR	RW	0x0000_0000	0x0018	32-bit
OFA0R	R	0x????_????	0x001C	32-bit
ISR	R	0x0000_0000	0x0020	32-bit
IMR	RW	0xFFFF_FFFF	0x0024	32-bit
ICR	W	0x0000_0000	0x0028	32-bit

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

IIFIDR	R	0x0000_0000	0x002C	32-bit
IOFIDR	R	0x0000_0000	0x0030	32-bit
DMACR	RW	0x0000_0000	0x0034	32-bit
DMASR	R	0x0000_0000	0x0038	32-bit
IDMADESR	RW	0x????_????	0x003C	32-bit
ODMADESR	RW	0x????_????	0x0040	32-bit
IDMACMDR	R	0x????_????	0x0044	32-bit
ODMACMD0R	R	0x????_????	0x0048	32-bit
CFCR	RW	0x0000_0000	0x005C	32-bit
CFRCER0	RW	0x0000_0000	0x0060	32-bit
CFRCER1	RW	0x0000_0000	0x0064	32-bit
CFRCER2	RW	0x0000_0000	0x0068	32-bit
VEECR	RW	0x0000_0000	0x006C	32-bit
HUECR	RW	0x0000_0000	0x0070	32-bit
HUECER	RW	0x0000_0000	0x0074	32-bit
CSCR	RW	0x0000_0000	0x0078	32-bit
CSCER	RW	0x0000_0000	0x007C	32-bit
DTCR	RW	0x0000_0000	0x0080	32-bit
IFIDR	R	0x????_????	0x0084	32-bit
OFIDR	R	0x????_????	0x0088	32-bit
IFWSR	RW	0x0000_0000	0x0094	32-bit
XRIDR	RW	0x0000_0000	0x009C	32-bit
XWIDR	RW	0x0000_000F	0x00A0	32-bit
CSCCR	RW	0x0000_0000	0x00A4	32-bit
CRCR	RW	0x0000_0000	0x00A8	32-bit
CFGCER0	RW	0x0000_0000	0x00AC	32-bit
CFGCER1	RW	0x0000_0000	0x00B0	32-bit
CFGCER2	RW	0x0000_0000	0x00B4	32-bit
CFBCER0	RW	0x0000_0000	0x00B8	32-bit
CFBCER1	RW	0x0000_0000	0x00BC	32-bit
CFBCER2	RW	0x0000_0000	0x00C0	32-bit
OFA1R	R	0x????_????	0x00C4	32-bit
ODMACMD0R	R	0x????_????	0x00C8	32-bit
VEE_LUT0- VEE_LUT	--	0x????_????	0x1000- 0x10FF	8-bit (only write ,data[31:30] instead of addr[1:0])
CR_LUT0- CR_LUT4095	--	0x????_????	0x2000- 0x2FFF	8/16-bit (write only data[31:30] instead of addr[1:0])

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.2 EPDCE Control Register (CTRLR)

	CTRLR																													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																														EPDCE_RST	EPDCE_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
31:2	Reserved	Writing has no effect. Read as zero.	RW
1	EPDCE_RST	Write 0 to this field to disable reset the controller. Write 1 to this field to reset the controller.	RW
0	EPDCE_EN	0: Disable the controller. 1: Enable the controller.	RW

8.3.3 Input Frame Size Register (IFSR)

	IFSR																													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			IN_FRM_VS													Reserved			IN_FRM_HS												
RST	0	0	0	0	0	0	0	0	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.	RW
28:16	IN_FRM_VS	Vertical size of input frame. If this field is n, there are (n+1) pixels in vertical. For color filer, the number of vertical pixels must be equal to or greater than 3.	RW
15:13	Reserved	Writing has no effect. Read as zero.	RW
12:0	IN_FRM_HS	Horizontal size of input frame. If this field is n, there are (n+1) pixels in vertical. The n+1 must be multiple of 8.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.4 Input Frame Offset Register (IFOFR)

	IFOFR																														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			IN_FRM_VOF													Reserved			IN_FRM_HOF												
RST	0	0	0	0	0	0	0	0	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.	RW
28:16	IN_FRM_VOF	Vertical offset of input frame. If this field is n, there are n-pixel offset in vertical.	RW
15:13	Reserved	Writing has no effect. Read as zero.	RW
12:0	IN_FRM_HOF	Horizontal size of input frame. If this field is n, there are n-pixel offset in vertical.	RW

8.3.5 Input Frame Address Register (IFAR)

	IFAR																														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
	IFA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	IFA	Input frame address. This field will be updated automatically by input DMA controller.	R

8.3.6 Input Frame Window Size Register (IFWSR)

	IFWSR																														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
	WIN_V_EN	Reserved		IN_FRM_WIN_VS													WIN_H_EN	Reserved		IN_FRM_WIN_HS												
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	WIN_V_EN	0: All the lines will be needed. 1: Only partial lines will be needed.	RW
30:29	Reserved	Writing has no effect. Read as zero.	RW
28:16	IN_FRM_WIN_VS	Frame window vertical size of input frame. If this field is n, there are (n+1) pixels in vertical. For color filer, the number of vertical pixels must be equal to or greater than 3.	RW
15	WIN_H_EN	0: All the pixels in horizontal will be needed. 1: Only partial pixels in horizontal will be needed.	RW
14:13	Reserved	Writing has no effect. Read as zero.	RW
12:0	IN_FRM_WIN_HS	Frame window horizontal size of input frame. If this field is n, there are (n+1) pixels in horizontal. The n+1 must be multiple of 8.	RW

8.3.7 Output Frame Size Register (OFSR)

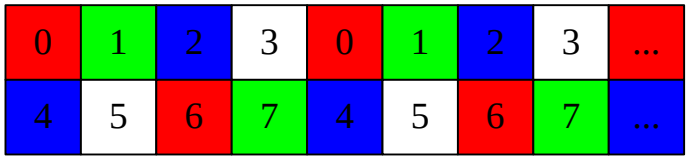
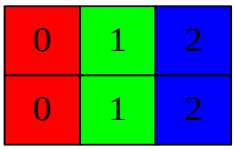
OFSR																														0x0010	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	O_WIN_HE	OUT_FRM_VS															Reserved	OUT_FRM_HS														
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31	O_WIN_HE	Enable one window to be selected for output frame. 0: Disable 1: Enable	RW
30:16	O_WIN_HS	The horizontal byte number of output window.	RW
15	Reserved	Writing has no effect. Read as zero.	RW
14:0	O_FRM_HS	The horizontal byte number of output frame.	RW

8.3.8 Output Frame Control Register (OFCR)

OFCR																														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
	SEL7		SEL6		SEL5		SEL4		SEL3		SEL2		SEL1		SEI0		Reserved															SEL
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:30	SEL7	This field will be used only when SEL is b00 or b10. When SEL is b00: 00: The corresponding data stands for R 01: The corresponding data stands for G 10: The corresponding data stands for B 11: The corresponding data stands for W  When SEL is b10:  	RW
29:28	SEL6		RW
27:26	SEL5		RW
25:24	SEL4		RW
23:22	SEl3		RW
21:20	SEl2		RW
19:18	SEL1		RW
17:16	SEL0		RW
15:2	Reserved	Writing has no effect. Read as zero.	RW
1:0	SEL	00: The output content includes R, G, B and W. 01: The output content includes Y only. 10: The output content includes R, G and B.	RW

8.3.9 Output Frame Address 0 Register (OFA0R)

	OFA0R																														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
	OFA0																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	OFA0	Output frame address 0. Under individual CFA component translation mode, this only indicates the output frame address of even lines; Otherwise, this field indicates all lines. This field will be updated automatically by output DMA controller.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.10 Output Frame Address 1 Register (OFA1R)

	OFA1R																													0x00C4
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFA1																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	OFA1	Output frame address 1. Under individual CFA component translation mode, this only indicates the output frame address of odd lines; Otherwise, this field indicates nothing. This field will be updated automatically by output DMA controller.	R

8.3.11 Interrupt Status Register (ISR)

	ISR																													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																										OSTOP	OEOF	OSOF	ISTOP	IEOF	ISOF
RST	0	0	0	0	0	0	0	0	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	Read as zero.	R
5	OSTOP	0: Common output DMA doesn't stop 1: Common output DMA stops	R
4	OEOF	0: No OEOF 1: End of frame interrupt generate by common output DMA	R
3	OSOF	0: No OSOF 1: Start of frame interrupt generated by common output DMA	R
2	ISTOP	0: Common input DMA doesn't stop 1: Common input DMA stops	R
1	IEOF	0: No IEOF 1: End of frame interrupt generate by common input DMA	R
0	ISOF	0: No ISOF 1: Start of frame interrupt generated by common input DMA	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.12 Interrupt Mask Register (IMR)

	IMR																												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
	Reserved																										OSTOP	OEOF	OSOF	ISTOP	IEOF	ISOF
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
31:6	Reserved	Writing has no effect. Read as 1.	RW
5	OSTOP	0: Enable the corresponding interrupt	RW
4	OEOF	1: Mask the corresponding interrupt	RW
3	OSOF		RW
2	ISTOP		RW
1	IEOF		RW
0	ISOF		RW

8.3.13 Interrupt Clear Register (ICR)

	ICR																												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
	Reserved																										OSTOP	OEOF	OSOF	ISTOP	IEOF	ISOF
RST	0	0	0	0	0	0	0	0	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.	R
5	OSTOP	Write 1 to the corresponding interrupt.	R
4	OEOF		R
3	OSOF		R
2	ISTOP		R
1	IEOF		R
0	ISOF		R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.14 Interrupt Input Frame ID Register (IIFIDR)

	IIFIDR																														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
	I_IN_ID																															
RST	0	0	0	0	0	0	0	0	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	I_IN_ID	Interrupt input frame identifier. This field will be updated when input frame interrupt occurs.	RW

8.3.15 Interrupt Output Frame ID Register (IOFIDR)

	IOFIDR																														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
	I_OUT_ID																															
RST	0	0	0	0	0	0	0	0	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	I_OUT_ID	Interrupt output frame identifier. This field will be updated when output frame interrupt occurs.	RW

8.3.16 DMA Control Register (DMACR)

	DMACR																														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																															CDMA_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
31:1	Reserved	Writing has no effect. Read as zero.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	CDMA_EN	0: Disable common DMA 1: Enable common DMA	RW

8.3.17 DMA Status Register (DMASR)

	DMASR																														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
	Reserved																										OSTOP	OSTOP	OSTOP	OSTOP	OSTOP	OSTOP
RST	0	0	0	0	0	0	0	0	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	Read as zero.	R
5	OSTOP	0: Common output DMA doesn't stop 1: Common output DMA stops	R
4	OEOF	0: No OEOF 1: Common output DMA has finished the transfer of frame	R
3	OSOF	0: No OSOF 1: Common output DMA has started the transfer of frame	R
2	ISTOP	0: Common input DMA doesn't stop 1: Common input DMA stops	R
1	IEOF	0: No IEOF 1: Common input DMA has finished the transfer of frame	R
0	ISOF	0: No ISOF 1: Common input DMA has started the transfer of frame	R

8.3.18 Input DMA Descriptor Address Register (IDMADESR)

	IDMADESR																														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
	IN_NDA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	IN_NDA	Next descriptor physical address for common input DMA in external memory. DMA Controller gets the next descriptor according to it after	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		finishing the current one. The target address must be aligned double-word boundary.	

8.3.19 Output DMA Descriptor Address Register (ODMADESR)

	ODMADESR																														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
	OUT_NDA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	OUT_NDA	Next descriptor physical address for common output DMA in external memory. DMA Controller gets the next descriptor according to it after finishing the current one. The target address must be aligned double-word boundary.	RW

8.3.20 Input DMA Command Register (IDMACMDR)

	IDMACMDR																														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
	SOF_EN	EOF_EN	STOP_EN	Reserved													LEN															
RST	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31	SOF_EN	Interrupt enable for Common DMA starting a frame-buffer transfer. 1: Enable DMA to generate input SOF interrupt or status.	R
30	EOF_EN	Interrupt enable for Common DMA ending a frame-buffer transfer. 1: Enable DMA to generate input EOF interrupt or status.	R
29	STOP_EN	DMA stop. When Common DMA complete transferring data, STOP bit decides whether DMA should loading next descriptor or not. 1: Enable DMA to generate input STOP interrupt or status.	R
28:26	Reserved	Writing has no effect, read as zero.	R
25:0	LEN	Length of transfer in bytes. Indicate the number of bytes to be transferred by common DMA.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		LEN = 0 is not valid. DMA transfers data according to LEN. Each time one or more byte(s) been transferred, LEN is decreased automatically.	

8.3.21 Output DMA Command 0 Register (ODMACMD0R)

	ODMACMD0R																														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
	SOF_EN	EOF_EN	STOP_EN	Reserved			LEN0																									
RST	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31	SOF_EN	Interrupt enable for common DMA starting a frame-buffer transfer. 1: Enable DMA to generate output SOF interrupt or status.	R
30	EOF_EN	Interrupt enable for common DMA ending a frame-buffer transfer. 1: Enable DMA to generate output EOF interrupt or status.	R
29	STOP_EN	DMA stop. When common DMA complete transferring data, STOP bit decides whether DMA should loading next descriptor or not. 1: Enable DMA to generate output STOP interrupt or status.	R
28:26	Reserved	Writing has no effect, read as zero.	R
25:0	LEN0	Length of transfer in bytes. Under individual CFA component translation mode, this only indicates the number of bytes to be transferred of even lines; Otherwise, this field indicates all lines. LEN0 = 0 is not valid. Each time one or more byte(s) been transferred, LEN0 is decreased automatically.	R

8.3.22 Output DMA Command 1 Register (ODMACMD1R)

	ODMACMD1R																														0x00C8
Bit	31	30	29	28	27	26	25	24	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						LEN1																									
RST	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:26	Reserved	Writing has no effect, read as zero.	R
25:0	LEN0	Length of transfer in bytes. Under individual CFA component translation mode, this only indicates the number of bytes to be transferred of odd lines; Otherwise, this field indicates nothing. Each time one or more byte(s) been transferred, LEN0 is decreased automatically.	R

8.3.23 Input Frame ID Register (IFIDR)

	IFIDR																														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
	FID																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	FID	Input frame identifier. This field will be updated when a descriptor is read in.	R

8.3.24 Output Frame ID Register (OFIDR)

	OFIDR																														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
	FID																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	I_OUT_ID	Output frame identifier. This field will be updated when output frame interrupt occurs.	R

8.3.25 CSC Control Register (CSCCR)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	CSCCR																													0x00A4
Bit	31	30	29	28	27	26	25	24	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																								Y2R_MODE		Reserved	Y2R_EN	R2Y_MODE		Reserved	R2Y_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
31:8	Reserved	Writing has no effect. Read as zero.	RW
7:6	Y2R_MODE	CSC mode selection. 00: 601 format wide conversion 01: 601 format narrow conversion 10: 709 format wide conversion 11: 709 format narrow conversion	RW
5	Reserved	Writing has no effect. Read as zero.	RW
4	Y2R_EN	Enable or disable CSC from YUV to RGB. 0: Disable CSC from YUV to RGB 1: Enable CSC from YUV to RGB	RW
3:2	R2Y_MODE	CSC mode selection. 00: 601 format wide conversion 01: 601 format narrow conversion 10: 709 format wide conversion 11: 709 format narrow conversion	RW
1	Reserved	Writing has no effect. Read as zero.	RW
0	R2Y_EN	Enable or disable CSC from RGB to YUV. 0: Disable CSC from RGB to YUV 1: Enable CSC from RGB to YUV	RW

8.3.26 3x3 Color Filter Control Register (CFCR)

	CSCCR																													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
	Reserved																														YCF_EN	RGBCF_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
31:2	Reserved	Writing has no effect. Read as zero.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	YCF_EN	0: Disable 3x3 Color Filter for YUV.Y 1: Enable 3x3 Color Filter for YUV.Y	RW
0	RGBCF_EN	0: Disable 3x3 Color Filter for RGB 1: Enable 3x3 Color Filter for RGB	RW

8.3.27 3x3 Color Filter Red Coefficients Register (CFRCER0)

	CFRCER0																													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
	K3								K2								K1								K0							
RST	0	0	0	0	0	0	0	0	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	K3	Coefficient K3 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
23:16	K2	Coefficient K2 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
15:8	K1	Coefficient K1 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
7:0	K0	Coefficient K0 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW

8.3.28 3x3 Color Filter Red Coefficients Register (CFRCER1)

	CFRCER1																													0x0064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K7								K6								K5								K4							
RST	0	0	0	0	0	0	0	0	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	K7	Coefficient K7 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
23:16	K6	Coefficient K6 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
15:8	K5	Coefficient K5 for RGB.R and YUV.Y 3x3 Color Filter.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		It should be in 2's complement.	
7:0	K4	Coefficient K4 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW

8.3.29 3x3 Color Filter Red Coefficients Register (CFRCER2)

	CFRCER2																														0x0068	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCALE								OFFSET								Reserved								R8							
RST	0	0	0	0	0	0	0	0	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	SCALE	Scaling factor for RGB.R and YUV.Y 3x3 Color Filter.	RW
23:16	OFFSET	Offset K8 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW
15:8	Reserved	Writing has no effect. Read as zero.	RW
7:0	K8	Coefficient K8 for RGB.R and YUV.Y 3x3 Color Filter. It should be in 2's complement.	RW

8.3.30 3x3 Color Filter Green Coefficients Register (CFGCER0)

	CFGCER0																														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
	K3								K2								K1								K0							
RST	0	0	0	0	0	0	0	0	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	K3	Coefficient K3 for RGB.G Color Filter. It should be in 2's complement.	RW
23:16	K2	Coefficient K2 for RGB.G Color Filter. It should be in 2's complement.	RW
15:8	K1	Coefficient K1 for RGB.G Color Filter. It should be in 2's complement.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

7:0	K0	Coefficient K0 for RGB.G Color Filter. It should be in 2's complement.	RW

8.3.31 3x3 Color Filter Green Coefficients Register (CFGCER1)

	CFGCER1																													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
	K7								K6								K5								K4							
RST	0	0	0	0	0	0	0	0	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	K7	Coefficient K7 for RGB.G Color Filter. It should be in 2's complement.	RW
23:16	K6	Coefficient K6 for RGB.G Color Filter. It should be in 2's complement.	RW
15:8	K5	Coefficient K5 for RGB.G Color Filter. It should be in 2's complement.	RW
7:0	K4	Coefficient K4 for RGB.G Color Filter. It should be in 2's complement.	RW

8.3.32 3x3 Color Filter Green Coefficients Register (CFGCER2)

	CFGCER2																													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
	SCALE								OFFSET								Reserved								R8							
RST	0	0	0	0	0	0	0	0	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	SCALE	Scaling factor for RGB.G Color Filter.	RW
23:16	OFFSET	Offset K8 for RGB.G Color Filter. It should be in 2's complement.	RW
15:8	Reserved	Writing has no effect. Read as zero.	RW
7:0	K8	Coefficient K8 for RGB.G Color Filter. It should be in 2's complement.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.33 3x3 Color Filter Blue Coefficients Register (CFBCER0)

	CFBCER0																													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
	K3								K2								K1								K0							
RST	0	0	0	0	0	0	0	0	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	K3	Coefficient K3 for RGB.B Color Filter. It should be in 2's complement.	RW
23:16	K2	Coefficient K2 for RGB.B Color Filter. It should be in 2's complement.	RW
15:8	K1	Coefficient K1 for RGB.B Color Filter. It should be in 2's complement.	RW
7:0	K0	Coefficient K0 for RGB.B Color Filter. It should be in 2's complement.	RW

8.3.34 3x3 Color Filter Blue Coefficients Register (CFBCER1)

	CFBCER1																													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
	K7								K6								K5								K4							
RST	0	0	0	0	0	0	0	0	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	K7	Coefficient K7 for RGB.B Color Filter. It should be in 2's complement.	RW
23:16	K6	Coefficient K6 for RGB.B Color Filter. It should be in 2's complement.	RW
15:8	K5	Coefficient K5 for RGB.B Color Filter. It should be in 2's complement.	RW
7:0	K4	Coefficient K4 for RGB.B Color Filter. It should be in 2's complement.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.3.35 3x3 Color Filter Blue Coefficients Register (CFBCER2)

	CFBCER2																													0x00C0	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCALE								OFFSET								Reserved								R8							
RST	0	0	0	0	0	0	0	0	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	SCALE	Scaling factor for RGB.B Color Filter.	RW
23:16	OFFSET	Offset K8 for RGB.B Color Filter. It should be in 2's complement.	RW
15:8	Reserved	Writing has no effect. Read as zero.	RW
7:0	K8	Coefficient K8 for RGB.B Color Filter. It should be in 2's complement.	RW

8.3.36 Color Linearization Control Register (VEECR)

	VEECR																													0x006C	
Bit	31	30	29	28	27	26	25	24	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																															VEE_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
31:1	Reserved	Writing has no effect. Read as zero.	RW
0	VEE_EN	0: Disable Color Linearization function 1: Enable Color Linearization function	RW

8.3.37 Color Correction Control Register (HUECR)

	HUECR																													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																															HUE_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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:1	Reserved	Writing has no effect. Read as zero.	RW
0	HUE_EN	0: Disable Color Correction function 1: Enable Color Correction function	RW

8.3.38 Color Correction Coefficients Register (HUECER)

	HUECER																														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						HUE_COS										Reserved						HUE_SIN									
RST	0	0	0	0	0	0	0	0	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.	RW
25:16	HUE_COS	Cosθ for Color Correction function. The range of Cosθ is [-1, 1], Here the correspond range will be [-256, 256]. It should be in 2's complement.	RW
15:10	Reserved	Writing has no effect. Read as zero.	RW
9:0	HUE_SIN	Sinθ for Color Correction function The range of Cosθ is [-1, 1], Here the correspond range will be [-256, 256]. It should be in 2's complement.	RW

8.3.39 Color Saturation Control Register (CSCR)

	CSCR																														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																															CS_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
31:1	Reserved	Writing has no effect. Read as zero.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	CS_EN	0: Disable Color Saturation function 1: Enable Color Saturation function	RW

8.3.40 Color Saturation Coefficients Register (CSCER)

	CSCER																														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
	Reserved																							CS_K								
RST	0	0	0	0	0	0	0	0	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.	RW
8:0	CS_K	Coefficient for Color Saturation function. The range is [0, 511].	RW

8.3.41 Dither Control Register (DTCR)

	DTCR																														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
	Reserved																										DITHER_SEL		Reserved	DITHERB_EN	DITHERG_EN	DITHERR_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
31:6	Reserved	Writing has no effect. Read as zero.	RW
5:4	DITHER_SEL	00: 2-bit Dither 01: 3-bit Dither 10: 4-bit Dither 11: Reserved If the Dither function is not wanted, this field must be configured for the next following functions.	RW
3	Reserved	Writing has no effect. Read as zero.	RW
2	DITHERB_EN	0: Disable Color Saturation function for RGB.B	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Enable Color Saturation function for RGB.B	
1	DITHERG_EN	0: Disable Color Saturation function for RGB.G 1: Enable Color Saturation function for RGB.G	RW
0	DITHERR_EN	0: Disable Color Saturation function for RGB.R or YUV.Y 1: Enable Color Saturation function for RGB.R or YUV.Y	RW

8.3.42 Color Remap Control Register (CRCR)

	CRCR																												0x00A8	
Bit	31	30	29	28	27	26	25	24	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																														CR_MODE	CR_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
31:2	Reserved	Writing has no effect. Read as zero.	RW
1	CR_MODE	0: Individual CFA Component translation mode 1: Pixel Array mode	RW
0	CR_EN	0: Disable color remap function 1: Enable color remap function	RW

8.3.43 AXI Read Identifier Register (XRIDR)

	XIDSR																												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
	Reserved																												ID_R			
RST	0	0	0	0	0	0	0	0	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.	RW
3:0	ID_R	AXI ID used for read operation.	RW

8.3.44 AXI Write Identifier Register (XWIDR)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	XIDPR																													0x00A0
Bit	31	30	29	28	27	26	25	24	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																												ID_W			
RST	0	0	0	0	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

Bits	Name	Description	RW
31:4	Reserved	Writing has no effect. Read as zero.	RW
3:0	ID_W	AXI ID used for writing operation.	RW

8.4 CSC from RGB888 to YUV444

There are four modes supported for the Color Space Conversion (CSC) from RGB888 toYUV444.

8.4.1 601 Format Wide Conversion

The precision is 8-bit.

The R, G and B have a nominal range of 16-235.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The functions are as following:

Y= 0.299R+0.587G+0.114B

Cb= -0.172R-0.339G+0.511B+128

Cr= 0.511R-0.428G-0.083B+128

8.4.2 601 Format Narrow Conversion

The precision is 8-bit.

The R, G and B have a nominal range of 0-255.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The functions are as following:

Y= 0.257R+0.504G+0.098B+16

Cb= -0.148R-0.291G+0.439B+128

Cr= 0.439R-0.368G-0.071B+128

8.4.3 709 Format Wide Conversion

The precision is 8-bit.

The R, G and B have a nominal range of 16-235.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The functions are as following:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Y= 0.213R+0.715G+0.072B

Cb= -0.117R-0.394G+0.511B+128

Cr= 0.511R-0.464G-0.047B+128

8.4.4 709 Format Narrow Conversion

The precision is 8-bit.

The R, G and B have a nominal range of 0-255.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The functions are as following:

Y= 0.183R+0.614G+0.062B+16

Cb= -0.101R-0.338G+0.439B+128

Cr= 0.439R-0.399G-0.040B+128

8.5 CSC from YUV444 to RGB888

There are four modes supported for the Color Space Conversion (CSC) from YUV444 to RGB888.

8.5.1 601 Format Wide Conversion

The precision is 8-bit.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The R, G and B have a nominal range of 16-235.

The functions are as following:

R= Y+1.371(Cr-128)

G= Y-0.689(Cr-128)-0.336(Cb-128)

B= Y+1.732(Cb-128)

8.5.2 601 Format Narrow Conversion

The precision is 8-bit.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The R, G and B have a nominal range of 0-255.

The functions are as following:

R= 1.164(Y-16)+1.596(Cr-128)

G= 1.164(Y-16)-0.813(Cr-128)-0.391(Cb-128)

B= 1.164(Y-16)+2.018(Cb-128)

8.5.3 709 Format Wide Conversion

The precision is 8-bit.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The R, G and B have a nominal range of 16-235.

The functions are as following:

R= Y+1.540(Cr-128)

G= Y-0.459(Cr-128)-0.183(Cb-128)

B= Y+1.816(Cb-128)

8.5.4 709 Format Narrow Conversion

The precision is 8-bit.

The Y has a nominal range of 16-235.

The Cb and Cr have a nominal range of 16-240.

The R, G and B have a nominal range of 0-255.

The functions are as following:

R= 1.164(Y-16)+1.793(Cr-128)

G= 1.164(Y-16)-0.534(Cr-128)-0.213(Cb-128)

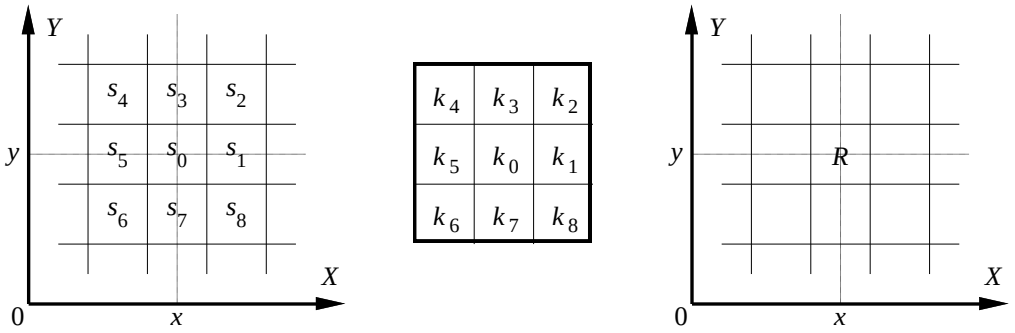
B= 1.164(Y-16)+2.115(Cb-128)

8.6 3x3 Color Filter

The precision is 8-bit.

The function can be used for RGB.R, RGB.B, RGB.B and YUV.Y.

The arithmetic is described as following:



(a) (b) (c)

Here:

S0-8 are source data

K0-8 are coefficients.

The result is:

$$R = k_0 s_0 + k_1 s_1 + \cdots + k_8 s_8$$

8.7 Color Linearization(VEE)

The precision is 8-bit.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The function is used for YUV.Y.

There is a 256 Bytes LUT to help to do VEE. And The LUT is programmable.

8.8 Color Correction (HUE)

The Color Correction function is 8-bit precision, and can be enabled or disabled.

This function is used for YUV.UV, and the arithmetic is

$$U' = U*\cos\theta + V*\sin\theta$$

$$V' = V*\cos\theta - U*\sin\theta$$

Here, sinθ and cosθ are configurable.

If the function is disabled, then

$$U' = U$$

$$V' = V$$

8.9 Color Saturation

The Color Saturation function is 8-bit precision, and can be enabled or disabled.

This function is used for YUV.UV, and the arithmetic is

$$U'=kU$$

$$V'=kV$$

Here, k is configurable.

If the function is disabled, then

$$U'=U$$

$$V'=V$$

8.10 Dither

There are three Dither controllers in the chip. One is for RGB.R and YUV.Y, one is for RGB.G and the last is for RGB.B.

The output for Dither is 8-bit precision, while the output of it can be 2/3/4 bits.

The controllers use a constant LUT as same as each other, which is stored internally and can't be programmed.

If the controller is disabled, then the output will use the most significant 2/3/4-bit of output data.

8.10.1 YUV.Y Dither

If YUV.Y Dither is needed, the following functions must be disabled: CSC from YUV to RGB, 3x3 Color Filter for RGB.R and the Color Remap.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8.11 Color Remap

The Color Remap function is used in RGB color space.

The output precision can be 2/3/4-bit.

8.11.1 Translate Mode

The translate-mode can be configured to Individual CFA Component or Pixel Array mode.

The following figure illustrates the Individual CFA Component translation mode.

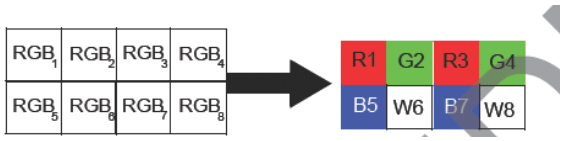


Figure 8-1 Individual CFA Component Translation Mode

The following figure illustrates the Pixel Array translation mode.

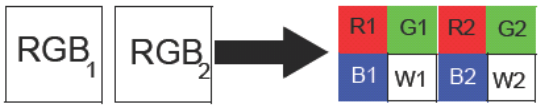
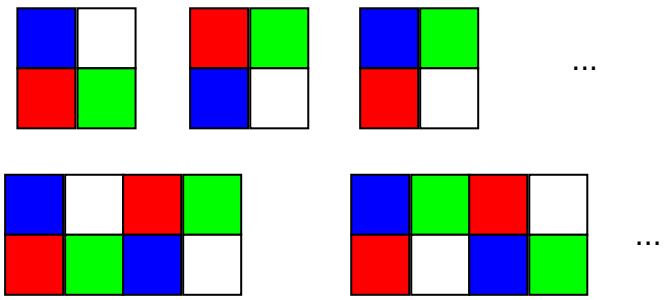


Figure 8-2 Pixel Array Translation Mode

8.11.2 Output Format of Color Remap

Programmable array format can be 2x2 or 2x4 and the output data can be any order for R, G, B and W.



The output data storage format:

For 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

For 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

31:30	29:28	27:26	25:24	23:22	21:20	19:18	17:16
p16	p15	p14	p13	p12	p11	p10	p9

8.11.3 LUT of Color Remap

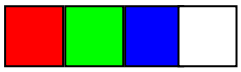
Color Remap controller uses a LUT of RGB to generate R'G'B'W. Because the output precision can be 2/3/4-bit, so three types of LUT will be used.

For 2-bit precision, a 64x8 bit LUT will be used.

For 3-bit precision, a 512x12 bit LUT will be used.

For 4-bit precision, a 4096x16 bit LUT will be used.

Whatever type LUT will be used, the LUT entry order should always be R-G-B-W, which is illustrated as following:



8.12 Common Input DMA

Common input DMA can work in descriptor mode only.

8.12.1 Descriptor of Common Input DMA

Word	Bit	Name	Description
0	31:0	NDA	Next descriptor physical address for common input DMA in external memory. DMA Controller gets the next descriptor according to it after finishing the current one. The target address It must be aligned double-word boundary. This value will be copied to IDMADESR after the descriptor is read into the controller. The address must be double-word aligned.
1	31:0	FA	Input frame buffer physical address. This value will be copied to IFAR after the descriptor is read into the controller. The address must be double-word aligned.
2	31	SOF_EN	Interrupt enable for DMA starting a frame-buffer transfer. 1: Enable DMA to generate input SOF interrupt or status. This value will be copied to IDMACMDR.SOF_EN after the descriptor is read into the controller.
	30	EOF_EN	Interrupt enable for DMA ending a frame-buffer transfer. 1: Enable DMA to generate input EOF interrupt or status. This value will be copied to IDMACMDR.EOF_EN after the descriptor is read into the controller.
	29	STOP_EN	DMA stop. When DMA complete transferring data, STOP bit decides whether DMA should loading next descriptor or not.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			1: Enable DMA to generate input STOP interrupt or status. This value will be copied to IDMACMDR.STOP_EN after the descriptor is read into the controller.
	28:26	Reserved	Writing has no effect, read as zero.
	25:0	LEN	Length of transfer in bytes. Indicate the number of bytes to be transferred by DMA. LEN = 0 is not valid. DMA transfers data according to LEN. Each time one or more byte(s) been transferred, LEN is decreased automatically. This value will be copied to IDMACMDR.LEN after the descriptor is read into the controller.
3	31:0	ID	Interrupt input frame identifier. This field will be updated when input frame interrupt occurs. This value will be copied to IFIDR after the descriptor is read into the controller.

8.12.2 Software Flow to use Common Input DMA without Interrupt

The flow for software to use common input DMA is described as following:

1. Allocates a 4-word descriptor space with start address DesAddr
2. Get frame buffer start address FrmAddr
3. Write FrmAddr to Descriptor.FA
4. Set Descriptor.EOF_EN to 1
5. Set Descriptor.STOP_EN to 1
6. Write the frame length to Descriptor.LEN
7. Write 0 to Descriptor.FID
8. Write DesAddr to IDMADESR
9. Enable Common Input DMA
10. Enable EPDCE controller
11. Polling for DMASR.IEOF to 1
12. Clear DMASR.IEOF
13. Disable Common Input DMA

Here, using one descriptor.

8.12.3 Software Flow to use Common Input DMA with Interrupt

The flow for software to use common input DMA is described as following:

1. Allocates a 4-word descriptor space with start address DesAddr
2. Get frame buffer start address FrmAddr
3. Write FrmAddr to Descriptor.FA
4. Set Descriptor.EOF_EN to 1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5. Set Descriptor.STOP_EN to 1
6. Write the frame length to Descriptor.LEN
7. Write 0 to Descriptor.FID
8. Write DesAddr to IDMADESR
9. Clear IMR.IEOF
10. Enable Common Input DMA
11. Enable EPDCE controller
12. Waiting for IEOF interrupt
13. Clear interrupt flag IEOF
14. Disable Common Input DMA

Here, using one descriptor

8.13 Common Output DMA

Common output DMA can work in descriptor mode only.

8.13.1 Descriptor of Common Output DMA

Word	Bit	Name	Description
0	31:0	NDA	Next descriptor physical address for common output DMA in external memory. DMA Controller gets the next descriptor according to it after finishing the current one. The target address It must be aligned double-word boundary. This value will be copied to ODMADESR after the descriptor is read into the controller. The address must be double-word aligned.
1	31:0	FA0	Input frame buffer physical address 0. Under individual CFA component translation mode, this only indicates the output frame address of even lines; Otherwise, this field indicates all lines. This value will be copied to OFA0R after the descriptor is read into the controller. The address must be double-word aligned.
2	31	SOF_EN	Interrupt enable for DMA starting a frame-buffer transfer. 1: Enable DMA to generate output SOF interrupt or status. This value will be copied to ODMACMDR.SOF_EN after the descriptor is read into the controller.
	30	EOF_EN	Interrupt enable for DMA ending a frame-buffer transfer. 1: Enable DMA to generate output EOF interrupt or status. This value will be copied to ODMACMD0R.EOF_EN after the descriptor is read into the controller.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	29	STOP_EN	DMA stop. When DMA complete transferring data, STOP bit decides whether DMA should loading next descriptor or not. 1: Enable DMA to generate output STOP interrupt or status. This value will be copied to ODMACMD0R.STOP_EN after the descriptor is read into the controller.
	28:26	Reserved	Writing has no effect, read as zero.
	25:0	LEN0	Length of transfer in bytes. Indicate the number of bytes to be transferred by DMA. Under individual CFA component translation mode, this only indicates the number of bytes to be transferred of even lines; Otherwise, this field indicates all lines. LEN0 = 0 is not valid. Each time one or more byte(s) been transferred, LEN0 is decreased automatically. This value will be copied to ODMACMD0R.LEN0 after the descriptor is read into the controller.
3	31:0	FID	Interrupt input frame identifier. This field will be updated when input frame interrupt occurs. This value will be copied to OFIDR after the descriptor is read into the controller.
4	31:0	FA1	Input frame buffer physical address . Under individual CFA component translation mode, this only indicates the output frame address of even lines; Otherwise, this field indicates nothing. This value will be copied to OFAR after the descriptor is read into the controller. The address must be double-word aligned.
5	31:26	Reserved	Writing has no effect, read as zero.
	25:0	LEN1	Length of transfer in bytes. Indicate the number of bytes to be transferred by DMA. Under individual CFA component translation mode, this only indicates the number of bytes to be transferred of even lines; Otherwise, this field indicates all lines Each time one or more byte(s) been transferred, LEN1 is decreased automatically. This value will be copied to ODMACMD1R.LEN1 after the descriptor is read into the controller.

8.13.2 Software Flow to use Common Input DMA without Interrupt

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Under individual CFA component translation mode, 6-word descriptor is needed; otherwise, 4-word descriptor is needed.

Here, a example of 4-word.

The flow for software to use common input DMA is described as following:

1. Allocates a 4-word descriptor space with start address DesAddr0
2. Allocates a frame buffer with start address FrmAddr0
3. Write FrmAddr0 to Descriptor0.FA0
4. Write the frame length LEN0 to Descriptor1.LEN0
5. Write 0 to Descriptor0.FID
6. Allocates a 4-word descriptor space with start address DesAddr1
7. Write DesAddr1 to Descriptor0.NDA
8. Allocates a frame buffer with start address FrmAddr1
9. Write FrmAddr1 to Descriptor1.FA0
10. Write the frame length LEN1 to Descriptor1.LEN0
11. Write 1 to Descriptor1.FID
12. Set Descriptor1.EOF_EN to 1
13. Set Descriptor1.STOP_EN to 1
14. Write DesAddr0 to IDMADESR
15. Enable Common Output DMA
16. Enable EPDCE controller
17. Polling for DMASR.OEOF to 1
18. Clear DMASR.OEOF
19. Disable Common Output DMA

Here, using two descriptors.

8.13.3 Software Flow to use Common Input DMA with Interrupt

The flow for software to use common input DMA is described as following:

1. Allocates a 4-word descriptor space with start address DesAddr0
2. Allocates a frame buffer with start address FrmAddr0
3. Write FrmAddr0 to Descriptor0.FA0
4. Write the frame length LEN0 to Descriptor1.LEN0
5. Write 0 to Descriptor0.FID
6. Allocates a 4-word descriptor space with start address DesAddr1
7. Write DesAddr1 to Descriptor0.NDA
8. Allocates a frame buffer with start address FrmAddr1
9. Write FrmAddr1 to Descriptor1.FA0
10. Write the frame length LEN1 to Descriptor1.LEN0
11. Write 1 to Descriptor1.FID

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12. Set Descriptor1.EOF_EN to 1
13. Set Descriptor1.STOP_EN to 1
14. Clear IMR.OEOF
15. Write DesAddr0 to ODMADESR
16. Enable Common Output DMA
17. Enable EPDCE controller
18. Waiting for OEOF interrupt
19. Clear interrupt flag OEOF
20. Disable Common Output DMA

Here, using two descriptors.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9 Camera Interface Module

9.1 Overview

The camera interface module (CIM) supports commonly available CMOS or CCD type image sensors. The CIM sources the digital image stream through a common 8-bit parallel digital protocol. The CIM can directly connect to external CMOS image sensors and ITU656 standard video decoders.

9.1.1 Features

- Input image size up to 2048x2048 pixels
- Max. VGA for image preview
- Max. VGA for video record
- Integrated DMA
- Supported data format: YCbCr 4:4:4, YCbCr 4:2:2 and other formats
- Output format: csc mode is YCbCr 4:2:2 or YCbCr 4:2:0, bypass mode is the input data format
- Output frame format
  - Packaged : for all data format
  - Separated: for YCbCr 4:4:4, YCbCr 4:2:2 and YCbCr 4:2:0
- Supports ITU656 (YCbCr 4:2:2) input
- Configurable CIM_VSYNC and CIM_HSYNC signals: active high/low
- Configurable CIM_PCLK: active edge rising/falling
- 256x33 image data receive FIFO (RXFIFO)
- PCLK max. 80MHz
- Configurable output order

9.1.2 Pin Description

Table 9-1 Camera Interface Pins Description

Name	I/O	Description
CIM_MCLK	O	CIM work clock
CIM_PCLK	I	Pixel clock from Image Sensor
CIM_VSYNC	I	Vertical synchronous from Image Sensor
CIM_HSYNC	I	Horizontal synchronous from Image Sensor
CIM_DATA[7:0]	I	Data bus from Image Sensor

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2 CIM Special Register

9.2.1 CIM Register Map

There are 2 controllers in chip. And the base address of CIM0 is 0x13060000, of CIM1 is 0x130E0000.

The special registers are for CIM to configure and control the interface and DMA operation. The table below lists these registers.

Table 9-2 CIM Registers

Name	RW	Reset Value	Offset	Access Size
CIMCFG	RW	0x00000000	0x0000	32
CIMCR	RW	0x00000000	0x0004	32
CIMST	RW	0x00150002	0x0008	32
CIMIID	R	0x00000000	0x000C	32
CIMDA	RW	0x00000000	0x0020	32
CIMFA	R	0x00000000	0x0024	32
CIMFID	R	0x00000000	0x0028	32
CIMCMD	R	0x00000000	0x002C	32
CIMWSIZE	RW	0x00000000	0x0030	32
CIMWOFFSET	RW	0x00000000	0x0034	32
CIMYFA	R	0x00000000	0x0038	32
CIMYCMD	R	0x00000000	0x003C	32
CIMCBFA	R	0x00000000	0x0040	32
CIMCBCMD	R	0x00000000	0x0044	32
CIMCRFA	R	0x00000000	0x0048	32
CIMCRCMD	R	0x00000000	0x004C	32
CIMCR2	RW	0x00000000	0x0050	32
CIMFS	RW	0x00000000	0x0054	32
CIMIMR	RW	0x00000F0D	0x0058	32
CIMTC	RW	0x00000000	0x005C	32
CIMTINX	RW	0x00000000	0x0060	32
CIMTCNT	R	0x????????	0x0064	32

9.2.2 CIM Configuration Register (CIMCFG)

	CIMCFG																														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
	EEOFEN	EXP	Reserved				BS		BW			SEP	ORDER		DF		INV_DAT	VSP	HSP	PCP	BURST_T	YPE	DUMMY	E_VSYNC	LM	PACK			FP	Reserved	DSM	
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31</td><td>EEOFEN</td><td>Early EOF Mode Enable.
0: EEOF mode is disabled
1: When CIMCR.EEOF_LINE lines data has been transferred of a frame, the EEOF flag will be set, and the EEOF interrupt will occur</td><td>RW</td></tr>
<tr><td>30</td><td>EXP</td><td>Expand mode for CIM_DATA width.
0: CIM_DATA width = 8
1: CIM_DATA width > 8</td><td></td></tr>
<tr><td>29:24</td><td>Reserved</td><td>Writing has no effect. Read as zero.</td><td>R</td></tr>
<tr><td>23:22</td><td>BW</td><td>Bus width of CIM_DATA Interface.
When BW is n, the bus width is (n+9) bits.</td><td>RW</td></tr>
<tr><td>20</td><td>SEP</td><td>Separate frame format enable.
Used in output data format of YCbCr 4:4:4 and YcbCr 4:2:2.
0: Output is packaged frame format
1: Output is separated frame format</td><td>RW</td></tr>
<tr><td>19:18</td><td>ORDER</td><td>Input data stream order.
<table>
<tr><th></th><th>YCbCr 4:4:4</th><th>ITU656/YCbCr 4:2:2</th></tr>
<tr><td>00</td><td>YCbCr</td><td>Y_0CbY_1Cr</td></tr>
<tr><td>01</td><td>YCrCb</td><td>Y_0CrY_1Cb</td></tr>
<tr><td>10</td><td>CbCrY</td><td>CbY_0CrY_1</td></tr>
<tr><td>11</td><td>CrCbY</td><td>CrY_0CbY_1</td></tr>
</table></td><td>RW</td></tr>
<tr><td>17:16</td><td>DF</td><td>Input data format.
00: The data format has no effect.
01: YCbCr 4:4:4
10: YCbCr 4:2:2
11: ITU656 YCbCr 4:2:2</td><td>RW</td></tr>
<tr><td>15</td><td>INV_DAT</td><td>Inverse every bit of input data.
0: not inverse; 1: inverse.</td><td>RW</td></tr>
<tr><td>14</td><td>VSP</td><td>VSYNC polarity selection. When VSYNC signal is input from pin. CIM_VSYNC, this bit specifies the VSYNC signal active level and leading edge. When VSYNC is retrieved from SAV&EAV, this bit is ignored.
0: VSYNC signal active high, VSYNC signal leading edge is rising edge
1: VSYNC signal active low, VSYNC signal leading edge is falling edge</td><td>RW</td></tr>
<tr><td>13</td><td>HSP</td><td>Specifies the HSYNC signal active level and leading edge.
0: HSYNC signal active high, HSYNC signal leading edge is rising edge
1: HSYNC signal active low, HSYNC signal leading edge is falling edge</td><td>RW</td></tr>
<tr><td>12</td><td>PCP</td><td>Specifies the PCLK working edge.
0: Data is sampled by PCLK rising edge
1: Data is sampled by PCLK falling edge</td><td>RW</td></tr>
<tr><td>11:10</td><td>BURST_
TYPE</td><td>DMA burst type.
00: INCR4
01: INCR8</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>10: INCR16
11: INCR32</td><td></td></tr>
<tr><td>9</td><td>DUMMY</td><td>DUMMY zero function. When DUMMY is 1, CIM hardware adds one byte zero to every 3 input data bytes to form 32-bit data.
0: DUMMY zero function disabled
1: DUMMY zero function enabled</td><td>RW</td></tr>
<tr><td>8</td><td>E_VSYNC</td><td>External / internal VSYNC selection. When DSM is ITU656Progressive Mode, VSYNC can be external (provided by sensor) or internal (retrieved from SAV&EAV). This bit only valid for ITU656Progressive Mode; In other DSM modes, this bit should always be 0.
0: Internal VSYNC mode, pin CIM_VSYNC is ignored
1: External VSYNC mode, VSYNC is provided by image sensor via pin CIM_VSYNC</td><td>RW</td></tr>
<tr><td>7</td><td>LM</td><td>Line Mode for ITU656.
0: EAV is before SAV in each line
1: SAV is before EAV in each line</td><td>RW</td></tr>
<tr><td>6:4</td><td>PACK</td><td>Data packing mode, pack 8-bit input data into 32-bit data for FIFO.
<table>
<tr><th>PACK</th><th>Bypass Mode</th><th>CSC Mode</th></tr>
<tr><td>3’b000</td><td>0x 11 22 33 44</td><td>0x Y_0 Cb Y_1 Cr</td></tr>
<tr><td>3’b001</td><td>0x 22 33 44 11</td><td>0x Cb Y_1 Cr Y_0</td></tr>
<tr><td>3’b010</td><td>0x 33 44 11 22</td><td>0x Y_1 Cr Y_0Cb</td></tr>
<tr><td>3’b011</td><td>0x 44 11 22 33</td><td>0x Cr Y_0 Cb Y_1</td></tr>
<tr><td>3’b100</td><td>0x 44 33 22 11</td><td>0x Cr Y_1 Cb Y_0</td></tr>
<tr><td>3’b101</td><td>0x 33 22 11 44</td><td>0x Y_1 Cb Y_0 Cr</td></tr>
<tr><td>3’b110</td><td>0x 22 11 44 33</td><td>0x Cb Y_0 Cr Y_1</td></tr>
<tr><td>3’b111</td><td>0x 11 44 33 22</td><td>0x Y_0 Cr Y_1 Cb</td></tr>
</table>
In this table, 0x11, 0x22, 0x33 and 0x44 mean the received data from the sensor, 0x11 is received first and 0x44 is received last, and Y0 is received before Y1.</td><td>6:4</td></tr>
<tr><td>3</td><td>FP</td><td>Field flag polarity selection. When ITU656 progressive stream is input, this bit specifies the field flag active level. When other modes are used, this bit is ignored.
0: Field flag active low
1: Field flag active high</td><td>RW</td></tr>
<tr><td>2</td><td>Reserved</td><td>Writing has no effect. Read as zero.</td><td>R</td></tr>
<tr><td>1:0</td><td>DSM</td><td>Data sample mode. Please refer to the table below.
<table>
<tr><th>DSM</th><th>Description</th></tr>
<tr><td>2’b00</td><td>ITU656Progressive Mode</td></tr>
<tr><td>2’b01</td><td>ITU656Interlace Mode</td></tr>
<tr><td>2’b10</td><td>Gated Clock Mode</td></tr>
<tr><td>2’b11</td><td>Reserved</td></tr>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2.3 CIM Control Register (CIMCR)

	CIMCR																													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
	EEOF_LINE												FRC				Reserved	WINE	Reserved						DMA_SYNC	MBEN	H_SYNC	Reserved	SW_RST	DMA_EN	RF_RST	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
31:20	EEOF_LINE	When EEOF_LINE lines data has been transferred of a frame, the EEOF flag will be set, and the EEOF interrupt will occur.	RW
19:16	FRC	CIM frame rate control. If FRC = n, CIM sampling one frame from every (n+1) frames from the sensor.	RW
15	Reserved	Read as zero.	R
14	WINE	To enable window-image. Used to indicate whether the registers CIMWSIZE and CIMWOFFSET work or not. 0: the value in CIMWSIZE and CIMWOFFSET will be ignored 1: the value in CIMWSIZE and CIMWOFFSET will be used	RW
13:8	Reserved	Read as zero.	R
7	DMA_SYNC	The control bit to enable DMA synchronization. 0: The valid data input to CIM will be transferred by DMA to external memory 1: When a new descriptor-DMA transfer starts with writing CIMDA, a frame synchronization will be done, and the data in RXFIFO will be ignored	RW
6	MBEN	Macro Block Mode Enable. It effects the output data storage only when the output data format is YUV420. 0: The output data of a frame is separated to Y, U and V. 1: The data storage is according to the 16x16 macro blocks.	RW
5	H_SYNC	Horizontal Sync Enable. 0: disable 1: enable It is only used when CIMCFG.SEP is 1.	RW
4	Reserved	Writing has no effect, read as zero.	R
3	SW_RST	Software reset enable. 0: Don't care 1: Reset the CIM module.	RW
2	DMA_EN	Enable / disable the DMA function. 0: disable DMA; 1: enable DMA.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1	RF_RST	RXFIFO software reset. Setting 1 to RXF_RST can reset RXFIFO immediately. Setting 0 to RXF_RST can stop resetting RXFIFO. After reset, RXFIFO is empty.	RW
0	ENA	Enable or disable the CIM module. 0: CIM is not enabled, or disable CIM immediately 1: CIM is enabled, or enabling CIM	RW

9.2.4 CIM Control Register 2 (CIMCR2)

	CIMCR2																													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						FRAGHE	FRAGVE	FSC	ARIF	HRS		VRS		CSC		Reserved										OP		Reserved	OPE	EMM	APM
RST	0	0	0	0	0	0	0	0	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	FRAGHE	When horizontal frame scale is enabled and the frame horizontal size is not the multiple of the down scale, this bit decides the remainder will be ignored or not. 0: not ignore 1: ignore	RW
24	FRAGVE	When vertical frame scale is enabled and the frame vertical size is not the multiple of the down scale, this bit decides the remainder will be ignored or not. 0: not ignore 1: ignore	RW
23	FSC	Enable frame size check. 0: Disable frame size check. 1: Enable frame size check.	RW
22	ARIF	Enable auto recovery for incomplete frame. This field will effect only when FSC is 1. 0: Disable auto recovery. 1: Enable auto recovery.	RW
21:20	HRS	Specified the horizontal ratio for down scale. 00: no scale 01: 1/2 down scale 10: 1/4 down scale 11: 1/8 down scale	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td>19:18</td><td>VRS</td><td>Specified the vertical ratio for down scale.
00: no scale
01: 1/2 down scale
10: 1/4 down scale
11: 1/8 down scale</td><td>RW</td></tr>
<tr><td>17:16</td><td>CSC</td><td>CSC Mode Select.
00: Bypass mode.
01: Reserved.
10: CSC to YCbCr 422.
11: CSC to YCbCr 420.</td><td>RW</td></tr>
<tr><td>15:6</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>5:4</td><td>OP</td><td>Optional Priority Configuration. Only used when OPE is set to 1.
<table><tr><th>PG</th><th>CIM AHB Priority</th><th>Number of Data in FIFO</th></tr><tr><td>2’b00</td><td>0
1
2
3</td><td>n <= 8
8 < n <= 16
16 < n <= 32
32 < n</td></tr><tr><td>2’b01</td><td>0
1
2
3</td><td>n <= 16
16 < n <= 32
32 < n <= 64
64 < n</td></tr><tr><td>2’b10</td><td>0
1
2
3</td><td>n <= 32
32 < n <= 64
64 < n <= 96
96 < n</td></tr><tr><td>2’b11</td><td>0
1
2
3</td><td>n <= 64
64 < n <= 96
96 < n <= 128
128 < n</td></tr></table>It is suggested to use 2’b10.</td><td>RW</td></tr>
<tr><td>3</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>2</td><td>OPE</td><td>Optional Priority Mode Enable Control. Only used when APM is 1.
0: CIM calculates the priority according to the fifo status
1: CIM calculates the priority according to OPG which is configured by software</td><td>RW</td></tr>
<tr><td>1</td><td>EME</td><td>Emergency Mode Enable Control.
0: Emergency Mode Disable
1: Emergency Mode Enable</td><td>RW</td></tr>
<tr><td>0</td><td>APM</td><td>Auto Priority Mode Enable Control.
0: Auto priority mode disable. CIM uses the priority set by arbiter</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Auto priority mode enable. CIM can use the priority according the fifo status	

9.2.5 CIM Status Register (CIMST)

	CIMST																													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										Cr_RFOF	Cr_RFE	Cb_RFOF	Cb_RFE	Y_RFOF	Y_RFE	Reserved				DEEOF	DSTP	DEOF	DSOF	Reserved			TLBE	FSE	RFOF	RFE	VDD
RST	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bits	Name	Description	RW
31:22	Reserved	Writing has no effect, read as zero.	R
21	Cr_RFOF	Cr_RXFIFO over flow. When Cr_RXFIFO over flow happens, Cr_RX_OF is set 1. Can generate interrupt if CIMIMR.RFOFM bit is set. Write 0 to this bit to clear.	RW
20	Cr_RFE	Cr_RXFIFO empty. Indicates whether Cr_RXFIFO is empty or not. After reset, RXFIFO is empty, and Cr_RX_EMPTY is 1. 0: Cr_RXFIFO is not empty 1: Cr_RXFIFO is empty	R
19	Cb_RFOF	Cb_RXFIFO over flow. When Cb_RXFIFO over flow happens, Cb_RX_OF is set 1. Can generate interrupt if CIMIMR.RFOFM bit is set. Write 0 to this bit to clear.	RW
18	Cb_RFE	Cb_RXFIFO empty. Indicates whether Cb_RXFIFO is empty or not. After reset, Cb_RXFIFO is empty, and Cb_RX_EMPTY is 1. 0: Cb_RXFIFO is not empty 1: Cb_RXFIFO is empty	R
17	Y_RFOF	Y_RXFIFO over flow. When Y_RXFIFO over flow happens, Y_RX_OF is set 1. Can generate interrupt if CIMIMR.RFOFM bit is set. Write 0 to this bit to clear.	RW
16	Y_RFE	Y_RXFIFO empty. Indicates whether Y_RXFIFO is empty or not. After reset, Y_RXFIFO is empty, and Y_RX_EMPTY is 1. 0: Y_RXFIFO is not empty 1: Y_RXFIFO is empty	R
15:12	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11	DEEOF	When set to 1, indicates the DMA has transferred CIMCTRL.EEOF_LINE lines data of a frame. Can generate an interrupt if CIMIMR.DEEOFM is 0. Write 0 to clear.	RW
10	DSTOP	When set to 1, indicates the DMA complete transferring data and stop the operation. Can generate an interrupt if CIMIMR.DSTPM is 0. Write 0 to clear.	RW
9	DEOF	When set to 1, indicates the DMA complete a transfer from RXFIFO to a frame buffer. Can generate an interrupt if CIMIMR. DEOFM is 0. Write 0 to clear.	RW
8	DSOF	When set to 1, Indicates the DMA start a transfer from RXFIFO to a frame buffer. Can generate an interrupt if CIMIMR. DSOFM is 0. Write 0 to clear.	RW
7:5	Reserved	Read as zeros.	R
4	TLBE	TLB error. When set to 1, Indicates there is a TLB error. Write 0 to clear.	RW
3	FSE	Indicates that frame size check error. 0: No frame size check error occurs. 1: The received size of frame is not equal to the expected. Can generate an interrupt if CIMIMR.FSEM is 0. Write 0 to clear.	RW
2	RFOF	Indicates RXFIFO overflow. 0: RXFIFO is not overflow. 1: RXFIFO is overflow. Can generate an interrupt if CIMIMR.ROFM is 0. Write 0 to clear.	RW
1	RFE	Indicates RXFIFO is empty. 0: RXFIFO is not empty 1: RXFIFO is empty	R
0	VDD	CIM disable done. Indicate this module is disabled after clear the CIMCR.ENA bit to disable the CIM module. 0: CIM has not been disabled. 1: CIM has been disabled. Can generate an interrupt if CIMIMR.VDDM is 0. Write 0 to this bit to clear.	RW

9.2.6 CIM Interrupt Mask Register (CIMIMR)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	CIMIMR																												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																				DEEOFM	DSTPM	DEOFM	DSOFM	Reserved			TLBEM	FSEM	RFOFM	Reserved	VDDM
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	1	1	1	0	1

Bits	Name	Description	RW
31:12	Reserved	Read as zeros.	R
11	DEEOFM	The control bit to mask DMA EEOF interrupt. 0: enable 1: mask	RW
10	DSTPM	The control bit to mask DMA STOP interrupt. 0: enable 1: mask	RW
9	DEOFM	The control bit to mask DMA EOF interrupt. 0: enable 1: mask	RW
8	DSOFM	The control bit to mask DMA SOF interrupt. 0: enable 1: mask	RW
7:5	Reserved	Read as zeros.	R
4	TLBEM	This bit to control TLB error interrupt. 0: enable 1: mask	RW
3	FSEM	The control bit to mask frame size check error interrupt. 0: enable 1: mask	RW
2	RFOFM	The control bit to mask RXFIFO overflow interrupt. 0: enable 1: mask	RW
1	Reserved	Read as zeros.	R
0	VDDM	The control bit to mask VDD interrupt. 0: enable 1: mask	RW

9.2.7 CIM Interrupt ID Register (CIMIID)

	CIMIID																												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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	FID
RST	0 0 0 0 0 0 0 0 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	FID	Interrupt frame ID. Contains a copy of the Frame ID register (CIMFID) from the descriptor currently being processed when a DMA_SOF or DMA_EOF interrupt is generated. CIMIID is written to only when CIMCMD.SOFINT or CIMCMD.EOFINT is high. As such, the register is considered to be sticky and will be overwritten only when the associated interrupt is cleared by writing the CIM state register.	R

9.2.8 CIM Descriptor Address (CIMDA)

	CIMIDA																														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
	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	Next descriptor physical address in external memory. DMAC gets the next descriptor according to it after finishing the current one. The target address Bits [3:0] must be zero to be aligned to 16-byte boundary.	RW

9.2.9 CIM Frame buffer Address Register (CIMFA)

	CIMIFA																														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
	FA																															
RST	0	0	0	0	0	0	0	0	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	FA	Frame buffer virtual address in external memory when CIMCFG. SEP is 0.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		When starts CIM, DMA transfers data from RXFIFO to frame buffer. This address is increased by hardware automatically. Bits [6:0] must be zero to be aligned to 32-word boundary.	

NOTE: CIMFA comes from DMA Descriptor, so here it is read-only.

9.2.10 CIM Frame ID Register (CIMFID)

	CIMFID																															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
	FID																															
RST	0	0	0	0	0	0	0	0	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	FID	Frame ID. The particular use of this field is up to the software. This ID will be copied to the CIMIID register when an interrupt occurs.	R

NOTE: CIMFID comes from DMA Descriptor, so here it is read-only.

9.2.11 CIM DMA Command Register (CIMCMD)

	CIMFID																															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
	SOFINT	EOFINT	EEOFINTEn	STOP	Reserved												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	SOFINTEn	Interrupt enable for DMA starting a frame-buffer transfer. 1: DMA will set CIMSTATE.DMA_SOF when start of a frame-buffer transfer When one frame uses several buffers, it is suggested to set SOFINTEn of first buffer only.	R
30	EOFINTEn	Interrupt enable for DMA ending a frame-buffer transfer. 1: DMA will set CIMSTATE.DMA_EOF when CIMCMD.LEN is decreased to 0, which means end of a frame-buffer transfer	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		When one frame uses several buffers, it is suggested to set EOFINTEn of last buffer only.	
29	EEOFINTEn	Interrupt enable for DMA issuing an earlier eof interrupt.	R
28	STOP	DMA stop. When DMA complete transferring data, STOP bit decides whether DMA should loading next descriptor or not. 0: DMA start loading next descriptor 1: DMA stopped, and CIMSTATE.DMA_STOP bit is set 1 by hardware	R
27	OFRCVEN	Auto recovery enable when there is RXFIFO overflow. 0: No auto recovery when overflow occurs, thus the software should do something 1: Auto recovery enable, the hardware will correct the overflow	R
26:24	Reserved	Writing has no effect, read as zero.	R
23:0	LEN	Length of transfer in words. Indicate the number of words to be transferred by DMA to a frame buffer. LEN = 0 is not valid. DMA transfers data according to LEN. Each time one or more word(s) been transferred, LEN is decreased automatically.	R

9.2.12 CIM Window Size (CIMWSIZE)

	CIMWSIZE																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
	Reserved			LPF													Reserved			PPL												
RST	0	0	0	0	0	0	0	0	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	LPF	Lines per frame for CIM output.	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	PPL	PPL must be multiples of 2. In fact, the number of CIM output data in word is equal to PPL/2.	RW

NOTE:

When CIMCFG.SEP is 1, the total pixel number of window-size must be multiple of 4 or 8.

- When output data format is YCBCR4:4:4, it must be multiple of 4.
- When output data format is YCBCR4:2:2, it must be multiple of 8.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2.13 CIM Window Offset (CIMWOFFSET)

	CIMWOFFSET																													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			V_OFFSET													Reserved			H_OFFSET												
RST	0	0	0	0	0	0	0	0	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	V_OFFSET	Vertical offset.	RW
15:13	Reserved	Writing has no effect, read as zero.	R
12:0	H_OFFSET	Horizontal offset. It should be an even number.	RW

9.2.14 CIM Frame Size Register (CIMFS)

This register will indicate how many pixels and bytes are included in a frame. The maximum frame size is 8192*8192.

	CIMFS																													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			FVS													BPP		Reserved	FHS												
RST	0	0	0	0	0	0	0	0	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	FVS	Vertical size of the frame N indicates the vertical includes (n+1) pixels.	RW
15:14	BPP	Number of bytes per pixel. 00: One pixel includes 1 byte. 01: One pixel includes 2 bytes. 10: One pixel includes 3 bytes. 11: One pixel includes 4 bytes.	RW
13	Reserved	Writing has no effect, read as zero.	R
12:0	FHS	Horizontal size of the frame. N indicates the horizontal includes (n+1) pixels.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2.15 CIM Y Frame buffer Address Register (CIMYFA)

	CIMIYFA																															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
	YFA																															
RST	0	0	0	0	0	0	0	0	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	YFA	Y Frame buffer virtual address in external memory when CIMCFG. SEP is 1. When starts CIM, DMA transfers data from Y_RXFIFO to frame buffer. This address is increased by hardware automatically. Bits [7:0] must be zero to be aligned to 32-word boundary when the data format is YUV420; Otherwise, bits [6:0] must be zero to be aligned to 32-word boundary.	R

9.2.16 CIM Y DMA Command Register (CIMYCMD)

	CIMYCMD																															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
	SOFINT	EOFINT	EEOFINTEn	STOP	OFRCVEN	Reserved			YLEN																							
RST	0	0	0	0	0	0	0	0	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	SOFINTEn	Interrupt enable for DMA starting a frame-buffer transfer. 1: DMA will set CIMSTATE.DMA_SOF when start of a frame-buffer transfer When one frame uses several buffers, it is suggested to set SOFINTEn of first buffer only.	R
30	EOFINTEn	Interrupt enable for DMA ending a frame-buffer transfer. 1: DMA will set CIMSTATE.DMA_EOF when CIMYCMD.YLEN and CIMCbCMD.CbLEN and CIMCrCMD.CrLEN are decreased to 0, which means end of a frame-buffer transfer When one frame uses several buffers, it is suggested to set EOFINTEn of last buffer only.	R
29	EEOFINTEn	Interrupt enable for DMA issuing an earlier eof interrupt.	R
28	STOP	DMA stop. When DMA complete transferring data, STOP bit decides	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		whether DMA should loading next descriptor or not. 0: DMA start loading next descriptor 1: DMA stopped, and CIMSTATE.DMA_STOP bit is set 1 by hardware	
27	OFRCVEN	Auto recovery enable when there is RXFIFO overflow. 0: No auto recovery when overflow occurs, thus the software should do something 1: Auto recovery enable, the hardware will correct the overflow DMA will do a frame synchronization, and retransfer the current descriptor.	
26:24	Reserved	Writing has no effect, read as zero.	R
23:0	YLEN	Length of transfer in words. Indicate the number of words to be transferred by DMA to a frame buffer. YLEN = 0 is not valid. DMA transfers data according to YLEN. Each time one or more word(s) been transferred, YLEN is decreased automatically.	R

9.2.17 CIM Cb Frame buffer Address Register (CIMCBFA)

	CIMICBFA																														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
	CbFA																															
RST	0	0	0	0	0	0	0	0	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	CbFA	Cb Frame buffer virtual address in external memory when CIMCFG. SEP is 1. When starts CIM, DMA transfers data from Cb_RXFIFO to frame buffer. This address is increased by hardware automatically. Bits [6:0] must be zero to be aligned to 32-word boundary.	R

9.2.18 CIM Cb DMA Command Register (CIMCBCMD)

	CIMCBCMD																														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								CbLEN																							
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:0	CbLEN	Cb Length of transfer in words. Indicate the number of words to be transferred by DMA to a frame buffer. CbLEN = 0 is not valid. DMA transfers data according to CbLEN. Each time one or more word(s) been transferred, CbLEN is decreased automatically.	R

9.2.19 CIM Cr Frame buffer Address Register (CIMCRFA)

	CIMICRFA																														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
	CrFA																															
RST	0	0	0	0	0	0	0	0	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	CrFA	Cr Frame buffer virtual address in external memory when CIMCFG. SEP is 1. When starts CIM, DMA transfers data from Cr RXFIFO to frame buffer. This address is increased by hardware automatically. CrFA should be (CIMCBFA. CbFA + 8) when the data format is YUV420; Otherwise, bits [6:0] must be zero to be aligned to 32-word boundary.	R

9.2.20 CIM Cr DMA Command Register (CIMCRCMD)

	CIMFID																														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
	Reserved								CrLEN																							
RST	0	0	0	0	0	0	0	0	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	CrLEN	Cr Length of transfer in words. Indicate the number of words to be transferred by DMA to a frame buffer. CrLEN = 0 is not valid. DMA transfers data according to CrLEN. Each time one or more word(s) been transferred, CrLEN is decreased automatically.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.2.21 CIM TLB Control Register (CIMTC)

	CIMTC																														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
	TBA																														RST	ENB
RST	0	0	0	0	0	0	0	0	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	TBA	TLB entry physical address. It should be word aligned, and TBA includes from bit 31 to bit2 only, the low two bits are zeros.	RW
1	RST	TLB reset. 0: Do nothing. 1: Reset TLB logic.	RW
0	ENB	Enable TLB or not. 0: TLB is disabled. 1: TLB is enabled and valid.	RW

9.2.22 CIM TLB Index Register (CIMTINX)

	CIMTINX																														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																														Index	
RST	0	0	0	0	0	0	0	0	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	Read as zeros.	R
2:0	Index	It will be used with CIMCNT together. This field will be used to indicating which buffer will be needed. 0: Indicates VA of entry buffer 0 1: Indicates VA of entry buffer 1 2: Indicates VA of entry buffer 2 3: Indicates VA of entry buffer 3 4: Indicates PA of entry buffer 0 5: Indicates PA of entry buffer 1	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		6: Indicates PA of entry buffer 2 7: Indicates PA of entry buffer 3	

Note:

This register only is used for debug.

9.2.23 CIM TLB Content Register (CIMTCNT)

	CIMTCNT																															0x0064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	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
31:0	CNT	Read the register will get the content corresponding to CIMTINX.Index	R

9.3 CIM Data Sampling Modes

CIM module supports several types of data sampling mode. The modes and the corresponding signals used are shown in the following diagram:

Table 9-3 The modes and the corresponding signals used

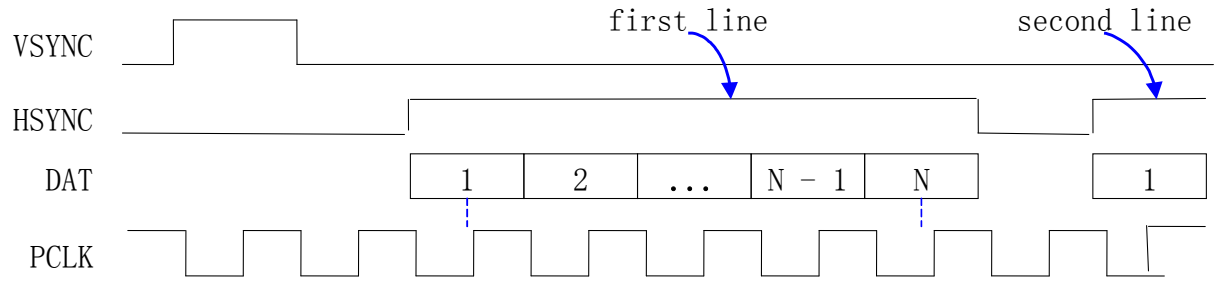
Mode \ Signals	CIM_VSYNC	CIM_HSYNC	CIM_PCLK	CIM_DATA
Gated Clock Mode	Y	Y	Y	Y
ITU656 Interlace Mode	N	N	Y	Y
ITU656 Progressive Mode	N	N	Y	Y

9.3.1 Gated Clock Mode

CIM_VSYNC, CIM_HSYNC, and CIM_PCLK signals are used in this mode.

A frame starts with VSYNC leading edge, then HSYNC goes active and holds the entire line. Data is sampled at the valid edge of PCLK when HSYNC is active; That means, HSYNC functions like “data enable” signal. Please refer to the figure below.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Gated Clock Mode Input Timing Diagram

The VSYNC leading edge, HSYNC active HIGH or LOW, PCLK valid edges are programmable.

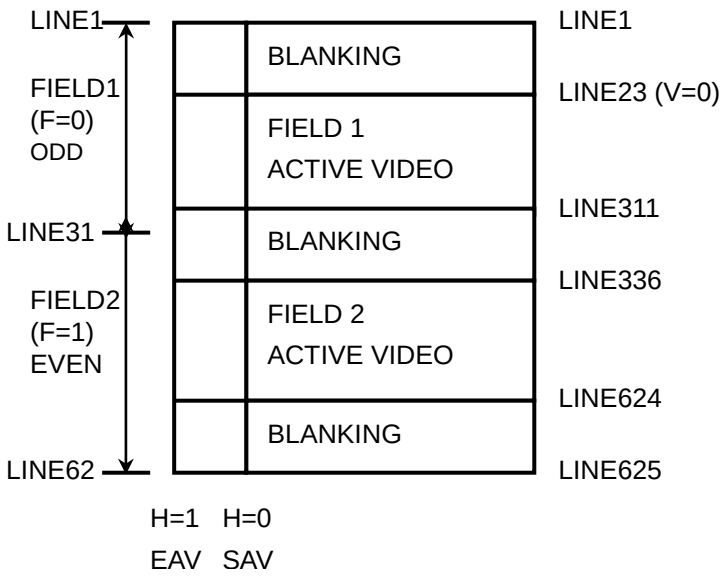
9.3.2 ITU656 Interlace Mode

In this mode, CIM_PCLK and CIM_DAT signals are used, CIM_VSYNC, CIM_HSYNC signals are ignored.

CIM utilizes the SAV & EAV code within ITU656data stream to get active video data.

The following diagrams and tables are quoted from ITU656standard. For more information about ITU656, please refer to ITU656 standard.

9.3.2.1 PAL Timing



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

LINE NUMBER	F	V	H (EAV)	H (SAV)	P0, P1, P2, P3
1-22	0 Field 1	1: blanking	1: in EAV, to indicate the end of active video	0: in SAV, to indicate the start of active video	Protection bits
23-310		0: video data			
311-312		1: blanking			
313-335	1 Field 2	1: blanking			
336-623		0: video data			
624-625		1: blanking			

Figure 9-1 Typical BT.656 Vertical Blanking Intervals for 625/50 Video Systems

9.3.2.2 Coding for Protection Bits

F	V	H	P3	P2	P1	P0
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	1	0
1	1	0	1	1	0	0
1	1	1	0	0	0	1

9.3.3 ITU656 Progressive Mode

CIM_PCLK and CIM_DAT signals are used in this mode. CIM_HSYNC signal is ignored.

CIM_VSYNC is optional in this mode. When the start of frame information is retrieved from SAV and EAV, it is known as internal VSYNC mode. When CIM_VSYNC is provided by sensor directly, it is known as external VSYNC mode. CIM supports both internal and external VSYNC modes.

ITU656Progressive Mode is a kind of Non-Interlace Mode. The image data are encoded within only one field. Most sensors support ITU656Progressive Mode.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

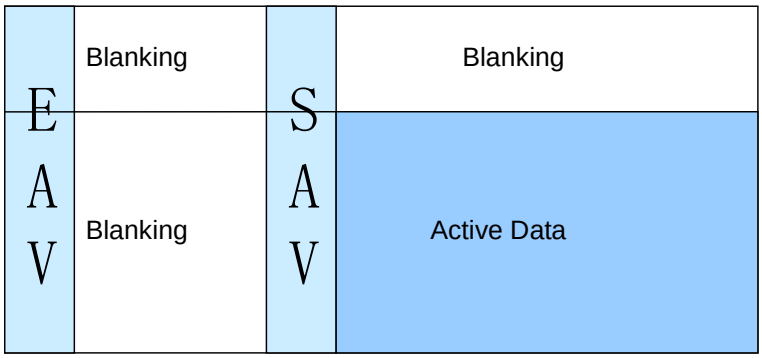


Figure 9-2 ITU656 Progressive Mode

9.4 DMA Descriptors

9.4.1 4-Word Descriptor

Used when output is packaged frame format.

A DMA descriptor is a 4-word block corresponding to the four DMA registers – CIMDA, CIMFA, CIMFID, and CIMCMD, aligned on 4-word (16-byte) boundary, in external memory:

word [0] contains the physical address for next CIMDA
word [1] contains the value for CIMFID
word [2] contains the physical address for CIMFA
word [3] contains the value for CIMCMD

Software must write the physical address of the first descriptor to CIMDA before enabling the CIM. Once the CIM is enabled, the first descriptor is read, and all 4 registers are written by the DMAC. The next DMA descriptor pointed to by CIMDA is loaded into the registers after all data for the current descriptor has been transferred.

9.4.2 8-Word Descriptor

Used when output is separated frame format.

A DMA descriptor is a 8-word block corresponding to the four DMA registers – CIMDA, CIMFA, CIMFID, and CIMCMD, aligned on 8-word (32-byte) boundary, in external memory:

word [0] contains the physical address for next CIMDA
word [1] contains the value for CIMFID
word [2] contains the physical address for CIMYFA
word [3] contains the value for CIMYCMD

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

word [4] contains the physical address for CIMCBFA
word [5] contains the value for CIMCBCMD
word [6] contains the physical address for CIMCRFA
word [7] contains the value for CIMCRCMD

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

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

9.5 Interrupt Generation

CIM has next interrupt sources:

Step 1. RXFIFO Over Flow Interrupt. (RFOF)
When the valid data number of RXFIFO reaches 32 and one more data are written to RXFIFO, CIMST.RFOF bit is set. At the same time, if RFOFM is 1, RFOF interrupt is generated.

Step 2. DMA Start Of Frame Data Transferring Interrupt. (DMA_SOF)
When the CIMCMD.SOFINT bit is 1 and DMA start transferring the first data from RXFIFO to frame buffer, CIMST.DMA_SOF bit is set. At the same time, if DMA_SOFM is 1, DMA_SOF interrupt is generated.

Step 3. DMA End Of Frame Data Transferring Interrupt. (DMA_EOF)
When the CIMCMD.EOFINT bit is 1 and DMA complete transferring the last data from RXFIFO to frame buffer, CIMST.DMA_EOF bit is set. At the same time, if DMA_EOFM is 1, DMA_EOF interrupt is generated.

Step 4. DMA Stop Transferring Interrupt. (DMA_STOP)
When the CIMCMD.STOP bit is 1 and DMA complete transferring the last data from RXFIFO to frame buffer, CIMST.DMA_STOP bit is set. At the same time, if DMA_STOPM is 1, DMA_STOP interrupt is generated.

Step 5. CIM Disable Done Interrupt. (VDD)
When disable the module by clearing the CIMCR.ENA, the module should be disabled after transferring current valid data. Then set the CIMST.VDD bit, at the same time, if VDDM is set, VDD interrupt is generated.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

9.6 Software Operation

9.6.1 Enable CIM with DMA

Step 1. Configure register CIMCFG.
Step 2. Prepare frame buffer and descriptors.
Step 3. Configure register CIMDA.
Step 4. Clear state register: write 0 to register CIMSTATE.
Step 5. Reset RXFIFO: configure register CIMCTRL with DMA_EN=1, RXF_RST=1, ENA=0.
Step 6. Stop resetting RXFIFO: configure register CIMCTRL with DMA_EN=1, RXF_RST=0, ENA=0.
Step 7. Enable CIM: configure register CIMCTRL with DMA_EN=1, RXF_RST=0, ENA=1.

9.6.2 Enable CIM without DMA

1 Configure register CIMCFG.
2 Clear state register: write 0 to register CIMSTATE.
3 Reset RXFIFO: configure register CIMCTRL with DMA_EN=0, RXF_RST=1, ENA=0.
4 Stop resetting RXFIFO: configure register CIMCTRL with DMA_EN=0, RXF_RST=0, ENA=0.
5 Enable CIM: configure register CIMCTRL with DMA_EN=0, RXF_RST=0, ENA=1.

9.6.3 Disable CIM

Method 1:

Step 1. Configure register CIMCTRL with RXF_RST=0, ENA=0. // quick disable
Step 2. Clear state register: write 0 to register CIMSTATE.

Method 2:

When DMA is enabled, the following sequence is recommended:

Step 1. Configure descriptor with STOP = 1.
Step 2. Wait DMA_STOP interrupt, then write 0 to CIMCTRL.ENA.
Step 3. Clear state register: write 0 to register CIMSTATE.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10 AC97/I2S/SPDIF Controller

10.1 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.1.1 Block Diagram

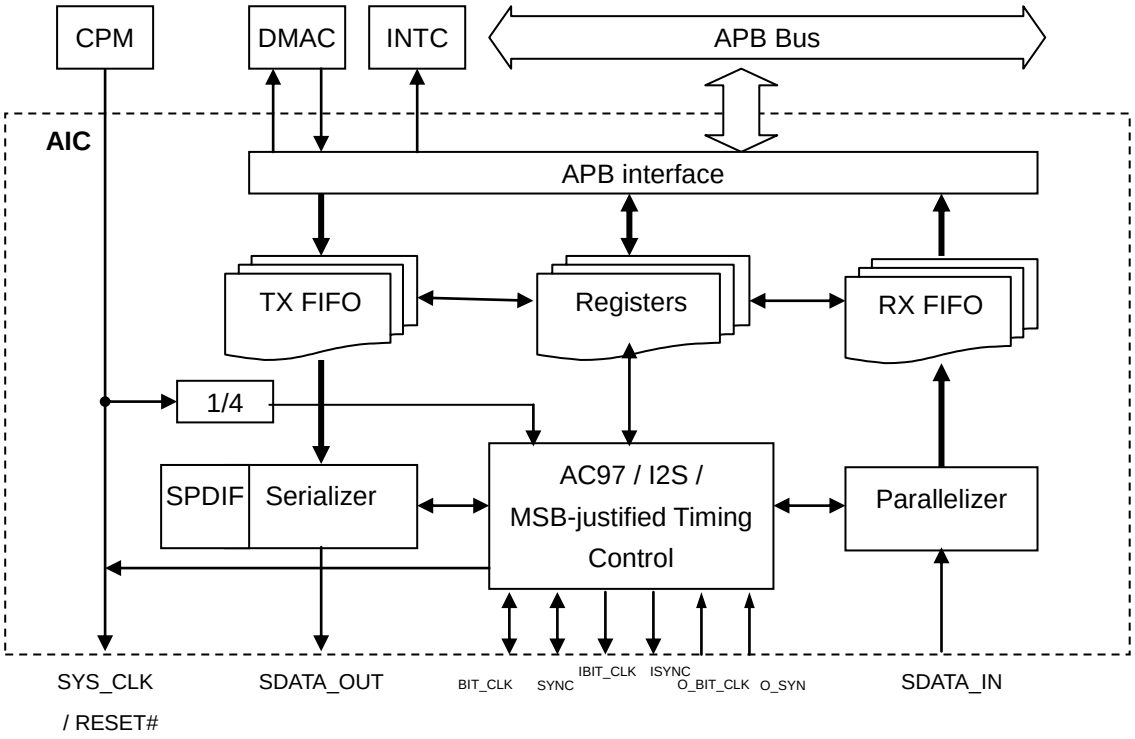


Figure 10-1 AIC Block Diagram

The O_BIT_CLK and O_SYNC ports are only used by inter CODEC.

10.1.2 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 © 2005-2015 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

10.1.3 Interface Diagram

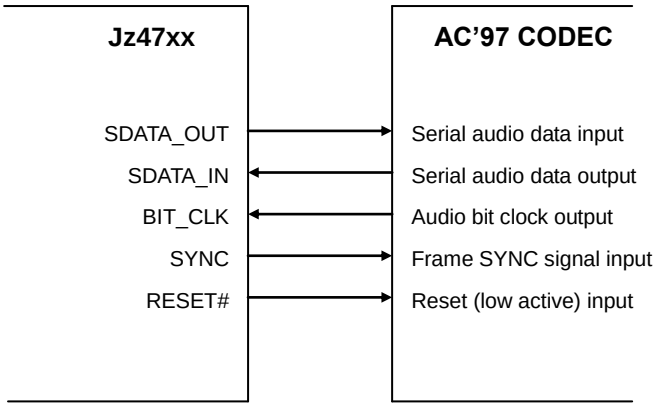


Figure 10-2 Interface to an External AC'97 CODEC Diagram

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

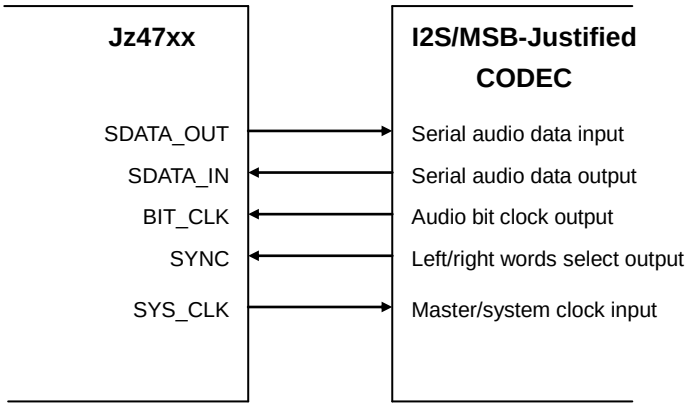


Figure 10-3 Interface to an External Master Mode I2S/MSB-Justified CODEC Diagram (Share Clock Mode)

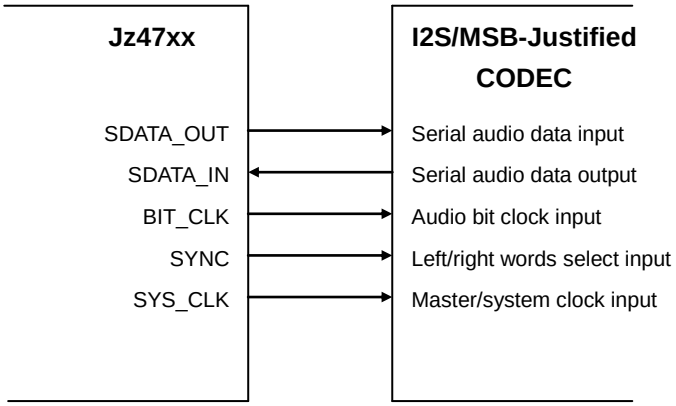


Figure 10-4 Interface to an External Slave Mode I2S/MSB-Justified CODEC Diagram (Share Clock Mode)

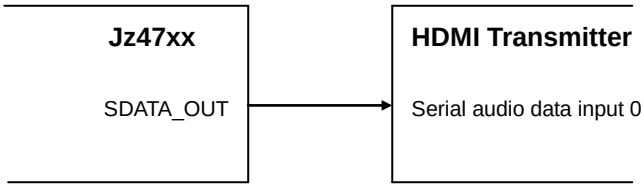


Figure 10-5 Interface to a HDMI Transmitter via SPDIF Diagram

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

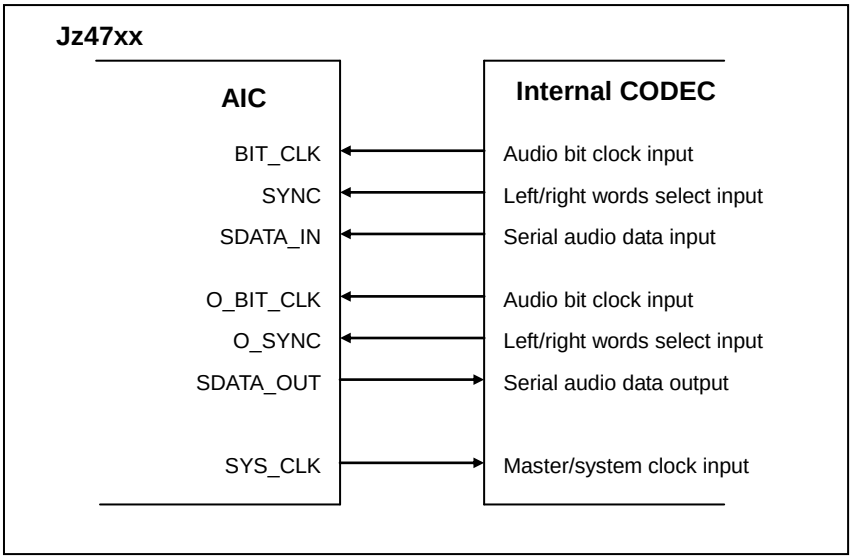


Figure 10-6 Interface to an internal Master Mode I2S CODEC Diagram

Please refer to the related CODEC specification for the details.

10.1.4 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 Table 1-1.

Table 10-1 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.1.4.1 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.

10.1.4.2 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. Table10-7 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.

10.1.4.3 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.

10.1.4.4 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.1.4.5 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.

10.2 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 10-2 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
RGADW	Address, data in and write	RW	0x00000000	0x100200A4	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	command for accessing to internal registers of embedded CODEC				
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.

10.2.1 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					IBCKD	ISYNCD	DMODE	Reserved	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
27:24	RFTH	Receive FIFO threshold for interrupt or DMA request. The RFTH valid value is 0 ~ 15.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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></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>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>Reserved</td><td>Writing has no effect, read as zero.</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).
<table>
<tr><th>AUSEL</th><th>Selected</th></tr>
<tr><td>0</td><td>Select AC-link format.</td></tr>
<tr><td>1</td><td>Select I2S/MSB-justified format.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>3</td><td>RST</td><td>Reset AIC. Write 1 to this bit reset AIC registers and FIF0 except AICFR and I2SDIV register. Writing 0 to this bit has no effect and this bit is always reading 0.</td><td>W</td></tr>
<tr><td>2</td><td>BCKD</td><td>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).
<table>
<tr><th>BCKD</th><th>BIT_CLK Direction</th></tr>
<tr><td>0</td><td>BIT_CLK is input from an external source.</td></tr>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1	BIT_CLK is generated internally and driven out to the CODEC.
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.**

10.2.2 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.

	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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>word contains two sample data or only one sample data with LSB align. The packed mode is only support 16bit sample size.
<table>
<tr><th>PACK16</th><th>Sample Size</th></tr>
<tr><td>0</td><td>Unpacked data mode.
One word only contains one 16bit sample data aligned LSB.</td></tr>
<tr><td>1</td><td>Packed data mode.
One word contains two 16 bit sample data.</td></tr>
</table></td><td></td></tr>
<tr><td>27</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>26:24</td><td>CHANNEL</td><td>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.
<table>
<tr><th>CHANNEL</th><th>Sample Size</th></tr>
<tr><td>0x0</td><td>1 channel, mono mode.</td></tr>
<tr><td>0x1</td><td>2 channels, stereo mode.</td></tr>
<tr><td>0x2 ~ 0x7</td><td>Reserved.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>23:22</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>21:19</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, 18, 20 and 24 bits. The sample data is LSB-justified in memory/register.
<table>
<tr><th>OSS</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>RW</td></tr>
<tr><td>18:16</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, 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>RW</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>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<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><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>RW</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</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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></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.<table><tr><th>EREC</th><th>Description</th></tr><tr><td>0</td><td>AIC Recording Function is Disabled.</td></tr><tr><td>1</td><td>AIC Recording Function is Enabled.</td></tr></table></td><td>RW</td></tr>
</table>

10.2.3 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.

<table>
<tr><td></td><td colspan="16">ACCR1</td><td colspan="16">0x10020008</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="6">Reserved</td><td colspan="10">RS</td><td colspan="6">Reserved</td><td colspan="10">XS</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></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

10.2.4 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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>enable or disable.<table><tr><th>ERSTO</th><th>RSTO Interrupt</th></tr><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td><td></td></tr>
<tr><td>17</td><td>ESADR</td><td>Enable SADR Interrupt. This bit is used to control the SADR interrupt enable or disable.<table><tr><th>ESADR</th><th>SADR 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>16</td><td>ECADT</td><td>Enable CADT Interrupt. This bit is used to control the CADT interrupt enable or disable.<table><tr><th>ECADT</th><th>CADT 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>15:4</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>3</td><td>SO</td><td>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.</td><td>RW</td></tr>
<tr><td>2</td><td>SR</td><td>RESET# pin level. When AC-link is selected, this bit is used to drive the RESET# pin.<table><tr><th>SR</th><th>RESET# Pin Voltage Level</th></tr><tr><td>0</td><td>High.</td></tr><tr><td>1</td><td>Low.</td></tr></table></td><td>RW</td></tr>
<tr><td>1</td><td>SS</td><td>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.</td><td>RW</td></tr>
<tr><td>0</td><td>SA</td><td>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 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.<table><tr><th>SA</th><th>SS</th><th>Description</th></tr><tr><td rowspan="4">0</td><td rowspan="2">0</td><td>When read, indicated SYNC is 0.</td></tr><tr><td>When write, not effect.</td></tr><tr><td rowspan="2">1</td><td>When read, indicated SYNC is 1.</td></tr><tr><td>When write, not effect.</td></tr><tr><td rowspan="4">1</td><td rowspan="2">0</td><td>When read, indicated SYNC is 0.</td></tr><tr><td>When write, SYNC is driven to 0.</td></tr><tr><td rowspan="2">1</td><td>When read, indicated SYNC is 1.</td></tr><tr><td>When write, SYNC is driven to 1.</td></tr></table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.2.5 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** 0 – **16 bit LSB and 16bit MSB is not switched.** 1 – **16 bit LSB and 16bit MSB is switched.**	RW
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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		AICFR.BCKD.	
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

10.2.6 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** 0x00 ~ 0x20 / RFL valid PCM data in receive FIFO. 0x21 ~ 0x3F / Reserved.	R
23:14	Reserved	Writing has no effect, read as zero.	R
13:8	TFL	Transmit FIFO Level. The bits indicate the amount of valid PCM data in Transmit FIFO. **TFL Value** / **Description** 0x00 ~ 0x3F / TFL valid PCM data in transmit FIFO.	R
7	Reserved	Writing has no effect, read as zero.	R
6	ROR	Receive FIFO Over Run. This bit indicates that receive FIFO has or has not experienced an overrun. **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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		When write, not effects.	
5	TUR	Transmit FIFO Under Run. This bit indicates that transmit FIFO has or has not experienced an under-run. **TUR** / **Description** 0: When read, indicates under-run has not been found. 0: When write, clear itself. 1: When read, indicates data has even been read from empty transmit FIFO. 1: When write, not effects.	RW
4	RFS	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. **RFS** / **Description** 0: Receive FIFO level below RFL threshold. 1: Receive FIFO level at or above RFL threshold.	R
3	TFS	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. **TFS** / **Description** 0: Transmit FIFO level exceeds TFL threshold. 1: Transmit FIFO level at or below TFL threshold.	R
2:0	Reserved	Writing has no effect, read as zero.	R

10.2.7 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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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.</td><td></td></tr>
<tr><td>20</td><td>CRDY</td><td>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.
<table>
<tr><th>CRDY</th><th>Description</th></tr>
<tr><td>0</td><td>CODEC is not ready.</td></tr>
<tr><td>1</td><td>CODEC is ready.</td></tr>
</table></td><td>R</td></tr>
<tr><td>19</td><td>CLPM</td><td>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.
<table>
<tr><th>CLPM</th><th>Description</th></tr>
<tr><td>0</td><td>BIT_CLK is active.</td></tr>
<tr><td>1</td><td>CODEC is switched to low power mode.</td></tr>
</table></td><td>R</td></tr>
<tr><td>18</td><td>RSTO</td><td>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.
<table>
<tr><th>RSTO</th><th>Description</th></tr>
<tr><td>0</td><td>When read, indicates time out has not occurred.</td></tr>
<tr><td>1</td><td>When read, indicates read status time out found.</td></tr>
</table>
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.</td><td>RW</td></tr>
<tr><td>17</td><td>SADR</td><td>External CODEC Registers Status Address and Data Received. This bit indicates that address and data of an external AC ’97 CODEC register 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>RW</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>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.2.8 AIC I2S/MSB-justified Status Register (I2SSR)

I2SSR is used to reflect AIC status in I2S/MSB-justified. It is a read-only register.

I2SSR																													0x1002001c
Bit	31	30	29	28	27	26	25	24	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																										CHBSY	TBSY	RBSY	BSY	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
31:3	Reserved	Writing has no effect, read as zero.	R
5	CHBSY	AIC Transmitter busy in I2S/MSB-justified format.(Multi-channel status) **CHBSY** / **Description** 0 / AIC Transmitter part is idle or disabled. 1 / AIC Transmitter part currently is transmitting or receiving a frame.	R
4	TBSY	AIC Transmitter busy in I2S/MSB-justified format. **TBSY** / **Description** 0 / AIC Transmitter part is idle or disabled. 1 / AIC Transmitter part currently is transmitting or receiving a frame.	R
3	RBSY	AIC Receiver busy in I2S/MSB-justified format. **RBSY** / **Description** 0 / AIC Receiver part is idle or disabled. 1 / AIC Receiver part currently is transmitting or receiving a frame.	R
2	BSY	AIC busy in I2S/MSB-justified format. **BSY** / **Description** 0 / AIC controller is idle or disabled. 1 / AIC controller currently is transmitting or receiving a frame.	R
1:0	Reserved	Writing has no effect, read as zero.	R

10.2.9 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 10.4.4 for software flow. These registers are valid only in AC-link. It is ignored in I2S/MSB-justified format.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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

10.2.10 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 10.4.4 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

10.2.11 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.

	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 can be even or odd and I2SDIV.IDV should be set to the dividing value minus 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 can be even or odd and I2SDIV.IDV should be set to the dividing value minus 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

10.2.12 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.

	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

10.2.13 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

10.2.14 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

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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Start reset	
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

10.2.15 SPDIF State Register (SPSTATE)

The register SPSTATE is used to keep the state of SPDIF.

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

10.2.16 SPDIF Configure 1 Register (SPCFG1)

The register SPCFG1 is used to configure SPDIF.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 1: Invalid	RW
15:14	Reserved	Writing has no effect, read as zero.	R
13:12	TRIG	Specify the trigger value of FIFO.	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

TRIG	Description
00	Trigger Value is 4.
01	Trigger Value is 8.
10	Trigger Value is 16.
11	Trigger Value is 32.

10.2.17 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 1100:32kHz 0101:96kHz 0001:192kHz Others: Reference IEC60958-3	RW
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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		11: Interface frame rate not matched to sampling frequency	
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. 0: Consumer mode 1: Professional mode Professional is not supported in the chip.	RW

10.2.18 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.3 Serial Interface Protocol

10.3.1 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.

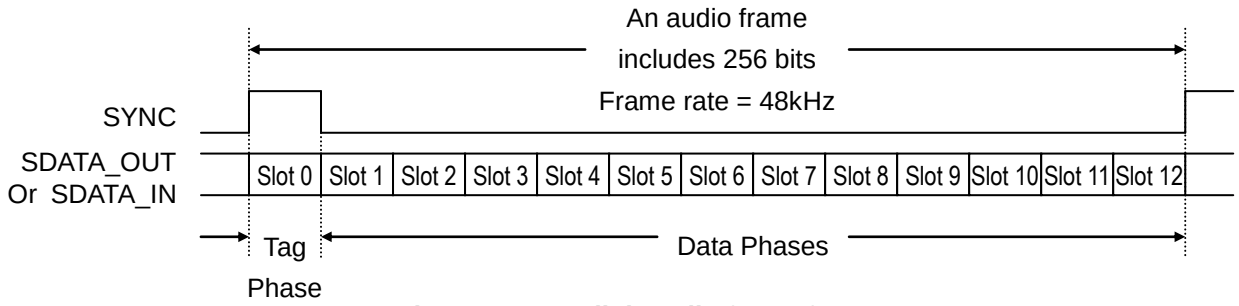


Figure 10-7 AC-link audio frame format

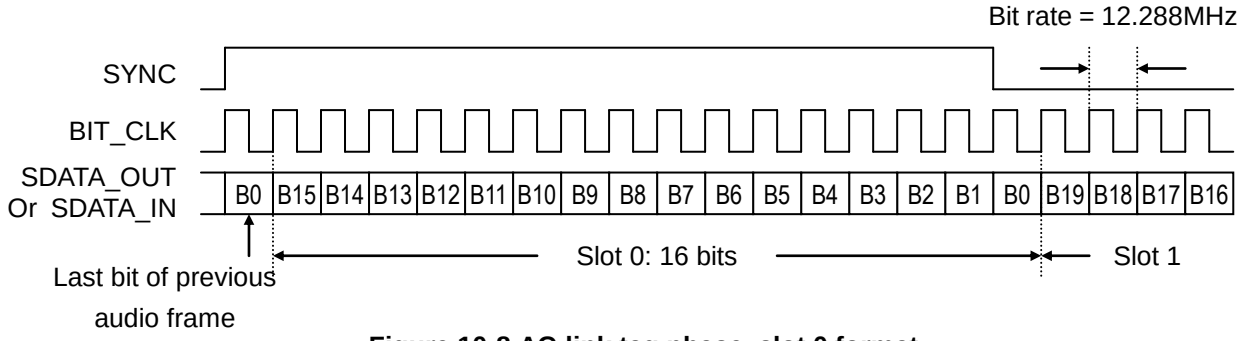


Figure 10-8 AC-link tag phase, slot 0 format

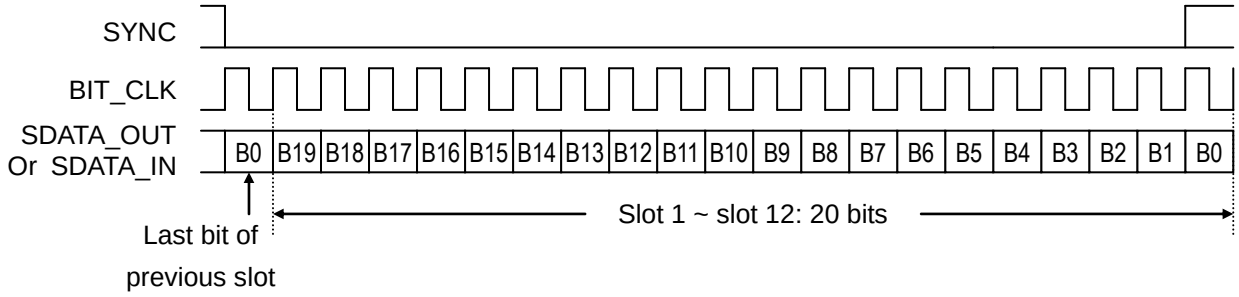


Figure 10-9 AC-link data phases, slot 1 ~ slot 12 format

10.3.2 I2S and MSB-justified serial audio format

Normal I2S and MSB-justified are similar protocols for digitized stereo audio transmitted over a

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

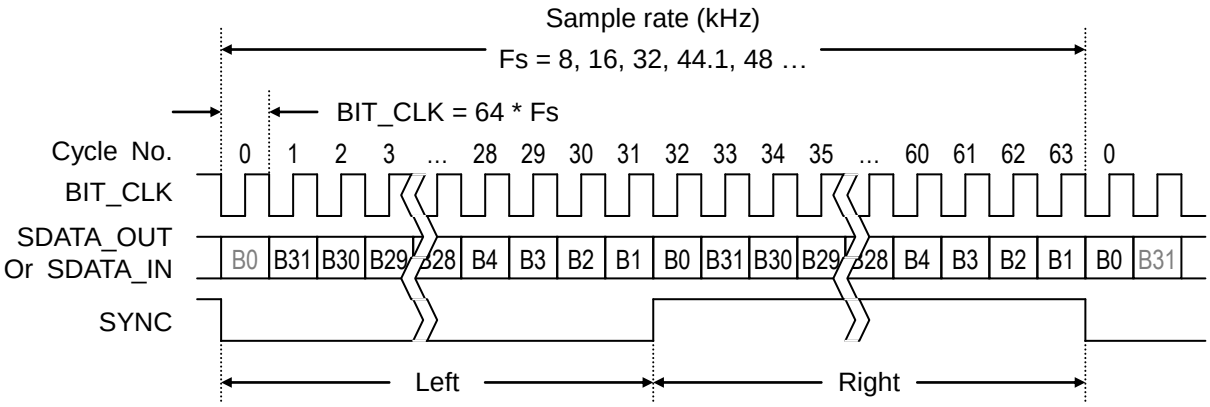


Figure 10-10 I2S data format (A: LR mode)

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.

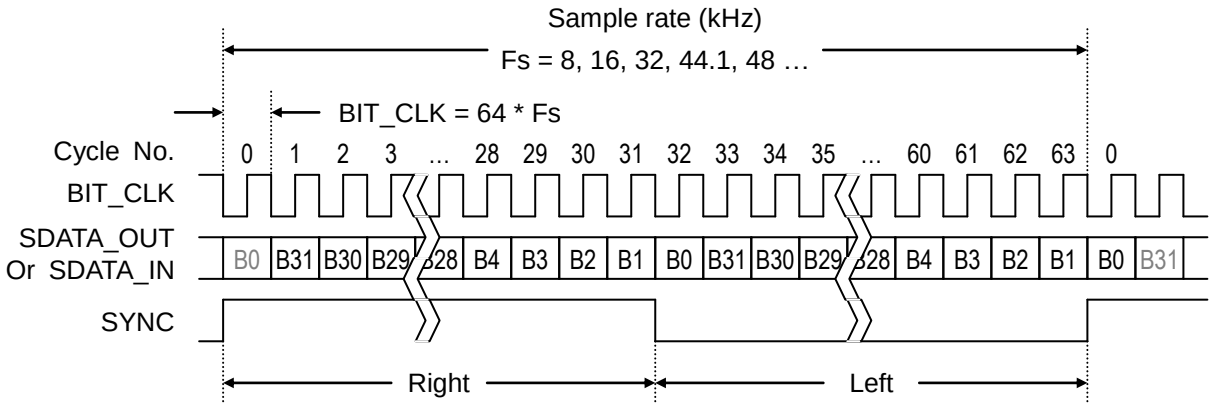


Figure 10-11 I2S data format (B: RL mode)

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

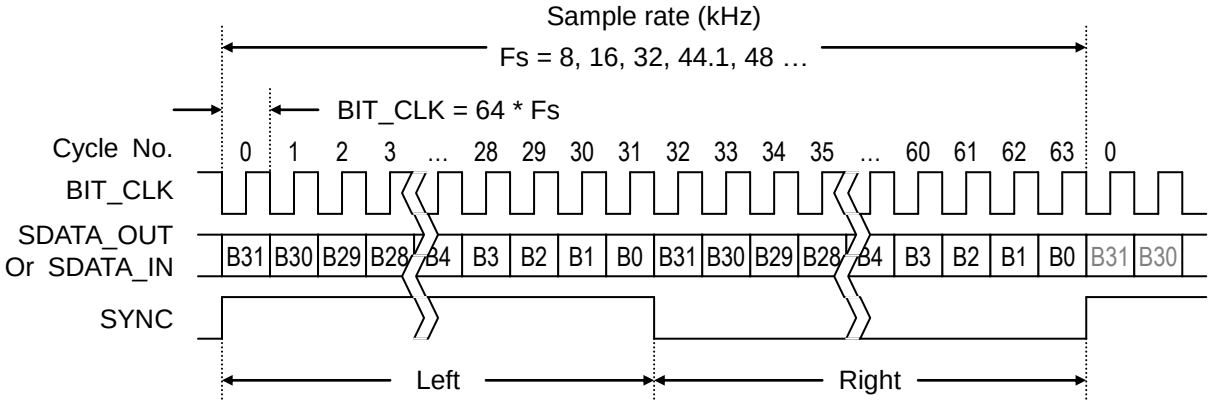


Figure 10-12 MSB-justified data format (C: LR mode)

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.

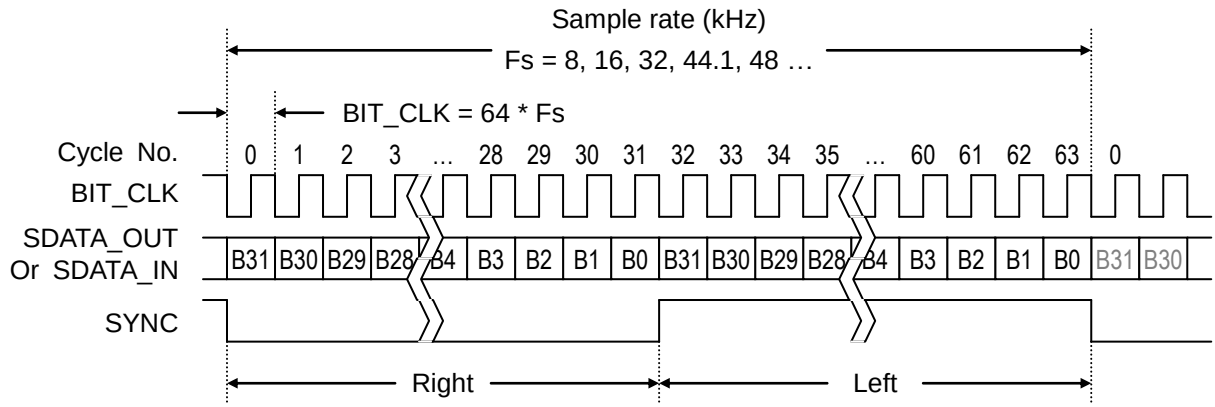


Figure 10-13 MSB-justified data format (D: RL mode)

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 10-10 and Figure 10-12 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

10.3.3 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.

Table 10-3 Sample data bit relate to SDATA_IN/SDATA_OUT bit

AC-link Format						I2S/MSB-Justified Format					
SDATA IN/OUT	Audio Sample Size (bit)					SDATA IN/OUT					
	8	16	18	20	24		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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

						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.

10.3.4 SPDIF Protocol

SPDIF block format is shown below:

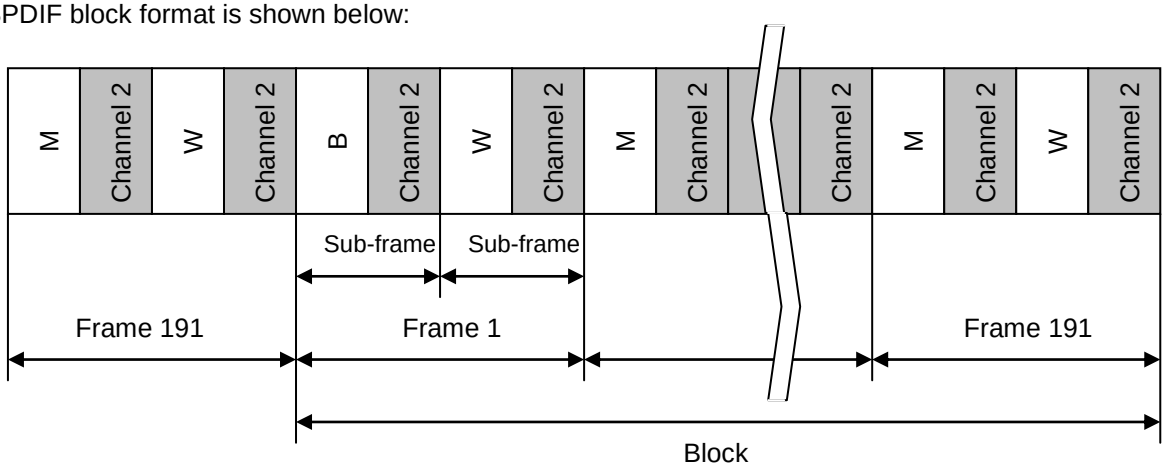


Figure 10-14 Block format

Sub-frame format in PCM mode is shown below:

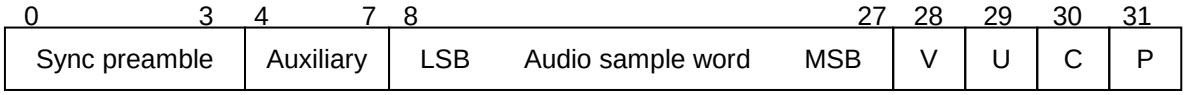


Figure 10-15 Sub-frame format in PCM mode

Sub-frame format in non-PCM mode is shown below:

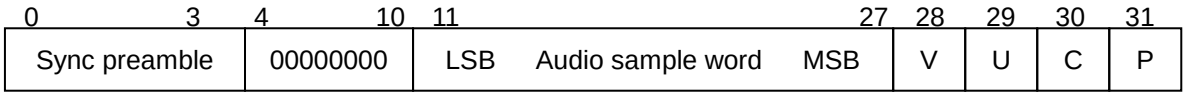


Figure 10-16 Sub-frame format in non-PCM mode

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4 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.
- **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 10.2.1 for further details regarding AICFR.TFTH.
- The receive FIFO request is based on the receive trigger-threshold (AICFR.RFTH) setting. See 10.2.1 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.

10.4.1 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 10-7. 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.

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.

10.4.2 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).

10.4.3 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.

10.4.3.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

RESET# is an asynchronous AC ‘97 CODEC input.

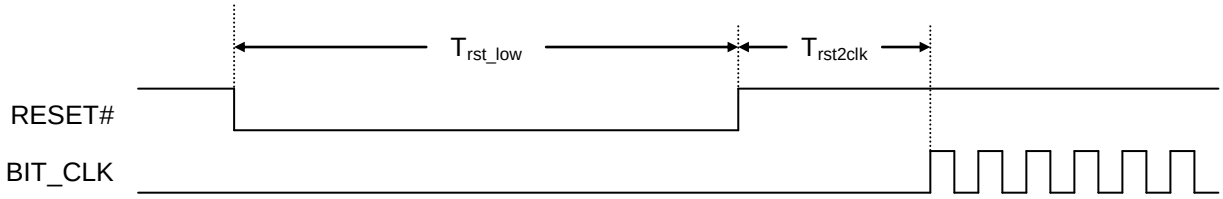


Figure 10-17 Cold AC ’97 CODEC Reset Timing

Table 10-4 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

10.4.3.2 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.

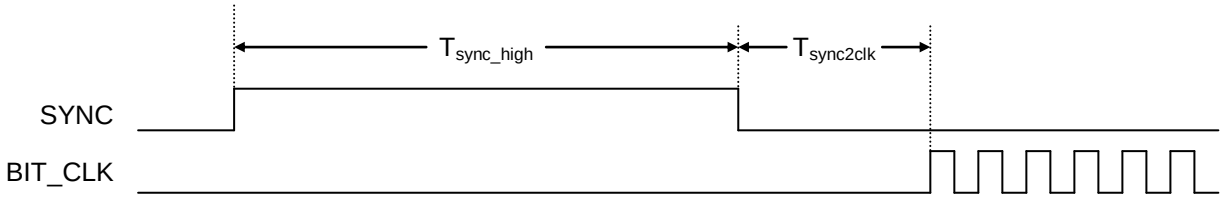


Figure 10-18 Warm AC ’97 CODEC Reset Timing

Table 10-5 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4.4 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.

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.

10.4.5 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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).
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 10.4.7 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.

10.4.6 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:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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).
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 10.4.7 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.

10.4.7 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 10-19 illustrates the FIFOs access.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

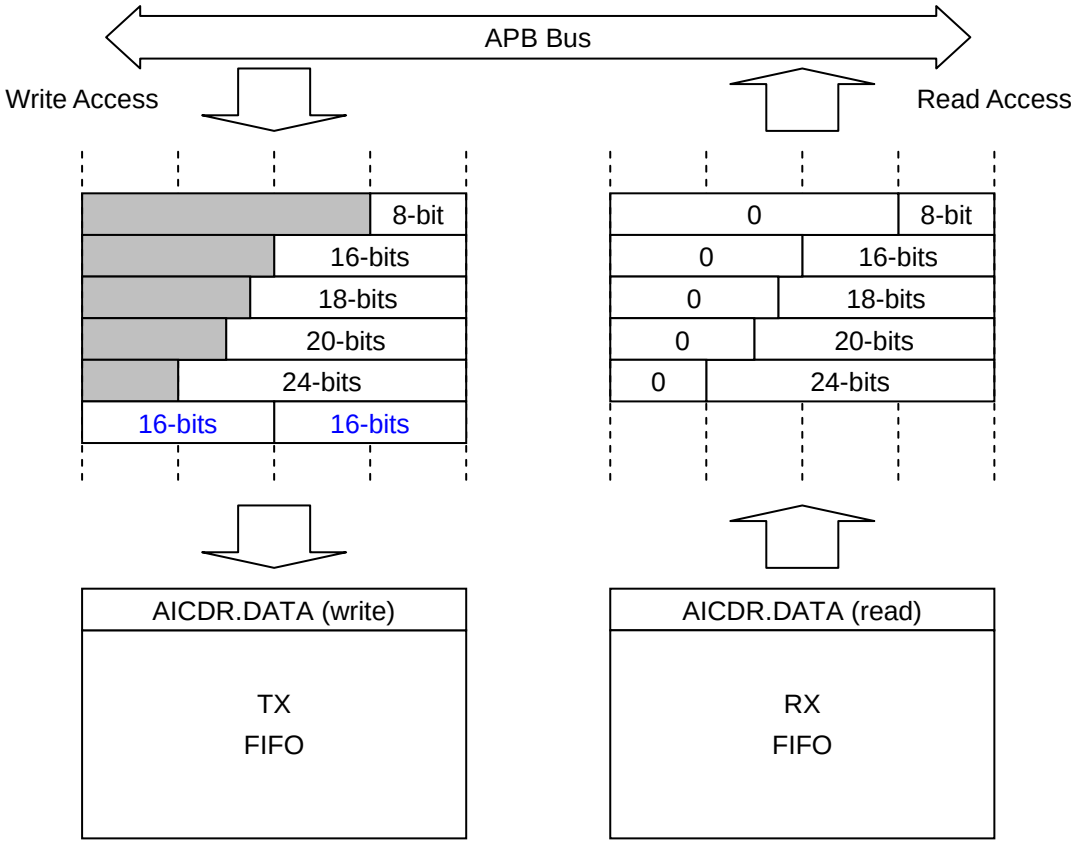


Figure 10-19 Transmitting/Receiving FIFO access via APB Bus

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 © 2005-2015 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).

10.4.8 Data Flow Control

There are three approaches provided to control/synchronize the audio incoming/outgoing data flow.

10.4.8.1 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.

10.4.8.2 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

10.4.8.3 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.

10.4.9 Audio Samples format

10.4.9.1 16 bits packed mode

One channel (mono) mode and two channels (stereo) mode:

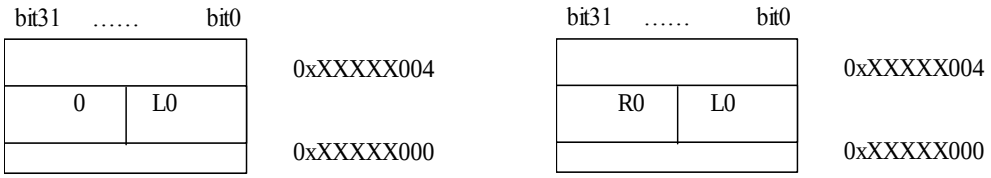


Figure 10-20 One channel (Left) and Two channels (right) mode (16 bits packed mode)

Four channels mode and six channels mode:

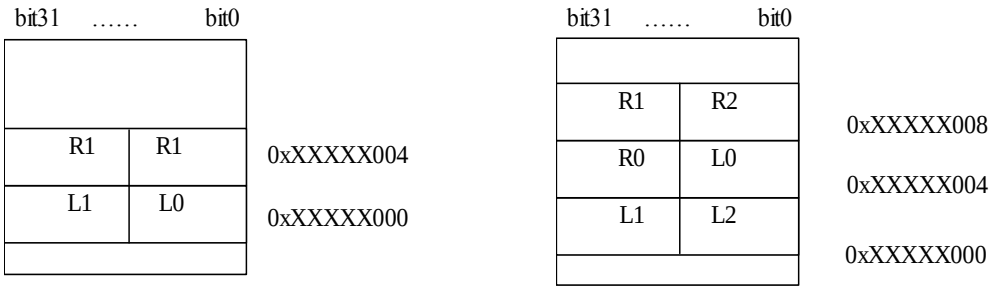


Figure 10-21 Four channels (Left) and Six channels (right) mode (16 bits packed mode)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Eight channels mode:

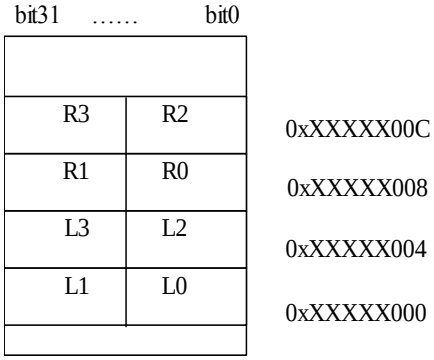


Figure 10-22 Eight channels mode (16 bits packed mode)

10.4.9.2 Normal mode.

One channel (Mono) and two channels (stereo) mode:

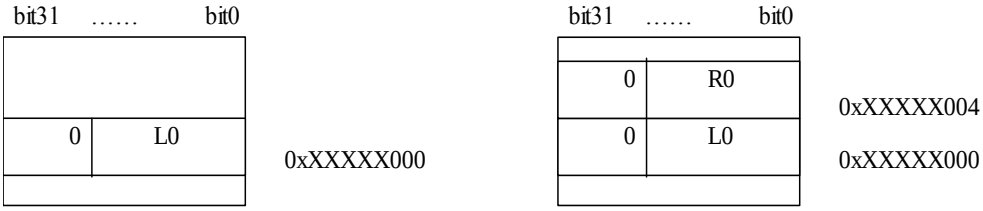


Figure 10-23 One channel (Left) and Two channels (right) mode

Four channels mode and six channels mode:

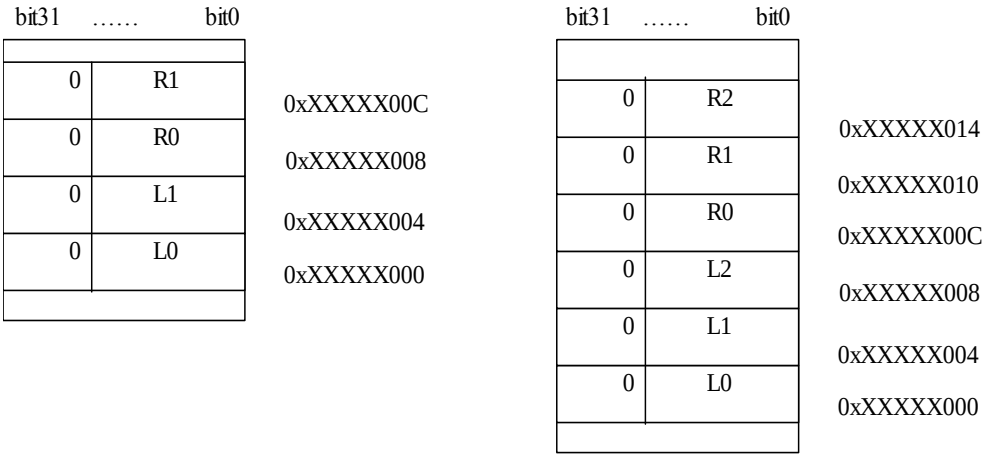


Figure 10-24 Four channels (Left) and Six channels (right) mode

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Eight channel mode:

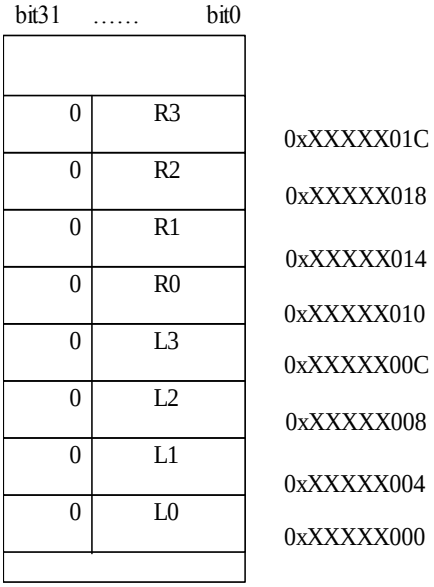


Figure 10-25 Eight channels mode

10.4.10 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, $f_{BIT\_CLK} = 64\ f_{S.}$ But SYNC signal frequency is not fixed when using internal CODEC.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 10-3 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 10-6 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.

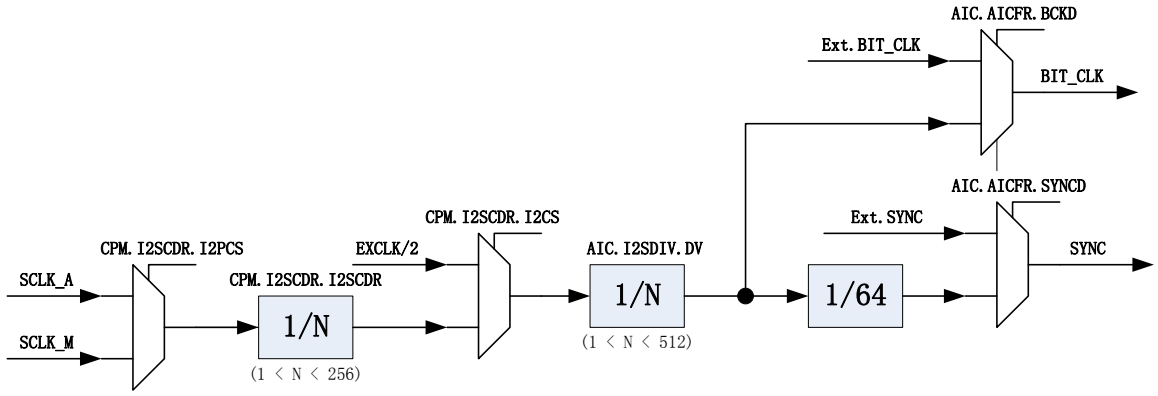


Figure 10-26 SYS_CLK, BIT_CLK and SYNC generation scheme

Note: SCLK_A and SCLK_M please refer to CPM.

The setting of I2SDIV.DV is shown in Following table.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 10-7 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 10-6, 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, Table 10-8 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 10-8 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
353.13	0.90
358.79	0.69
370.59	0.53

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

382.96	0.54
394.17	0.24

Take PLL = 270.64 MHz as an example, Table 1-9 lists the divider settings for various sample rates.

Table 10-9 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 10-8 as possible. Table 10-10 lists the PLL parameters and audio sample errors at different PLL frequencies for EXCLK at 12MHz.

Table 10-10 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%
178	7	305.14	0.67%
53	2	318	0.60%
302	11	329.45	0.70%
256	9	341.33	0.79%

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

10.4.11 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.

10.5 SPDIF Guide

10.5.1 Set SPDIF clock frequency

Set SPDIF clock frequency is as same as i2s clock.

10.5.2 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

10.5.3 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.

10.5.4 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11 PCM Interface

11.1 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

11.2 Pin Description

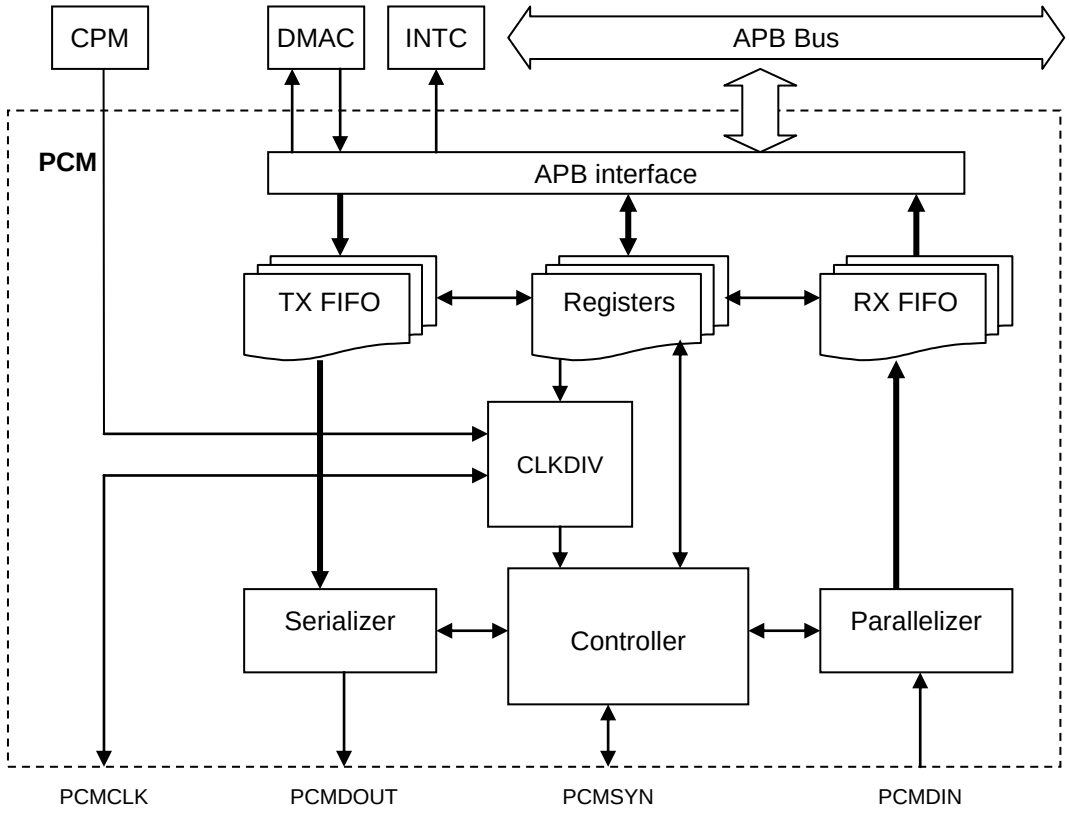
There are all 4 pins used to connect between PCM interface and an external device. They are listed and described in following table.

Table 11-1 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.3 Block Diagram



11.4 Register Description

Table 11-2 PCM Registers Description

Name	Description	RW	Reset Value	Address	Size
PCMCTL0	PCM Control Register	RW	0x00000000	0x10071000	32
PCMCFG0	PCM Configure Register	RW	0x00000110	0x10071004	32
PCMDP0	PCM FIFO Data Port Register	RW	0x00000000	0x10071008	32
PCMINTC0	PCM Interrupt Control Register	RW	0x00000000	0x1007100c	32
PCMINTS0	PCM Interrupt Status Register	RW	0x00000100	0x10071010	32
PCMDIV0	PCM Clock Divide Register	RW	0x00000001	0x10071014	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.4.1 PCM Control Register (PCMCTL)

	PCMCTL0																															0x10071000
Bit	31	30	29	28	27	26	25	24	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. **ETDMA** / **Transmit DMA** 0 / Disabled. 1 / Enabled.	RW
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

11.4.2 PCM Configuration Register (PCMCFG)

PCMCFG0 **0x10071004**

Bit	31	30	29	28	27	26	25	24	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
31:15	Reserved	Writing has no effect, read as zero.	R
14:13	SLOT	Controls the amount of valid PCM timeslot in one PCMSYN frame. **SLOT** / **Number of slot in one frame** 00 – 1 slot 01 – 2 slot 10 – 3 slot 11 – 4 slot	RW
12	ISS	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. **ISS** / **Sample Size** 0 – 8 bit 1 – 16 bit	RW
11	OSS	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. **OSS** / **Sample Size** 0 – 8 bit 1 – 16 bit	RW
10	IMSBPOS	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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		cycle that PCMSYNC is high 1: MSB is captured on the falling edge of PCMCLK during the cycle after the PCMSYNC is high	
9	OMSBPOS	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	RW
8:5	RFTH	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.	RW
4:1	TFTH	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 dependent.	RW
0	PCMMOD	PCM mode select. 0: Master mode; 1: Slave mode.	RW

11.4.3 PCM FIFO DATA PORT REGISTER (PCMDP)

	PCMDP0																0x10071008													
Bit	31	30	29	28	27	26	25	24	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

11.4.4 PCM INTERRUPT CONTROL REGISTER (PCMINTC)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PCMINTCR0																												0x1007100c			
Bit	31	30	29	28	27	26	25	24	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 enable or disable. **ERFS** / **RFS Interrupt** 0 / Disabled. 1 / Enabled.	RW
0	EROR	Enable ROR Interrupt. This bit is used to control the ROR interrupt enable or disable. **EROR** / **ROR Interrupt** 0 / Disabled. 1 / Enabled.	RW

11.4.5 PCM INTERRUPT STATUS REGISTER (PCMINTS)

	PCMINTS0																												0x10071010			
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31:15</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>14</td><td>RSTS</td><td>Soft reset / flush state.
0: Nothing / reset or flush operation has completed
1: reset or flush operation has not completed</td><td>R</td></tr>
<tr><td>13:9</td><td>TFL</td><td>Transmit FIFO Level. The bits indicate the amount of valid PCM data in Transmit FIFO.</td><td>R</td></tr>
<tr><td>8</td><td>TFS</td><td>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.
<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>RW</td></tr>
<tr><td>7</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>6:2</td><td>RFL</td><td>Receive FIFO Level. The bits indicate the amount of valid PCM data in Receive FIFO.</td><td>R</td></tr>
<tr><td>1</td><td>RFS</td><td>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.
<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>RW</td></tr>
<tr><td>0</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>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.4.6 PCM CLOCK DIVIDE REGISTER (PCMDIV)

	PCMDIV0																														0x10071014	
Bit	31	30	29	28	27	26	25	24	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

11.5 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11.5.1 Short Frame SYN

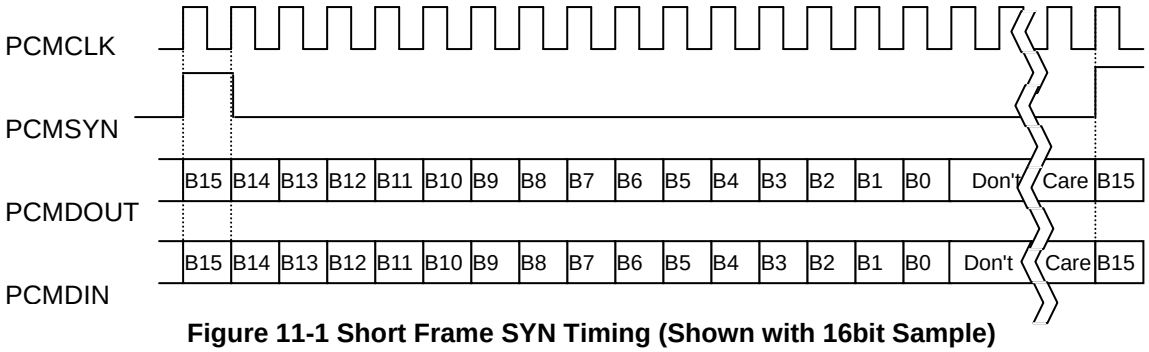


Figure 11-1 Short Frame SYN Timing (Shown with 16bit Sample)

NOTE: Above figure shows a PCM transfer with the MSB configured to be coincident with the PCMSYN.

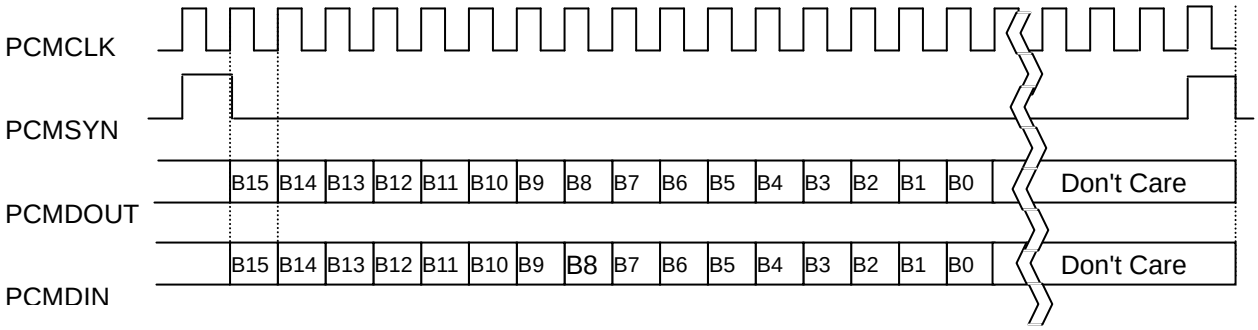


Figure 11-2 Short Frame SYN Timing (Shown with 16bit Sample)

NOTE: Above figure shows a PCM transfer with the MSB configured one shift clock after the PCMSYN.

11.5.2 Long Frame SYN

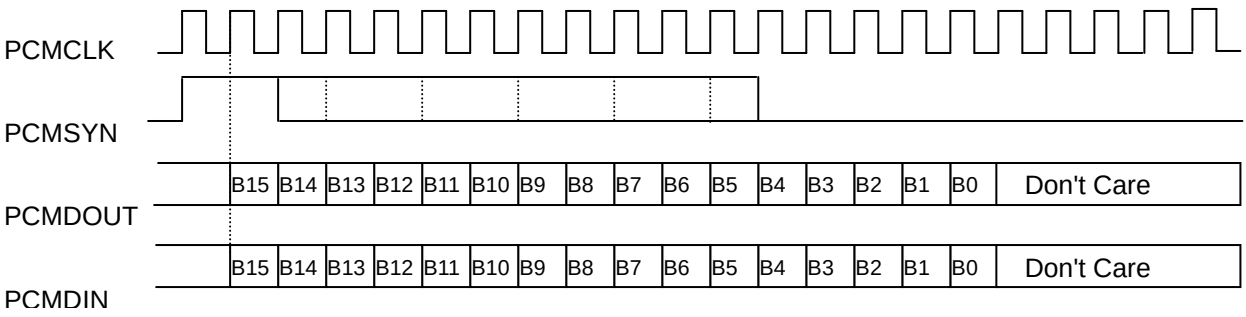


Figure 11-3 Long Frame SYN Timing (Shown with 16bit Sample)

NOTE: Above figure shows a PCM transfer with the MSB configured one shift clock after the

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

PCMSYN.

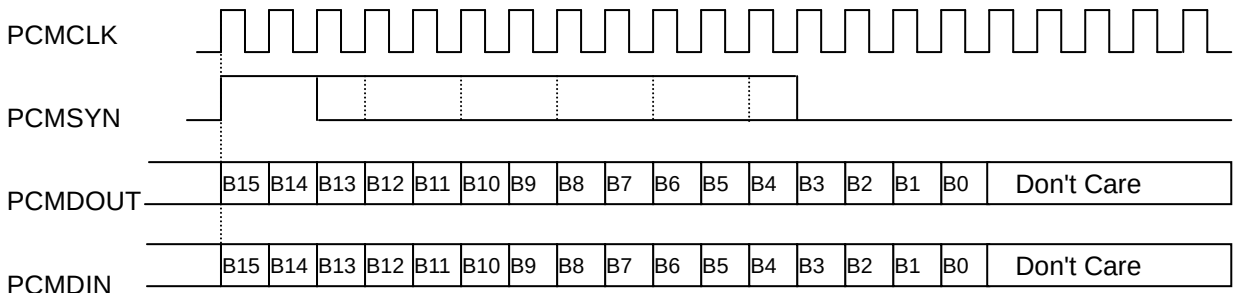


Figure 11-4 Long Frame SYN Timing (Shown with 16bit Sample)

NOTE: Above figure shows a PCM transfer with the MSB configured to be coincident with the PCMSYN.

11.5.3 Multi-Slot Operation

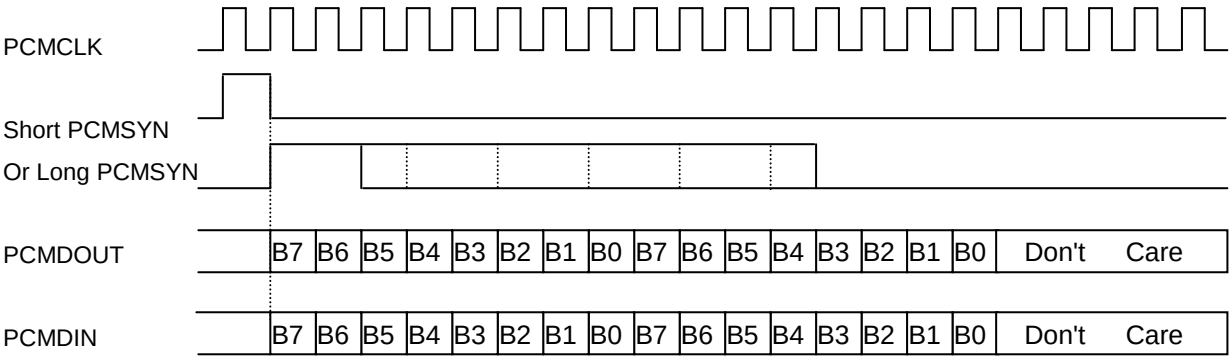


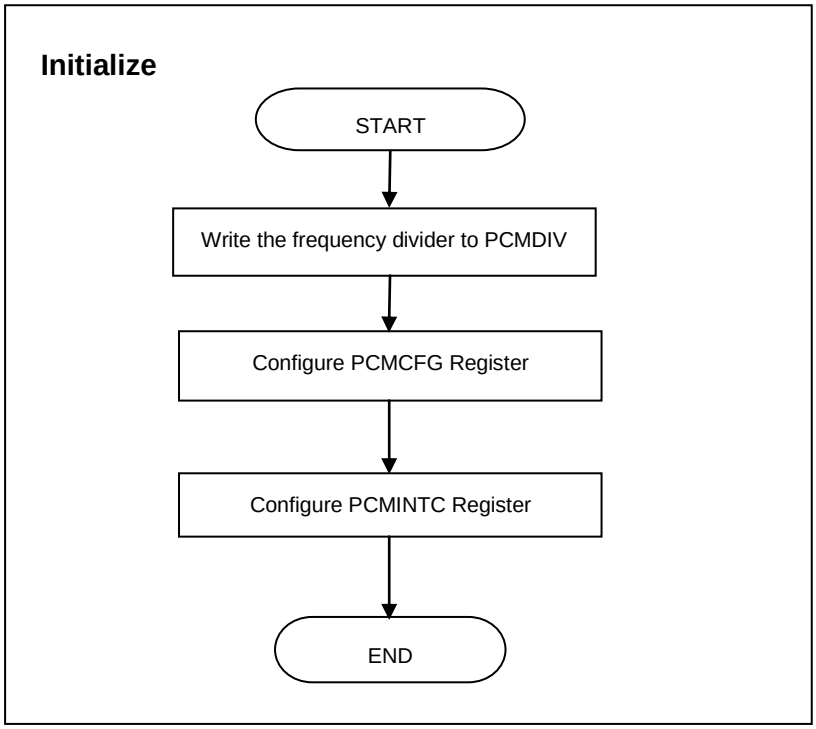
Figure 11-5 Multi-Slot Frame SYN Timing (Shown with two Slots and 8bit Sample)

11.6 PCM Operation

11.6.1 PCM Initialization

At power-on or other hardware reset (WDT and etc), PCM is disabled. Software must initiate PCM after power-on or reset.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



For further details, see the corresponding register description sections.

11.6.2 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.
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 11.6.4 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

11.6.3 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 11.6.4 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.

11.6.4 FIFOs operation

PCM has two FIFOs, one for transmitting and one for receiving. The FIFOs are in 16 bits width and 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. Figure 6 illustrates the FIFOs access.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

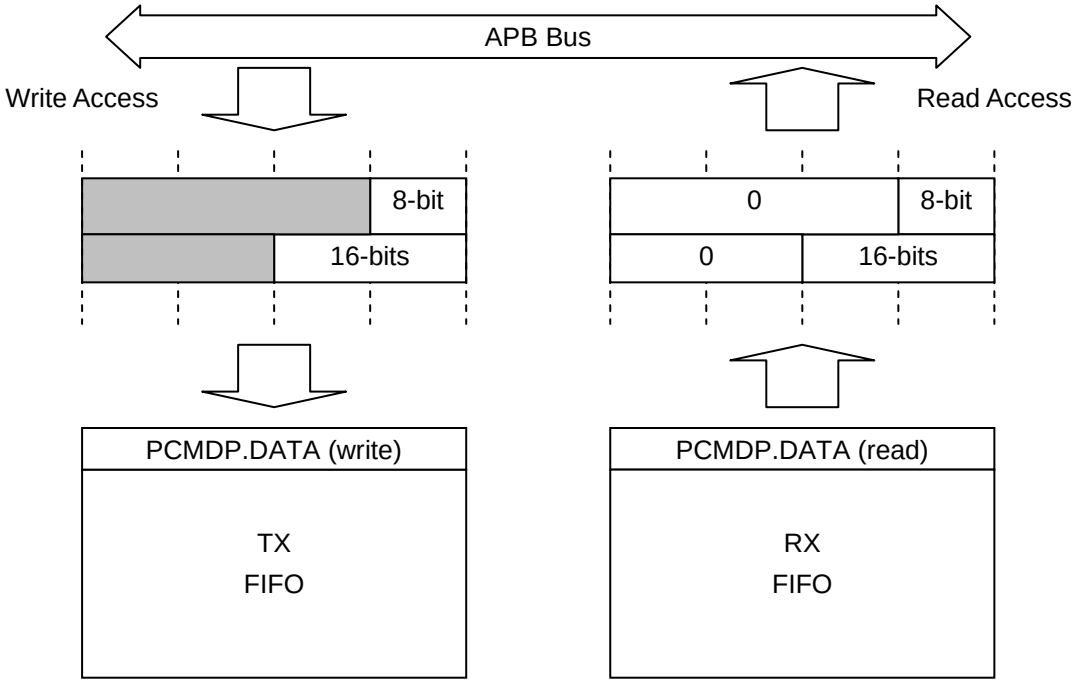


Figure 11-6 Transmitting/Receiving FIFO access via APB Bus

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.

11.6.5 Data Flow Control

There are three approaches provided to control/synchronize the incoming/outgoing data flow.

11.6.5.1 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.

Polling approach is in very low efficiency and is not recommended.

11.6.5.2 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

FIFO, and processor can store or load samples to or from the FIFO.

Interrupt approach is more efficient than polling approach.

11.6.5.3 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.

11.6.6 PCM Serial Clocks and Sampling Frequencies

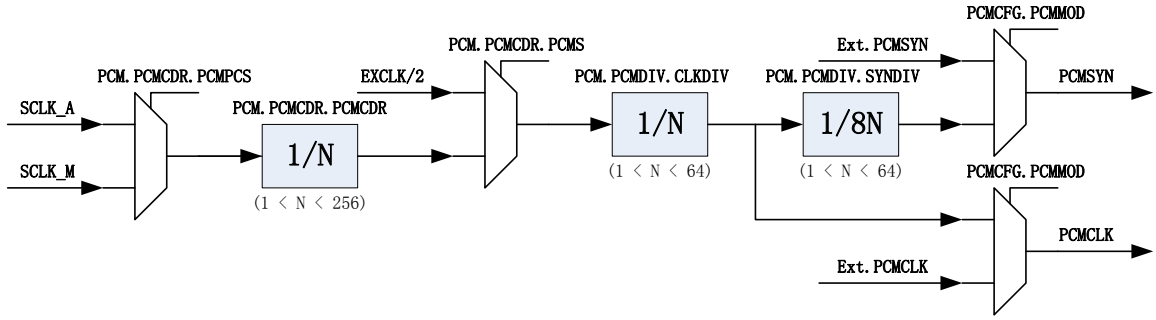


Figure 11-7 PCMCLK and PCMSYN generation scheme

Note: SCLK_A and SCLK_M please refer to CPM.

11.6.7 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).
- Receive Over-Run (PCMINTS.ROR).

For further details, see the corresponding register description sections.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12 Internal CODEC Interface

12.1 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.

12.1.1 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: 95dB A-Weighted, THD: -80dB @FS-1dB
- Line input to ADC path: SNR: 95dB 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
- Support Capacitor-coupled and Capacitor-less mode headphone connection

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

12.1.2 Signal Descriptions

CODEC has maximum 15 analog signal IO pins and 4 power pins on chip. They are listed and described in the following table.

Table 12-1 CODEC signal IO pin description

Pin Names	IO	Pin Description	Power
MICP1	AI	Microphone mono differential analog input 1 (MIC1), positive pin.	AVDCDC
MICN1	AI	Microphone mono differential analog input 1 (MIC1), negative pin.	AVDCDC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

MICP2	AI	Microphone mono differential analog input 2 (MIC2), positive pin.	AVDCDC
MICN2	AI	Microphone mono differential analog input 2 (MIC2), negative pin.	AVDCDC
MICBIAS	AO	Microphone bias.	AVDCDC
AIL	AI	Left line single-ended analog input.	AVDCDC
AIR	AI	Right line single-ended analog input.	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.	AVDHP
AOHPR	AO	Right headphone single-ended analog output.	AVDHP
AOHPM	AO	Headphone common mode output.	AVDHP
AOHPMS	AI	Headphone common mode sense input.	AVDHP
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
AVDHP	P	Headphone amplifier power, 2.5V.	-
AVSHP	P	Headphone amplifier ground.	-
AVDCDC	P	CODEC analog power, 2.5V, inter signal VREFP.	-
AVSCDC	P	CODEC analog ground, inter signal VREFN.	-
HPSENSE	AI	Headphone jack sense.	AVDHP
DMIC_IN	DI	Digital microphone data input pin.	AVDCDC
DMIC_CLK	DO	Digital microphone clock output pin.	AVDCDC

NOTES:

1 AVDHP = 2.5v (typ). AVDCDC= 2.5v (typ).

2 Inter signal VREFP is connected to AVDCDC, inter signal VREFN 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.1.3 Block Diagram

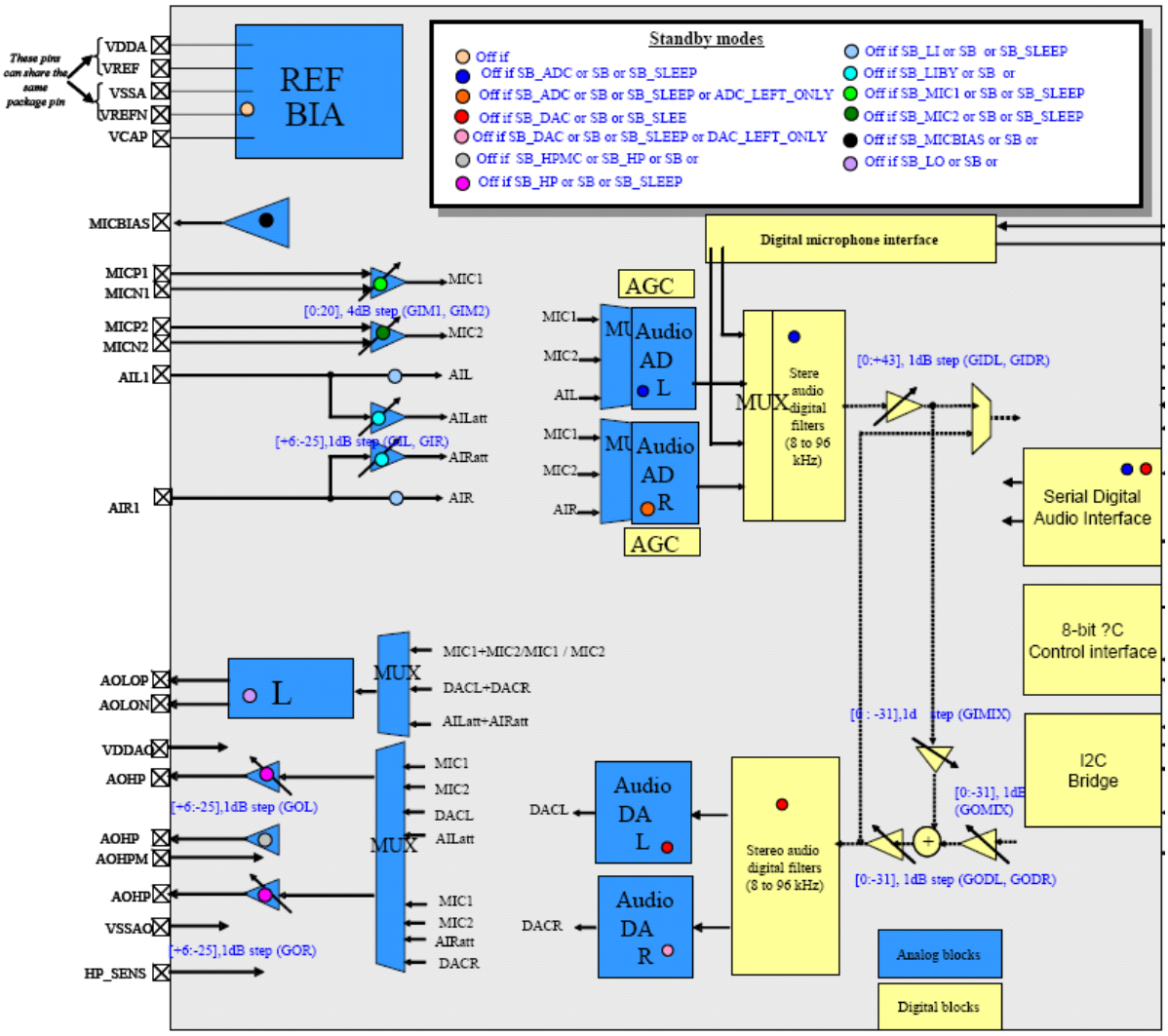


Figure 12-1 CODEC block diagram

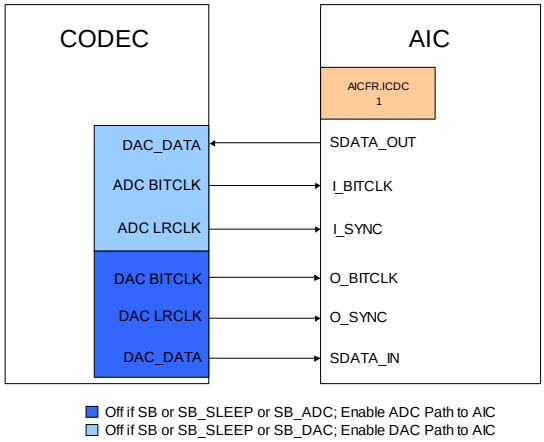


Figure 12-2 Internal CODEC works with AIC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.2 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 12-2 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.

12.2.1 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
	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
31:17	Reserved	Writing has no effect, read as zero.	R
16	RGWR	Writing 1 to this bit means issuing a write command to CODEC's	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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, 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.

12.2.2 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
	Reserved																							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.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		(RGADW.RGADDR field specifies the CODEC register's address)	

NOTE: AIC needs SYS_CLK (please refer to AIC spec), when write new value to or read from CODEC
internal registers.

12.3 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.

12.3.1 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.

12.3.2 CODEC controlling and typical operations

This section is about some typical operations. We assume the power supply of CODEC is on, and

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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:
  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.

12.3.3 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".

12.3.4 Pop noise and the reduction of it

Please refer to "Ramping system note" and "Anti-pop operation sequences" for details.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.3.4.1 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.

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).

12.4 Timing parameters

Parameter	Condition	Min.	Typ.	Max.	Unit
Tsbyu	Cext = 10uF/100nF +/-20%		250	500	ms

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

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.

12.5 AC & DC parameters

Voltages:

AVSHP and AVSCDC are connected to analog ground.

AVDHP = 2.5V (typ).

AVDCDC= 2.5V (typ).

power consumption

Mode	Analog	Digital	Unit
Sleep mode	650	250	uW
Playback stereo audio DAC only (capacitor coupled load configuration)	4.5	1.15	mW
Record line input only (audio ADC)	4.5	0.95	mW
Record mic stereo input only (audio ADC)	5.4	0.95	mW
Record mic mono input only (audio ADC)	3.2	0.95	mW
Bypass path (capacitor coupled load configuration)	4.4	0.25	mW

Current value is at AVDCDC = AVDHP = 2.5 V.

Chip pin Name	MAX Current across I/O @ AVDCDC = AVDHP = 2.5 V
AVDCDC	< 20 mA in normal working mode
AVSCDC	< 20 mA in normal working mode
AVDHP	< 160 mA in normal working mode
	< 1400 mA in case of short circuit
AVSHP	< 160 mA in normal working mode
	< 1400 mA in case of short circuit
VCAP	< 2 mA in normal working mode
MICP1, MICN1	< 2 mA in normal working mode
MICP2, MICN2	< 2 mA in normal working mode

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

MICBIAS	< 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
AOHPM	< 80 mA in normal working mode
	< 1200 mA in case of short circuit
AOHPMS	< 1 mA in normal working mode
AIL, AIR	< 1 mA in normal working mode
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.

12.6 CODEC internal Registers

Register Name	Function	Address	Reset value
SR	Status Register	000000 / 0x0 / 00	h00
AICR_DAC	DAC Audio Interface Control Register	000001 / 0x1 / 01	hC3
AICR_ADC	ADC Audio Interface Control Register	000010 / 0x2 / 02	hC3
CR_LO	differential line-out Control Register	000011 / 0x3 / 03	h90
CR_HP	HeadPhone Control Register	000100 / 0x4 / 04	h98
CR_DAC	DAC Control Register	000110 / 0x6 / 06	h90
CR_MIC	Microphone Control Register	000111 / 0x7 / 07	hB1
CR_LI	Control Register for line inputs	001000 / 0x8 / 08	h11
CR_ADC	ADC Control Register	001001 / 0x9 / 09	h10
CR_MIX	Control Register for digital mixer	001010 / 0xA / 10	h00
CR_VIC	Control Register for the codec	001011 / 0xB / 11	h03
CCR	Clock Control Register	001100 / 0xC / 12	h00
FCR_DAC	DAC Frequency Control Register	001101 / 0xD / 13	h00
FCR_ADC	ADC Frequency Control Register	001110 / 0xE /14	h40
ICR	Interrupt Control Register	001111 / 0xF / 15	h00
IMR	Interrupt Mask Register	010000 / 0x10 / 16	hFF
IFR	Interrupt Flag Register	010001 / 0x11 / 17	h00
GCR_HPL	left channel headphone Control Gain Register	010010 / 0x12 / 18	h06
GCR_HPR	right channel headphone Control Gain Register	010011 / 0x13 / 19	h06
GCR_LIBYL	left channel bypass line Control Gain	010100 / 0x14 / 20	h06

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Register		
GCR_LIBYR	right channel bypass line Control Gain Register	010101 / 0x15 / 21	h06
GCR_DACL	Left channel DAC Gain Control Register	010110 / 0x16 / 22	h00
GCR_DACR	right channel DAC Gain Control Register	010111 / 0x17 / 23	h00
GCR_MIC1	Microphone 1 Gain Control Register	011000 / 0x18 / 24	h00
GCR_MIC2	Microphone 2 Gain Control Register	011001 / 0x19 / 25	h00
GCR_ADCL	Left ADC Gain Control Register	011010 / 0x1A / 26	h00
GCR_ADCR	Right ADC Gain Control Register	011011 / 0x1B / 27	h00
GCR_MIXADC	ADC Digital Mixer Control Register	011101 / 0x1D / 29	h00
GCR_MIXDAC	DAC Digital Mixer Control Register	011110 / 0x1E / 30	h00
AGC1	Automatic Gain Control 1	011111 / 0x1F / 31	h34
AGC2	Automatic Gain Control 2	100000 / 0x20 / 32	h07
AGC3	Automatic Gain Control 3	100001 / 0x21 / 33	h44
AGC4	Automatic Gain Control 4	100010 / 0x22 / 34	h1F
AGC5	Automatic Gain Control 5	100011 / 0x23 / 35	h00

12.6.1 CODEC internal registers

12.6.1.1 SR: Status Register

Register Name: SR				Register Address: 0x0			
bit7-R-0	bit6-R-0	bit5-R-0	bit4-R-0	Bit3-R-0	bit2-R-0	bit1-R-0	bit0-R-0
PON_ACK	IRQ_ACK	JACK	Reserved				

Bits	Field	Description
7	PON_ACK	Acknowledge status bit after power on. Read 0: reset value 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
4:0	Reserved	Writing has no effect, read as zero.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.6.1.2 AICR_DAC: Audio Interface Control Register

Register Name: AICR_DAC				Register Address: 0x1			
bit7-RW-1	bit6-RW-1	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-1	bit0-RW-1
DAC_ADWL		Reserved				DAC_SERIAL	DAC_I2S

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
5:2	Reserved	Writing has no effect, read as zero.
1	DAC_SERIAL	Selection of DAC digital serial audio interface. Read / Write 0: Parallel interface 1: Serial interface
0	DAC_I2S	Working mode of DAC serial mode. (only relevant when serial interface is selected) Read/Write 0: DSP mode 1: I2S mode

NOTES:

1 DAC_SERIAL should be configured to 1.

2 DAC_I2S should be configured to 1.

12.6.1.3 AICR_ADC: Audio Interface Control Register

Register Name: AICR_ADC				Register Address: 0x2			
bit7-RW-1	bit6-RW-1	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-1	bit0-RW-1
ADC_ADWL		Reserved				ADC_SERIAL	ADC_I2S

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		11: 24-bit word length data
5:2	Reserved	Writing has no effect, read as zero.
1	ADC_SERIAL	Selection of the ADC digital serial audio interface. Read / Write 0: Parallel interface 1: Serial interface
0	ADC_I2S	Working mode of the ADC digital serial audio interface. (only relevant when serial interface is selected) Read/Write 0: DSP mode 1: I2S mode

NOTES:

1 ADC_SERIAL should be configured to 1.

2 ADC_I2S should be configured to 1.

12.6.1.4 CR_LO: differential line-out Control Register

Register Name: CR_LO				Register Address: 0x3			
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 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-out Amplifier input selection. Read/Write If MICSTEREO = 0 00 : Microphone 1 enabled 01 : Microphone 2 enabled 10 : Bypass path enabled 11 : DAC output enabled

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		If MICSTEREO = 1 00 : Microphone 1 & 2 enabled 01 : Microphone 1 & 2 enabled 10 : Bypass path enabled 11 : DAC output enabled

12.6.1.5 CR_HP: HeadPhone Control Register

Register Name: CR_HP				Register Address: 0x4			
bit7-RW-1	Bit6-RW-0	Bit5-RW-0	Bit4-RW-1	bit3-RW-1	bit2-RW-0	bit1-RW-0	bit0-RW-0
HP_MUTE	LOAD	Reserved	SB_HP	SB_HPCM	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	LOAD	Selection of load impedance value for ramp generation. Read/Write 0: 16 Ohm / 220 uF 1: 10 kOhm / 1 uF
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	SB_HPCM	headphone output stage common mode buffer power-down mode. Read/Write 0: active (capacitor less headphone output configuration) 1: power-down (line output configuration)
2	Reserved	Writing has no effect, read as zero.
1:0	HP_SEL	Headphone Output Amplifier input selection. Read/Write If MICSTEREO = 0 00: Microphone 1 input to left and right channels 01: Microphone 2 input to left and right channels 10: Bypass path enabled 11: DAC output enabled If MICSTEREO = 1 00: Microphone 1 input to left channel and microphone 2 input to right channel 01: Microphone 2 input to left channel and microphone 1 input

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		to right channel 10: Bypass path enabled 11: DAC output enabled

NOTE: The LOAD register is the load of AOHP.

12.6.1.6 CR_DAC: Control Register for DAC 3

Register Name: CR_DAC				Register Address: 0x6			
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	DAC_MONO	DAC_LEFT_ ONLY	SB_DAC	DAC_ LRSWAP	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	DAC_MONO	Digital stereo-to-mono conversion for DAC path. Read/Write 0: stereo 1: mono When DAC_MONO=1, the left and right channels are mixed in digital part, the mixed result is emitted on both left and right channel of DAC output. The result corresponds to the average of left and right channels. When DAC_MONO=0, left and right channels are emitted to their corresponding channel.
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	DAC_LRSWAP	swap between Left and right channels. Read/Write 0: left data are sent to right channel, right data to left channel (swap) 1: left data are sent to left channel, right data to left channel (do not

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		swap)
2:0	Reserved	Writing has no effect, read as zero.

NOTE: DAC_LRSWAP should be configured to 1.

12.6.1.7 CR_MIC: Control Register for microphone inputs

Register Name: CR_MIC				Register Address: 0x7			
bit7-RW-1	bit6-RW-0	bit5-RW-1	Bit4-RW-1	bit3-RW-0	bit2-RW-0	Bit1-RW-0	bit0-RW-1
MIC_STEREO	MICIDFF	SB_MIC2	SB_MIC1	Reserved		MICBIAS_V0	SB_MICBIAS

Bits	Field	Description
7	MIC_STEREO	Microphone input mode selection. Read/Write 0: Microphone mono inputs 1: Microphone stereo inputs This signal affects IN_SEL, HP_SEL, LO_SEL. Refer to their descriptions.
6	MICIDFF	Microphone input mode selection. Read/Write 0:Microphone single-ended inputs 1: Microphone differential inputs
5	SB_MIC2	Analog MIC2 Input conditioning circuitry power-down mode. Read/Write 0: active 1: power-down
4	SB_MIC1	Analog MIC1 Input conditioning circuitry power-down mode. Read/Write 0: active 1: power-down
3:2	Reserved	Writing has no effect, read as zero.
1	MICBIAS_V0	B-port MICBIAS stage output voltage in operating mode. Read/Write 0: 5/6*VREF output voltage 1: 4/6*VREF output voltage
0	SB_MICBIAS	Microphone biasing buffer power-down. Read/Write 0: active 1: power-down

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.6.1.8 CR_LI: Control Register for line inputs

Register Name: CR_LI				Register Address: 0x8			
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			SB_LIBY	Reserved			SB_LIN

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4	SB_LIBY	Linein used for bypass path power-down. Read/Write 0: active 1: power-down
3:1	Reserved	Writing has no effect, read as zero.
0	SB_LIN	Linein used for ADC power-down. Read/Write 0: active 1: power-down

12.6.1.9 CR_ADC: Control Register for ADC

Register Name: CR_ADC				Register Address: 0x9			
bit7-RW-0	bit6-RW-0	bit5-RW-0	Bit4-RW-1	bit3-RW-0	bit2-RW-0	Bit1-RW-0	bit0-RW-0
DMIC_SEL	ADC_MONO	ADC_LEFT_ ONLY	SB_ADC	ADC_LRSWAP	Reserved	IN_SEL	

Bits	Field	Description
7	DMIC_SEL	digital filter input selection. Read/Write 0: ADC 1: Digital microphone
6	ADC_MONO	Digital stereo-to-mono conversion for ADC path. Read/Write 0: stereo 1: mono When ADC_MONO=1, the left and right channels are mixed in digital part: the mixed result is emitted on both left and right channel of ADC digital output. The result corresponds to the average of left and right channels. When ADC_MONO=0, left and right channels are emitted to their corresponding channel.
5	ADC_LEFT_ONLY	Deactivation of ADC right channel.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Read/Write 0: ADC right channel active 1: ADC right channel inactive Note that when ADC right channel is deactivated, left channel is emitted on both left and right channel of ADC digital output.
4	SB_ADC	ADC power down mode. Read/Write 0: active 1: power-down
3	ADC_LRSWAP	swap between Left an d right channels. Read/Write 0: left data are sent to right channel, right data to left channel (swap) 1: left data are sent to left channel, right data to right channel (do not swap)
2	Reserved	Writing has no effect, read as zero.
1:0	IN_SEL	selection of the signal converted by the ADC. Read/Write If MICSTEREO = 0 00: Microphone 1 input to left and right channels (codec automatically considers that ADC_LEFT_ONLY equals '1' to optimize power consumption) 01: Microphone 2 input to left and right channels (codec automatically considers that ADC_LEFT_ONLY equals '1' to optimize power consumption) 10 : Line input 11 : Reserved for further use If MICSTEREO = 1 00: Microphone 1 input to left channel and microphone 2 input to right channel 01: Microphone 2 input to left channel and microphone 1 input to right channel 10 : Line input 11 : Reserved for test

NOTE: ADC_LRSWAP should be configured to 1.

12.6.1.10 CR_MIX: Control Register for digital mixer

Register Name: CR_MIX				Register Address: 0xA			
bit7-RW-1	bit6-RW-0	bit5-RW-0	Bit4-RW-0	bit3-RW-0	bit2-RW-0	Bit1-RW-0	bit0-RW-0
Reserved				MIX_REC		DAC_MIX	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Field	Description
7:4	Reserved	Writing has no effect, read as zero.
3:2	MIX_REC	Mixer mode on ADC Path. Read/Write 00: Record input only 01: Record input + DAC 10: Reserved for further use 11: Reserved for further use
1:0	DAC_MIX	Mixer mode on DAC Path. Read/Write 00: Playback DAC only 01: Playback DAC + ADC 10: Reserved for further use 11: Reserved for further use

12.6.1.11 CR_VIC: Control Register for the codec

Register Name: CR_VIC				Register Address: 0xB			
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

12.6.1.12 CCR: Control Clock Register

Register Name: CCR				Register Address: 0xC			
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			CRYSTAL			

Bits	Field	Description
7	DMIC_CLKON	Digital microphone clock (DMIC_CLK) enable. Read/Write

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: clock off 1: clock on, clock frequency varies with DMIC_RATE and MCLK MCLK / CRYSTAL / DMIC_CLK frequency 12 MHz / 0000 / 3 MHz 13 MHz / 0001 / 3.25 MHz
6:4	Reserved	Writing has no effect, read as zero.
3:0	CRYSTAL	Selection of the SYS_CLK frequency. Read/Write The sampling frequency value is given in the CRYSTAL table. CRYSTAL / Master Clock Frequency 0000 / 12 MHz 0001 / 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).

12.6.1.13 FCR_DAC: DAC Frequency Control Register

Register Name: FCR_DAC				Register Address: 0xD			
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.

12.6.1.14 FCR_ADC: ADC Frequency Control Register

Register Name: FCR_ADC				Register Address: 0xE			
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	Reserved		ADC_FREQ			

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6	ADC_HPF	ADC High Pass Filter enable. Read/Write 0: inactive 1: enable the ADC High Pass Filter
5:4	Reserved	Writing has no effect, read as zero.
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.

12.6.1.15 ICR: Interrupt Control Register

Register Name: ICR				Register Address: 0xF			
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 SYS_CLK cycles duration 11: The generated IRQ is a low level pulse with an 8 SYS_CLK cycles duration
5:0	Reserved	Writing has no effect, read as zero.

NOTE: The SYS_CLK frequency refers to 'CRYSTAL' of 'Control Clock Register'.

12.6.1.16 IMR: Interrupt Mask Register

Register Name: IMR				Register Address: 0x10			
bit7-RW-1	bit6-RW-1	Bit5-RW-1	bit4-RW-1	bit3-RW-1	bit2-RW-1	bit1-RW-1	bit0-RW-1
Reserved	SCLR_MASK	JACK_MASK	SCMC_MASK	RUP_MASK	RDO_MASK	GUP_MASK	GDO_MASK

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	SCLR_MASK	Mask for the SCLR flag. Read/Write 0: interrupt enabled

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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	SCMC_MASK	Mask for the SCMC flag. 0: interrupt enabled 1: interrupt masked (no IRQ generation)
3	RUP_MASK	Mask for the RUP flag. 0: interrupt enabled 1: interrupt masked (no IRQ generation)
2	RDO_MASK	Mask for the RDO flag. 0: interrupt enabled 1: interrupt masked (no IRQ generation)
1	GUP_MASK	Mask for the GUP flag. 0: interrupt enabled 1: interrupt masked (no IRQ generation)
0	GDO_MASK	Mask for the GDO flag. 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 & (!IMR) 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 or until a masked event is unmasked.

4 SYS_CLK must not be stopped in order to propagate IRQ signal.

12.6.1.17 IFR: Interrupt Flag Register

Register Name: IFR				Register Address: 0x11			
bit7-RW-0	Bit6-R-0	bit5-RW-0	bit4-R-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
Reserved	SCLR	JACK_EVENT	SCMC	RUP	RDO	GUP	GDO

Bits	Field	Description
7	Reserved	Writing has no effect, read as zero.
6	SCLR	Left or Right Output short circuit detection status. Read 0 : no event 1 : event detected (due to JACK flag change to '0' or '1') Write 1 to Reset of the flag.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 1 to Reset of the flag.
4	SCMC	Common mode buffer output short circuit detection status. Read 0: inactive 1: indicates that a short circuit has been detected by the output stage Write 1 to Update of the flag.
3	RUP	End of output stage ramp up flag. Read 0: the ramp-up sequence does not occur or is not completed 1: the ramp-up sequence is completed (output stage is active). Write 1 to Reset of the flag.
2	RDO	End of output stage ramp down flag. Read 0: the ramp-down sequence does not occur or is not completed 1: the ramp-down sequence is completed (output stage in stand-by mode) Write 1 to Reset of the flag.
1	GUP	End of mute gain up sequence flag. Read 0: the mute gain up sequence does not occur or is not completed 1: the mute gain up sequence is completed; the DAC input signal is transmitted to the DAC path Write 1 to Reset of the flag.
0	GDO	End of mute gain down sequence flag. Read 0: the mute gain down sequence does not occur or is not completed 1: the mute gain down sequence is completed, a 0V DC signal is transmitted to the DAC path Write 1 to 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.

12.6.1.18 GCR_HPL: left channel headphone Gain Control Register

Register Name: GCR_HPL				Register Address: 0x12			
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				

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

12.6.1.19 GCR_HPR: right channel headphone Gain Control Register

Register Name: GCR_HPR				Register Address: 0x13			
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.

12.6.1.20 GCR_LIBYL: left channel bypass line Gain Control Register

Register Name: GCR_LIBYL				Register Address: 0x14			
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.

NOTE: Please refer to section "Programmable Bypass path attenuation: GI" for more details.

12.6.1.21 GCR_LIBYR: right channel bypass line Gain Control Register

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Register Name: GCR_LIBYR				Register Address: 0x15			
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.

12.6.1.22 GCR_DACL: Left channel DAC Gain Control Register

Register Name: GCR_LIBYL				Register Address: 0x16			
bit7-RW-0	Bit6-RW-0	Bit5-RW-0	Bit4-RW-0	bit3-RW-0	bit2-RW-0	bit1-RW-0	bit0-RW-0
RLGOD	HCP_DIS	GODL					

Bits	Field	Description
7	RLGOD	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	HCP_DIS	Hard Clipping Preventer disable. Read/Write 0: Hard Clipping Preventer is enabled 1: Hard Clipping Preventer is disabled.
5:0	GODL	Left channel DAC digital gain programming value.

NOTE: Please refer to section "Programmable digital attenuation: GOD" for more details.

Hard Clipping Preventer function is described in section "Hard Clipping Preventer".

12.6.1.23 GCR_DACR: right channel DAC Gain Control Register

Register Name: GCR_DACR				Register Address: 0x17			
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
7:6	Reserved	Writing has no effect, read as zero.
5:0	GODR	Right channel DAC digital gain programming value.

NOTE: Please refer to section "Programmable digital attenuation: GOD" for more details.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.6.1.24 GCR_MIC1: Microphone 1 Gain Control Register

Register Name: GCR_MIC1				Register Address: 0x18			
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".

12.6.1.25 GCR_MIC2: Microphone 2 Gain Control Register

Register Name: GCR_MIC2				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
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".

12.6.1.26 GCR_ADCL: Left channel ADC Gain Control Register

Register Name: GCR_ADCL				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
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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.6.1.27 GCR_ADCR: Right channel ADC Gain Control Register

Register Name: GCR_ADCR				Register Address: 0x1B			
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".

12.6.1.28 GCR_MIXADC: ADC Digital Mixer Control Register

Register Name: GCR_MIXADC				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			GIMIX				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GIMIX	Mixer gain for input path. Read/Write 00000 : 0dB 00001 : -1dB … by step of 1dB 11111 : -31dB

12.6.1.29 GCR_MIXDAC: DAC Digital Mixer Control Register

Register Name: GCR_MIXDAC				Register Address: 0x1E			
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			GOMIX				

Bits	Field	Description
7:5	Reserved	Writing has no effect, read as zero.
4:0	GOMIX	Mixer gain for DAC path. Read/Write 00000 : 0dB 00001 : -1dB

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		…by step of 1dB 11111 : -31dB

12.6.1.30 AGC1: Automatic Gain Control Register 1

Register Name: AGC1				Register Address: 0x1F			
bit7-RW-0	bit6-RW-0	bit5-RW-1	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-0	bit0-RW-0
AGC_EN	AGC_STEREO	TARGET				Reserved	

Bits	Field	Description
7	AGC_EN	Enable of the AGC system. Read/Write 0 : inactive 1 : enable the automatic level control
6	AGC_STEREO	Select the same or different gains between left and right channels of the AGC system. Read/Write 0 : the same gain applied to Left and Right channel 1 : different gains applied to Left and Right channel
5:2	TARGET	Target output level of the ADC. Read/Write: 0000 : -6dB 0001 : -7.5dB …by step of 1.5 dB 1111 : - 28.5dB
1:0	Reserved	Writing has no effect, read as zero.

NOTE: Please refer to section "AGC system guide" for more details.

12.6.1.31 AGC2: Automatic Gain Control Register 2

Register Name: AGC2				Register Address: 0x20			
bit7-RW-0	bit6-RW-1	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	bit1-RW-1	bit0-RW-1
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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Input level (dB) < Noise Gate Level (dB). Read/Write 000: -72 dB 001: -66 dB …by step of 6dB 111: -30 dB
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.

12.6.1.32 AGC3: Automatic Gain Control Register 3

Register Name: AGC3				Register Address: 0x21			
bit7-RW-0	bit6-RW-1	bit5-RW-0	bit4-RW-0	bit3-RW-0	bit2-RW-1	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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

12.6.1.33 AGC4: Automatic Gain Control Register 4

Register Name: AGC4				Register Address: 0x22			
Bit7-RW-0	bit6-RW-0	bit5-RW-0	bit4-RW-1	bit3-RW-1	bit2-RW-1	bit1-RW-1	bit0-RW-1
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.

NOTES:

1 Please refer below table for AGC_MAX setup.

2 Please refer to section "AGC system guide" for more details.

12.6.1.34 AGC5: Automatic Gain Control Register 5

Register Name: AGC5				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
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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

12.7 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).

12.7.1 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	Gain value (dB)	Maximum input amplitude
000	0	0.85*VREF
001	4	0.536*VREF
010	8	0.338*VREF
011	12	0.213*VREF
100	16	0.134*VREF
101	20	0.085*VREF
110	20	0.085*VREF
111	20	0.085*VREF

NOTES:

1 Analog input amplitude value is not more than ‘Maximum input amplitude’ for Vpp differential.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

12.7.2 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:

GID	Decimal decoded	Gain (dB)	Maximum input amplitude (Vpp. 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	0.06 * VREF
1 1 1 1 1 1	63	43	0.06 * 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 Vpp differential. These values refer to the external voltage reference VREF equal to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

12.7.3 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 +32 to –31dB with 1 dB pitch. The gain and output levels are obtained according to the following table:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

GOD						Decimal decoded value	Gain Value (dB)
0	1	1	1	1	1	31	-31
0	1	1	1	1	0	30	-30
						…	
0	0	0	1	1	0	6	-6
						…	
0	0	0	0	0	1	1	-1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	-1	1
1	1	1	1	1	0	-2	2
1	1	1	0	1	0	-6	6
1	0	0	0	0	1	-31	31
1	0	0	0	0	0	-32	32

12.7.4 Programmable attenuation: GO

The attenuation of Headphone output amplifier may be programmed independently for the both 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 (Vpp)	Maximal PGAT output amplitude (Vpp)
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:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 Vpp single-ended.

3 These values refer to the external voltage reference VREF equal to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

12.7.5 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 (Vpp) (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
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 Vpp. These values refer to the external voltage reference VREF equal to (VREFP – VREFN). The voltage levels depend on the VREF voltage.

12.7.6 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

00011	-3
…	…
11101	-29
11110	-30
11111	-31

12.7.7 Hard Clipping Preventer

The Hard Clipping Preventer is a digital signal soft limiter. Its goal is to limit digital signal when amplification gain is applied in the digital domain (GOD* or GOMIX* > 0dB), to reduce occurrence of possible hard clipping event when input signal [1] reach fullscale [2]. This function is enabled by setting the bit HCP_DIS to '0' and disable by setting '1' (GCR_DACL: Left channel DAC Gain Control Register). When enabled, signals at the input of DAC noiseshaper is soft clipped when they are larger than a threshold of full scale minus 1dB.

The part above the threshold is divided by 8; thus, the signal above threshold at the output of the Hard Clipping Preventer is:

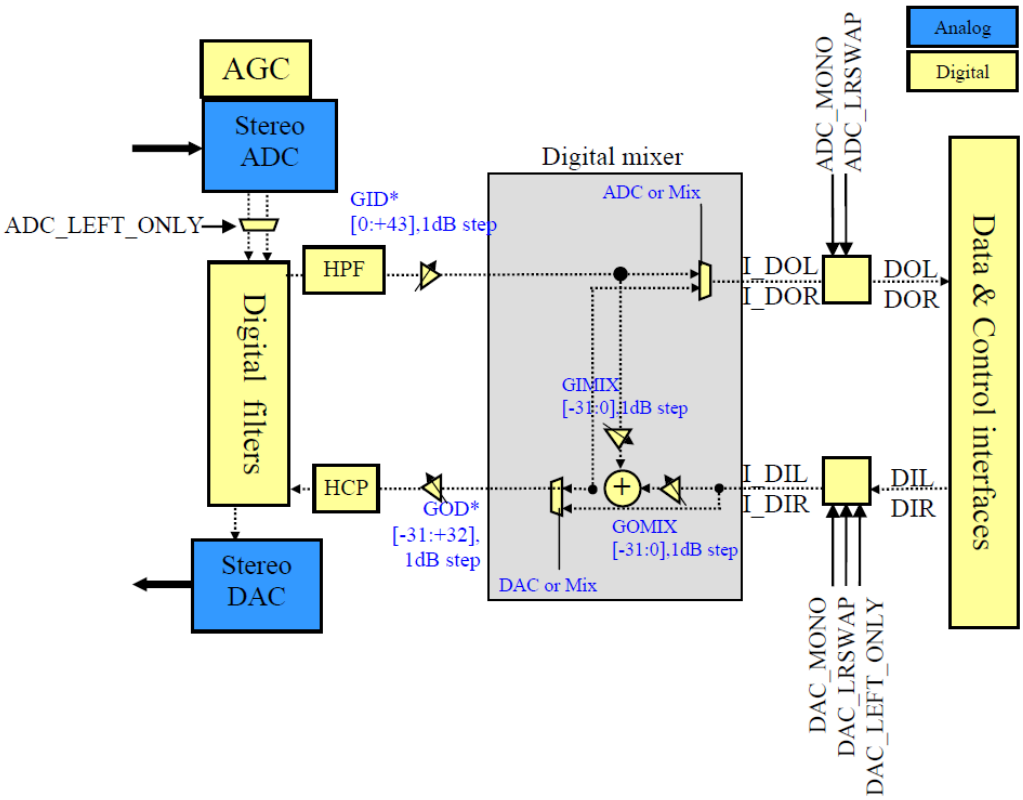
$$HCP_{output} = \frac{(DAC_{input} \times GOMIX \times GOD) - Threshold}{8} + Threshold$$

Note [1]: the input signal of the HCP function corresponds to the signal level at the DAC input multiply by GOD* and GOMIX*.

Note [2]: reference to full scale corresponds to the maximum signal level at the analog output, with GO* set to 0dB.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.7.8 Detailed description of the digital mixer



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 additional 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.

A digital multiplexer allows choosing between the ADC signal and the mixer output signal on the record path.

Another digital multiplexer allows choosing between the DAC signal and the mixer output signal on the playback path.

MONO, Left/Right swap, Left Only functionality are applied between the Digital Mixer and the Data interface. The behaviour is the following:

DAC_MONO	DAC_LRSWAP	DAC_LEFT_ONLY	I_DIL	I_DIR
0	0	0	DIR	DIL
0	0	1	DIR	DIR
0	1	0	DIL	DIR
0	1	1	DIL	DIL
1	x	x	(DIL+DIR)/2	(DIL+DIR)/2

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

ADC_MONO	ADC_LRSWAP	ADC_LEFT_ONLY	DOL	DOR
0	0	X	I_DOL	I_DOR
0	1	X	I_DOR	I_DOL
1	x	x	(I_DOL+I_DOR)/2	(I_DOL+I_DOR)/2

12.7.9 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

12.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.9 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.

12.10 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.

12.10.1 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’).

12.10.2 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.

12.10.3 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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)

12.11 Sampling frequency: FREQ

The sampling frequency value is given in the FREQ table below.

FREQ	Sampling Rate (Fs)
0000	96kHz
0001	48kHz
0010	44.1kHz
0011	32kHz
0100	24kHz
0101	22.05kHz
0110	16kHz
0111	12kHz
1000	11.025kHz
1001	8kHz
1010	Reserved for further use
….	….
1111	Reserved for further use

12.12 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The capability to use a data word length of 16 bits is kept for compatibility with standard applications.

12.13 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 AVDHP/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 AVDHP/2.

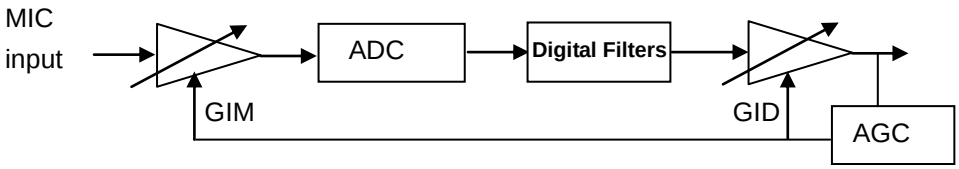
After power supplies ramp up, the CODEC starts its internal initialization sequence and SR. PON_ACK register is changed when the initialization sequence is complete.
An interrupt request is sent when the ramp completes.

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.

12.14 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.

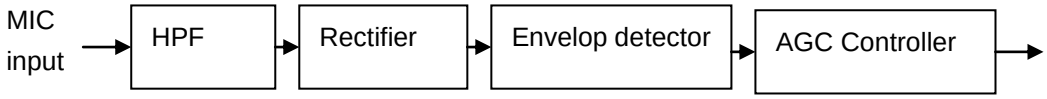


The AGC circuit compares the output of the ADC to a level and increases or decreases the gain of the microphone preamplifier and the digital gain to compensate. The full dynamic range of the ADC can be used automatically if the audio from the microphone is to be output digitally through the ADC.

The AGC_EN register bit enables the AGC system, in this case IN_SEL must be equal to "00" or "01".

AGC Block Diagram:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



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).

12.14.1 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 to the ATK register value.

The following figure illustrates the behavior of AGC system:

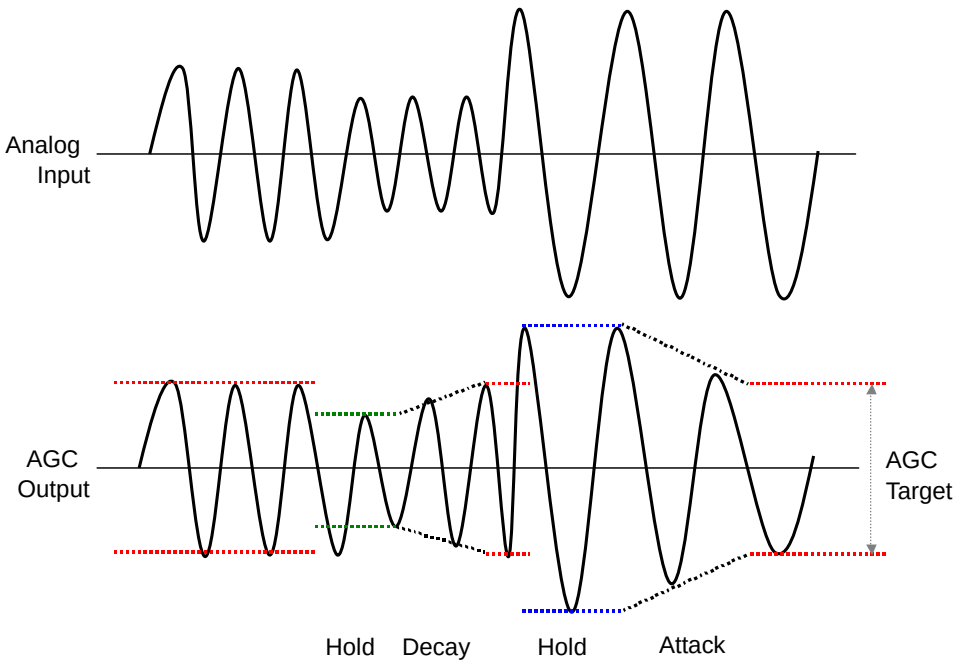


Figure 12-3 AGC adjusting waves

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

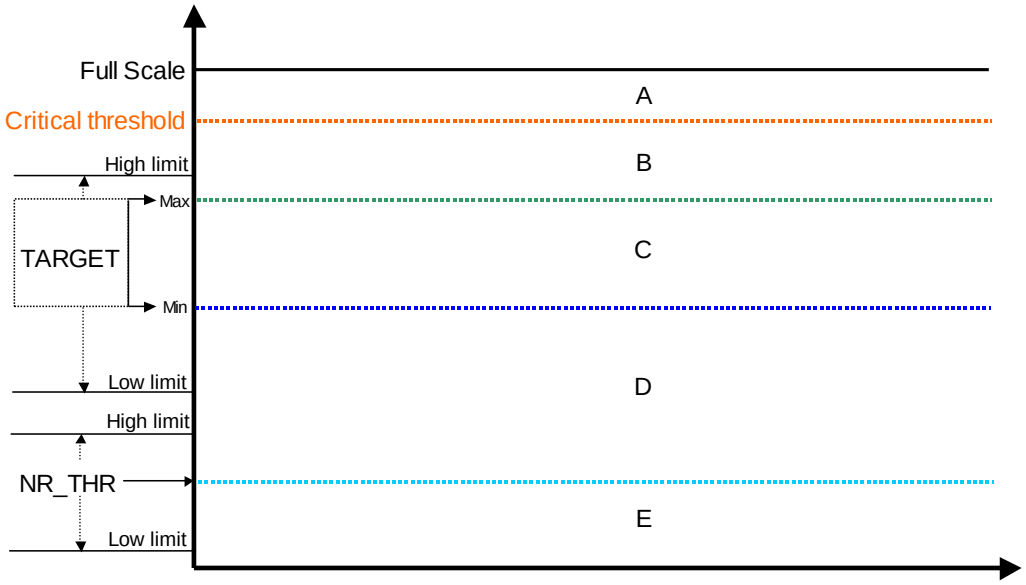


Figure 12-4 AGC adjust areas

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.

12.15 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 and GOMIX control each input of the mixer to adapt the amplitude of the mixed

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

signal. A zero-crossing detection is included on each gain stage to minimize the zipper noise.

A digital multiplexer allows choosing between the ADC signal and the mixer output signal on the record path.

Another digital multiplexer allows choosing between the DAC signal and the mixer output signal on the playback path.

Please refer to “CODEC Operating modes” for the digital mixer diagram in the “CODEC Power Diagram”.

12.16 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.

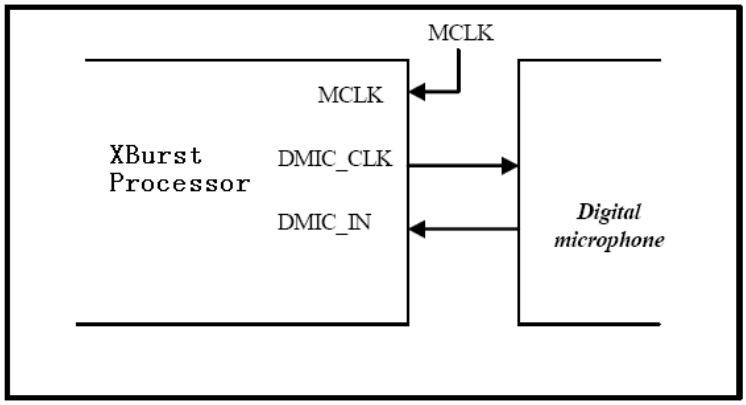


Figure 12-5 Digital microphone interface connection

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.16.1 Timing Diagram

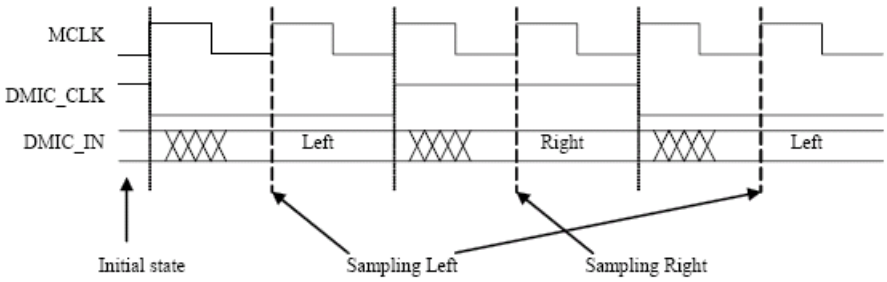
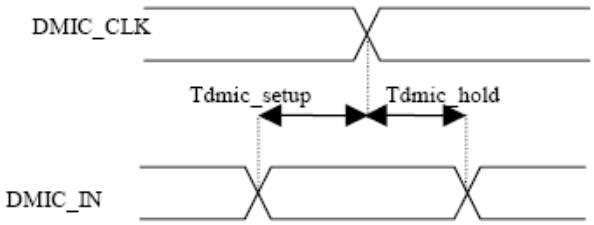


Figure 12-6 Digital microphone timing diagram at MCLK = 12 MHz

(DMIC_CLK = 3 MHz) and MCLK = 13 MHz (DMIC_CLK = 3.25 MHz)

12.16.2 Timings



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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.16.3 Noise template (TBC)

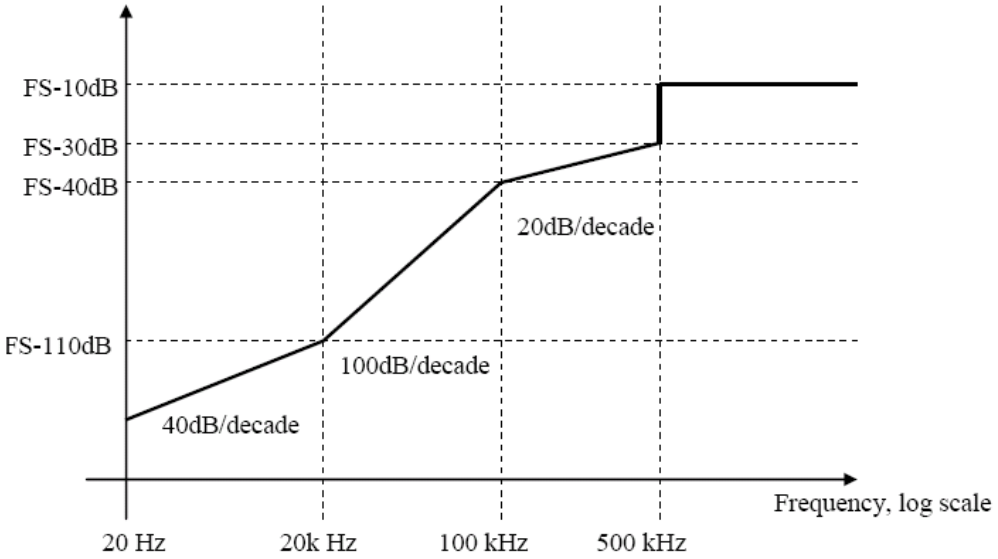


Figure 12-7 Digital microphone modulation noise reference spectrum

(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.

12.17 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Figure 12-8 CODEC Power Diagram

There are many power parts of CODEC. Any part could be powered down independently.

12.17.1 Power-On mode and Power-Off mode

When the power supply ramps up, CODEC enters the power-on mode. During the reset, the CODEC 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.

12.17.2 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

After system reset the CODEC will exit Reset mode and go to STANDBY mode.

NOTES:

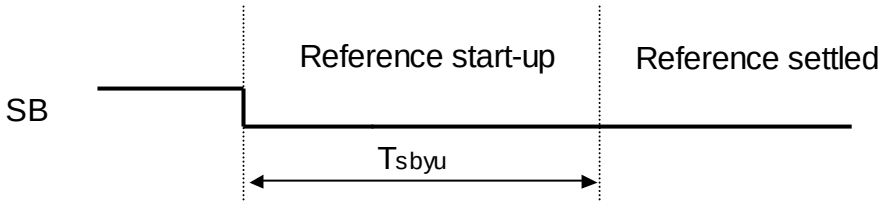
1 Except during the power-up mode, do NOT perform any reset in order to avoid audible pops.

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.

12.17.3 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.

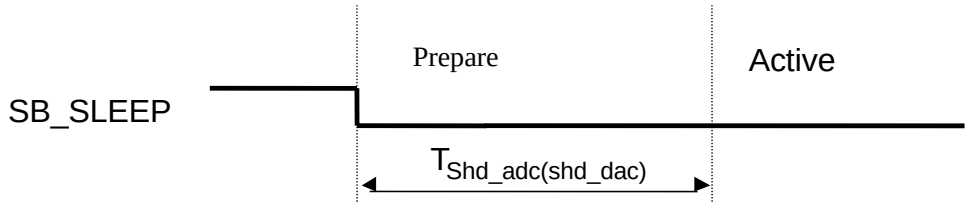


Please refer to the section "Timing parameters" for the Tsbyu Value.

12.17.4 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Please refer to the section "Timing parameters" for the Tshd_adc and Tshd_dac Value.

12.17.5 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.

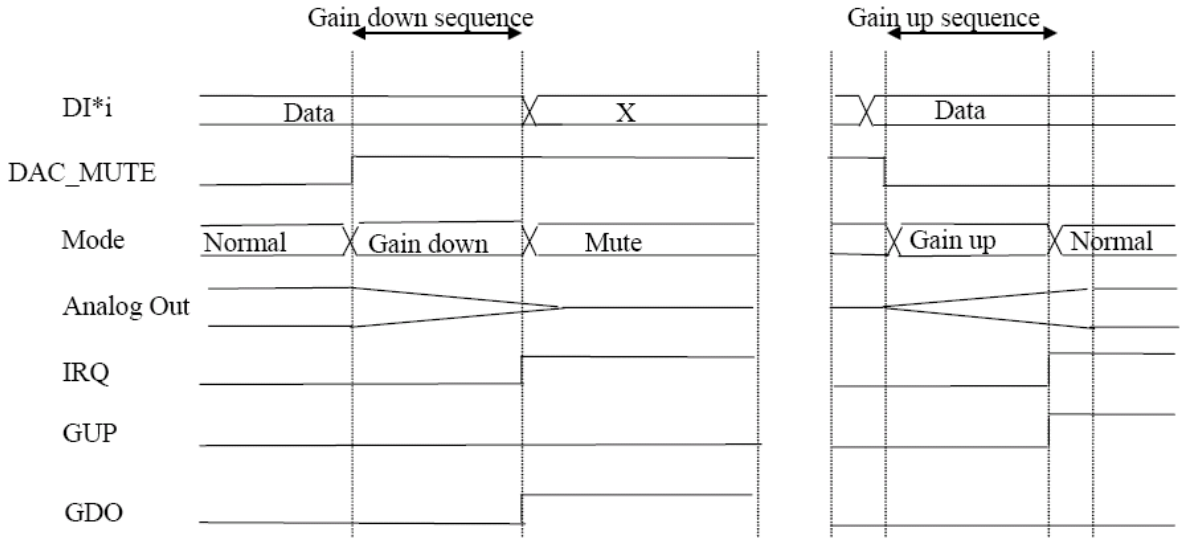


Figure 12-9 Gain up and gain down sequence

In the opposite, when DAC_MUTE is set to 0, the DAC leaves the Soft Mute mode by increasing 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

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.

12.17.6 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".

12.17.7 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_MICBIAS	SB_MIC1	SB_MIC2	SB_LIN	SB_LIBY	IN_SEL	HP_SEL	LO_SEL	MIC_STEREO	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	00	00	00	1	0	0	1	1	1
Complete power-down mode	1	x	x	x	x	x	x	x	x	x	x	xx	xx	xx	x	x	x	x	x	x
Sleep mode	0	1	x	x	x	x	x	x	x	x	x	xx	xx	xx	x	x	x	x	x	x
Record Mode																				
Mono MIC1 input	0	0	x	x	x	0	x	0	x	x	x	00	xx	xx	0	x	x	x	x	x
Mono MIC2 input	0	0	x	x	x	0	x	x	0	x	x	01	xx	xx	0	x	x	x	x	x
Record Mode, stereo MIC inputs,																				
MIC1 to left channel	0	0	x	x	x	0	x	0	x	x	x	00	xx	xx	1	x	0	x	x	x
MIC2 to left channel	0	0	x	x	x	0	x	x	0	x	x	01	xx	xx	1	x	0	x	x	x
Record Mode, Line input	0	0	x	x	x	0	x	x	x	0	x	10	xx	xx	x	x	0	x	x	x
Playback mode, DAC to HP	0	0	0	0	x	x	x	x	x	x	x	xx	11	xx	x	0	x	0	0	x
Playback mode, DAC to LO	0	0	0	x	0	x	x	x	x	x	x	xx	xx	11	x	x	x	x	x	0
Bypass mode, Line to HP	0	0	x	0	x	x	x	x	x	x	0	xx	10	xx	x	x	x	x	0	x
Bypass mode, Line to LO	0	0	0	x	0	x	x	x	x	x	x	xx	xx	10	x	x	x	x	x	0
Sidetone mode,																				

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Mono MIC1 input to HP	0	0	x	0	x	x	x	0	x	x	x	xx	00	xx	0	x	x	x	0	x
Mono MIC2 input to HP	0	0	x	0	x	x	x	x	0	x	x	xx	01	xx	0	x	x	x	0	x
Sidetone mode, stereo MIC to HP,																				
MIC1 to left channel	0	0	x	0	x	x	x	0	x	x	x	xx	00	xx	1	x	x	x	0	x
MIC2 to left channel	0	0	x	0	x	x	x	x	x	x	x	xx	01	xx	1	x	x	x	0	x
Sidetone mode,																				
Mono MIC1 input to LO								0	x	x	x	xx	xx	00	0	x	x	x	x	0
Mono MIC2 input to LO	0	0	x	x	0	x	x	x	0	x	x	xx	xx	01	0	x	x	x	x	0
MIC1+MIC2 to LO	0	0	x	x	0	x	x	0	0	x	x	xx	xx	0x	1	x	x	x	x	0

12.18 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.

12.19 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’.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.20 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.

12.20.1 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

12.20.2 Start up sequence (DAC)

This sequence is from Power-on mode to CODEC REPLAY mode.
The output sound is drived 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

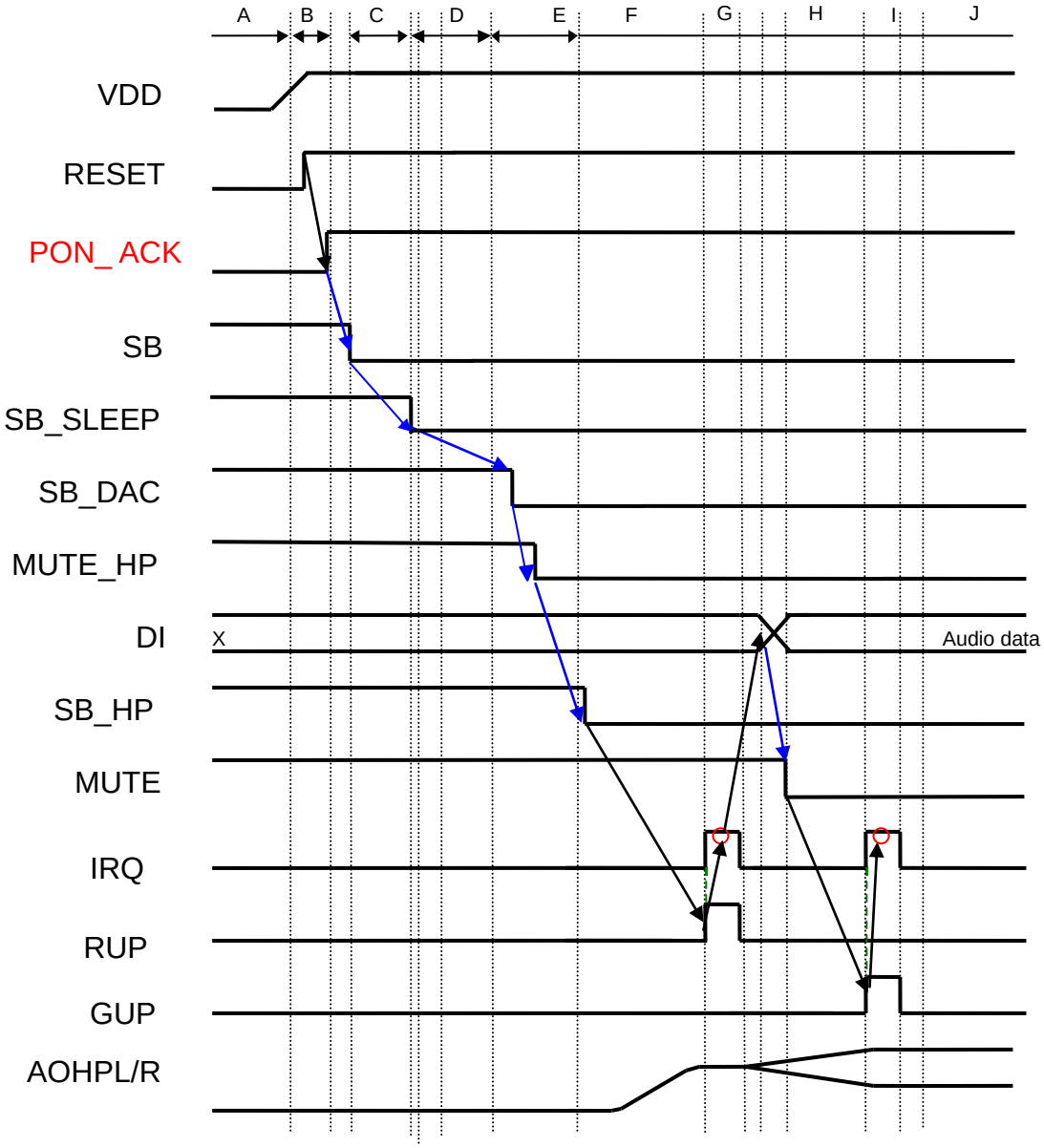


Figure 12-10 Start up sequence

NOTES:

1 The sequences in **blue** are manually handled by the software.

2 The sequences in **black** are automatically handled by the CODEC.

3 **Red** circles are interrupts automatically generated by the CODEC.

SEQUENCE:

A Initial state.

The power supply is off.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

B Power supply ramp up.

The RESET of CODEC is '0' during system reset. The CODEC starts its internal initialization sequence and set PON_ACK register bit once completed.

C Starting of CODEC reference.

The software turns the CODEC on SLEEP mode by clearing SB register bit to 0. The duration equals Tsbyu . After waiting the Tsbyu duration (for example, on event generated by a timer at the application level), the CODEC is in SLEEP mode, the ADC and DAC path are ready to be turn to active mode.

Please refer to the section "Timing parameters" for the Tsbyu Value.

D Go from SLEEP mode to active.

The application turns on the DAC by clearing SB_SLEEP register bits to 0.

E Turn on DAC.

Once after leaving SLEEP mode, the application turns on the DAC (SB_DAC = 0) and after 0.5 ms switch the analog mute signal of the port to activated (MUTE_HP = 0).

F Ramp up cycle.

After waiting 1 ms, the application turns on the headphone output stage (SB_HP = 0).

G Ramp up IRQ generation.

Once the ramp up cycle completes, the CODEC sets the RUP flag to 1 and generates an interrupt.

H IRQ handling and gain up cycle.

The application handles the interrupt, resets the RUP flag by writing 1 on it and releases the mute of the DAC (DAC_MUTE = 0). In the same time, the application sends valid audio data to the CODEC DAC.

I Gain up IRQ generation.

Once the gain up cycle completes, the CODEC sets the GUP flag to 1 and generates an interrupt.

J IRQ handling and DAC active mode.

The application handles the interrupt and resets the GUP flag by writing 1 on. The CODEC DAC path is now fully activated.

12.20.3 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

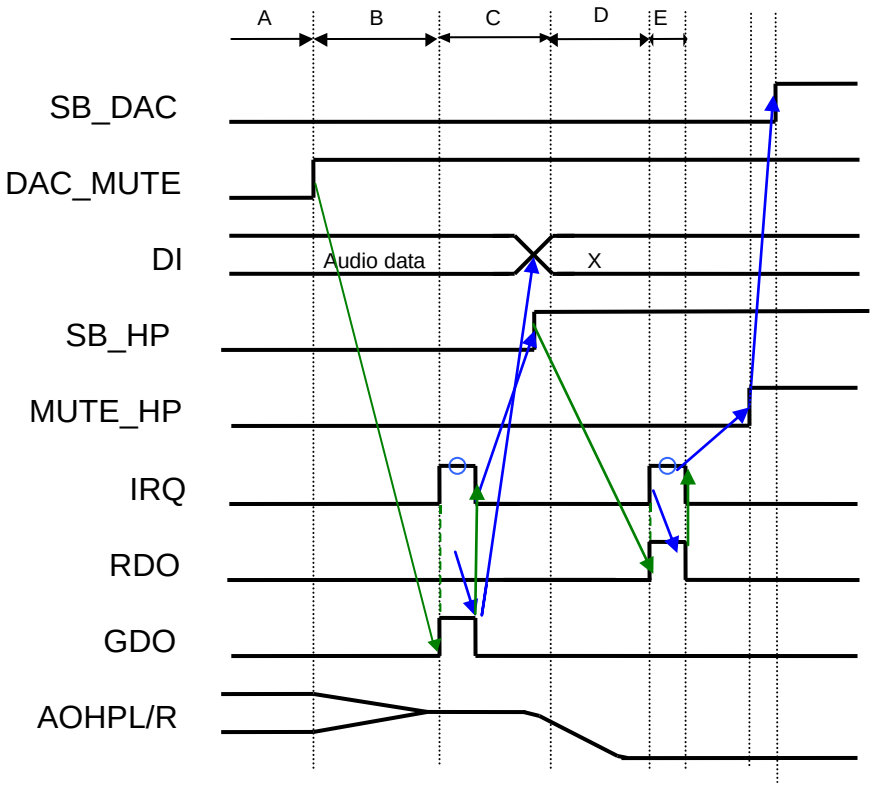


Figure 12-11 Shutdown sequence

NOTES:

1 The sequences in blue are handled by the software.

2 The sequences in green are automatically handled by the CODEC.

SEQUENCE:

A Initial state.

The power supply is on, CODEC DAC path is fully activated.

B Gain down cycle.

The application activates the mute of the DAC (DAC_MUTE=1). Once the gain down cycle completes, the CODEC sets the GDO flag to 1 and generates an interrupt.

C Gain down IRQ handling and ramp down cycle.

The application handles the interrupt and resets the GDO flag by writing 1 on it. The application can then stop sending audio data and turns off the headphone output stage (SB_HP=1).

D Ramp down IRQ generation.

Once the ramp down cycle completes, the CODEC sets the RDO flag to 1 and generates an interrupt.

E IRQ handling.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The application handles the interrupt and resets the RDO flag by writing 1 on it. Then, the application can activate the analog mute (MUTE_HP=1). Finally, the application turns off the DAC path (SB_DAC=1) to be in sleep mode or turn off the CODEC (SB_SLEEP=1, SB=1).

12.20.4 Start up sequence (Line input)

This sequence is from Power-on mode to CODEC REPLAY mode.
The output sound is driving by Line input.
The intent of the following sequence is to prevent for large audible glitches due to the system start-up with the CODEC.

SEQUENCE:

A initial state.
DAC or Line in channel is already in use, valid analog audio signals are available at the input of the switch matrix.

B initializing output port.
The application first set the line in and headphone gain stages to their minimum values (gain automatically is forced to the minimum values when the port is in power-down mode). This setting is taken into account in few clocks cycles. Set the MUTE_HP=0, Then the application turns on the headphone output stages (SB_HP = 0).

C Ramp up IRQ generation.
Once the ramp up cycle completes, the CODEC sets the RUP flag to 1 and generates an interrupt.

D Ramp up IRQ handling and line in stage gain up.
The application handles the interrupt and resets the RUP flag by writing 1 on it. The application then set the line in gain stage to the wished value.
The maximum duration of the gain ramping equals Trlinemax:

$$Trlinemax = N1 * Tcrossout$$

N1 is the number of line in gain steps.
Please Refer to section "Gain refresh strategy" for the value of Tcrossout.

E Headphone stage gain up.
The application set the headphone gain stage to the wished value. The maximum duration of the gain ramping equals Troutmax:

$$Troutmax = N2 * Tcrossout$$

N2 is the number of headphone gain steps.

F active mode.
The signal path is now fully activated.

12.20.5 Shutdown sequence (Line input)

This sequence is from CODEC REPLAY mode to STANDBY mode.
The output sound is driving by Line input.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The intent of the following sequence is to prevent for large audible glitches due to the system shutdown with the CODEC.

SEQUENCE:

A active mode.

The signal path is now fully activated.

B headphone stage gain down.

The application set the headphone gain stage to the minimum value. The maximum duration of the gain ramping equals Tdoutmax:

Tdoutmax = N3 * Tcrossout

N3 is the number of headphone gain steps.

Please Refer to section "Gain refresh strategy" for the value of Tcrossout.

C line in stage gain down.

The application set the line in gain stage to the minimum value. The maximum duration of the gain ramping equals Tdlinemax:

Tdlinemax = N4 * Tcrossout

N4 is the number of line in gain steps.

D Ramp down cycle.

Then, the application can activate the analog mute (MUTE_HP=1) and turns off the headphone output stages (SB_HP=1).

E Ramp down IRQ generation.

Once the ramp up cycle completes, the CODEC sets the RDO flag to '1' and generates an interrupt.

F Ramp down IRQ handling.

The application handles the interrupt and resets the RDO flag by writing '1' on it. The signal path is now off.

12.21 Circuits design suggestions

This section lists a few PCB design suggestions with different using mode.

12.21.1 Avoid quiet ground common currents

12.21.1.1 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).

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.21.1.2 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.

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).

12.21.2 Headphone connection (Capacitor-coupled)

Capacitor-coupled headphone and line connection

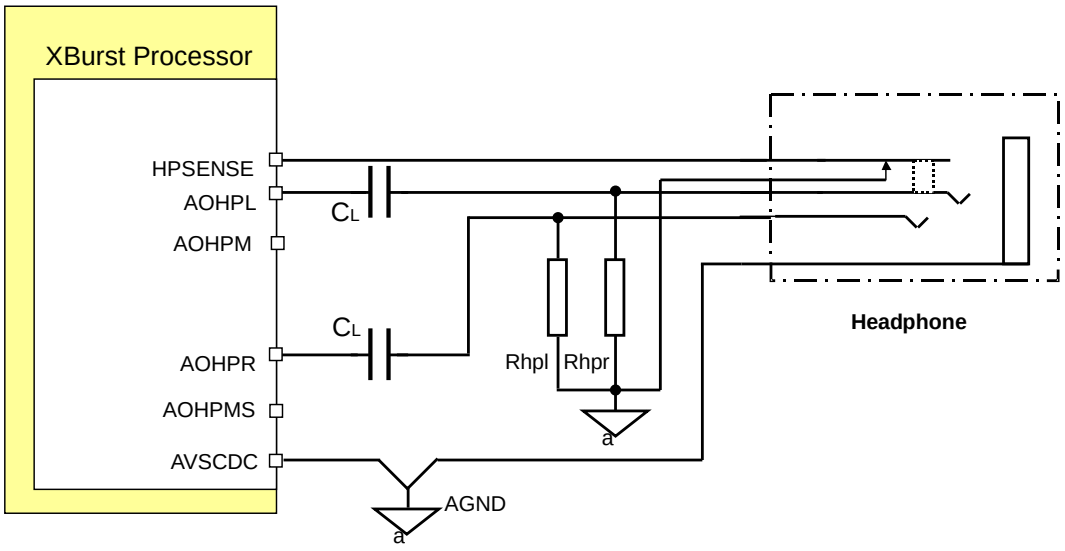


Figure 12-12 Capacitor-coupled connection

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

Capacitor-less headphone connection

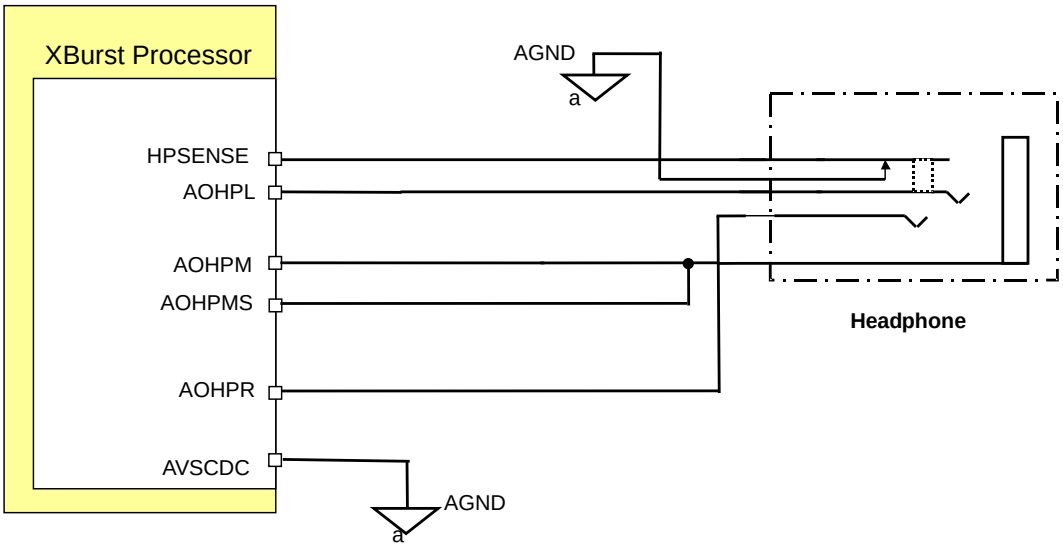


Figure 12-13 Capacitor-less connection

The signals AOHPL and AOHPR from chip are applied directly to the loads. The ground of the headphone is connected to AOHPM. The DC value of the signal AOHPi equals to VREF/2.
The measurement ground reference corresponds to the physical interconnection of AOHPM and AOHPMS.
The measurement is done between AOHPL/R and the measurement ground reference.

NOTE: If you want to use headphone as the antenna for FM , you had better choose this mode.

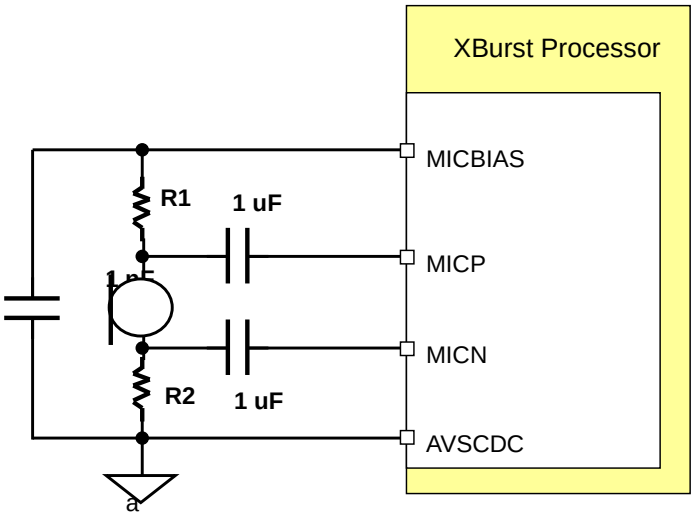
12.21.3 Microphone connection

The optimal performance for the SNR is obtained in differential Microphone inputs with a FS input level corresponding to peak-to-peak amplitude of the signal is 0.2125V when GIM = 7 (Gain value = 20).

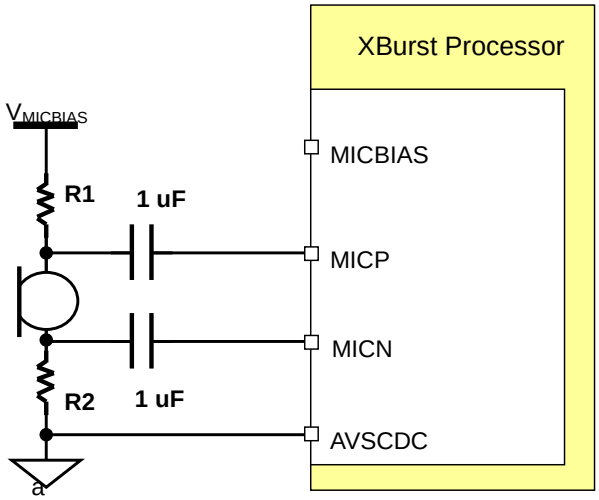
We recommend customer to use differential MIC input for better performance.

Application schematic with differential MIC input (using MICBIAS pin):

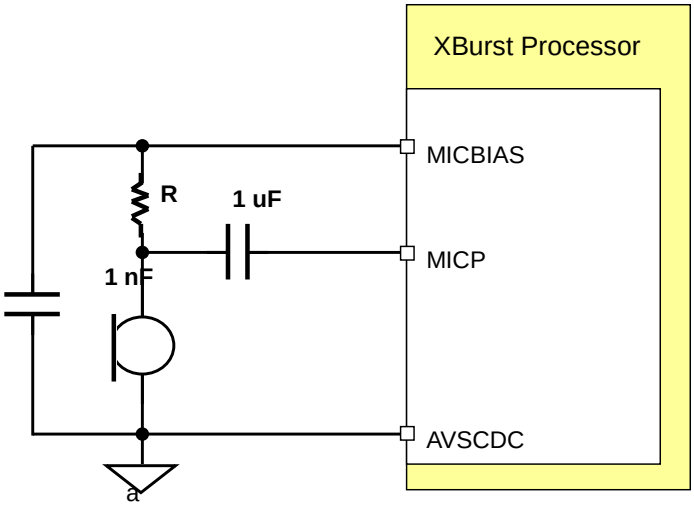
Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Application schematic with differential MIC input (Vmicbias generated on board):

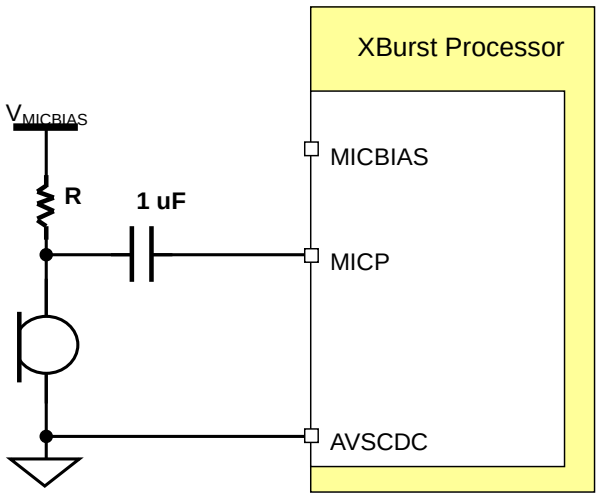


Application schematic with single-ended MIC input (using MICBIAS pin):



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Application schematic with single-ended MIC input (Vmicbias generated on board):



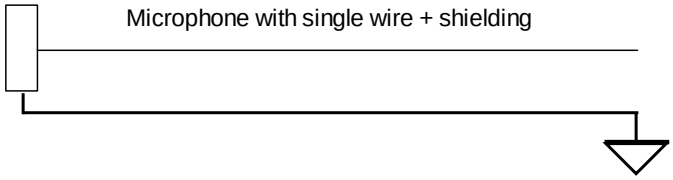
In single-ended connection, one external resistor (R) has to be used to bias the electret microphone. In differential connection, a pair of external resistor (R1, R2) has to be used to bias the electret microphone. The resistors value relation between them is R1 = R2 = R/2.

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.

This configuration is better suited for microphone with single wire + shielding.

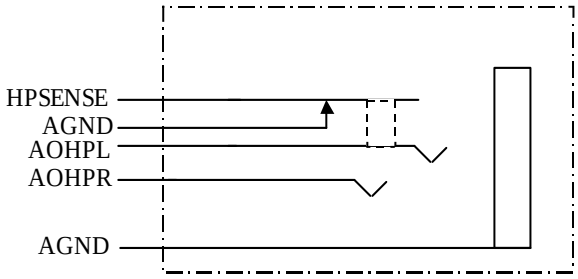


The AVSCDC Pin is connected the analog quiet reference ground in the chip (refers to Grounds and analog signal references). So the ground of MIC must be connected to AVSCDC using a star connection.

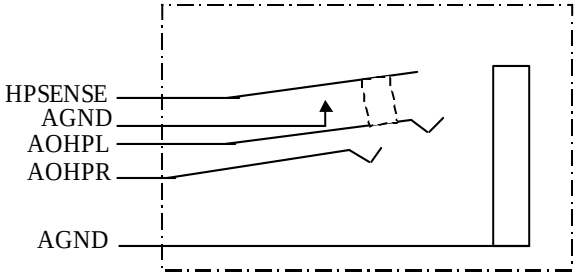
12.21.4 Description of the connections to the jack

When the jack is inserted, “sense” and “ground” are disconnected.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



No jack plugged: the switch acts as a short-circuit.



Jack plugged: the switch acts as an open circuit.

12.21.4.1 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.

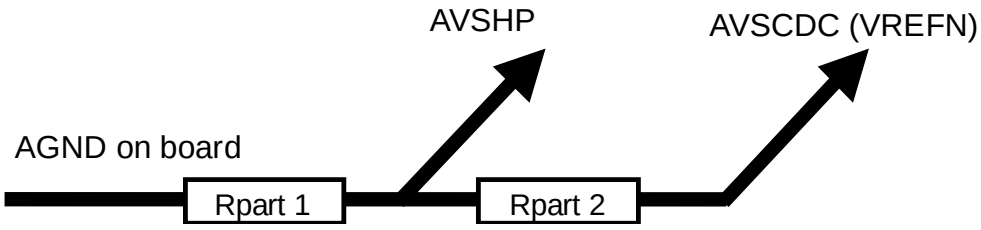


Figure 12-14 Ground distributing

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.

12.21.5 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

(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:

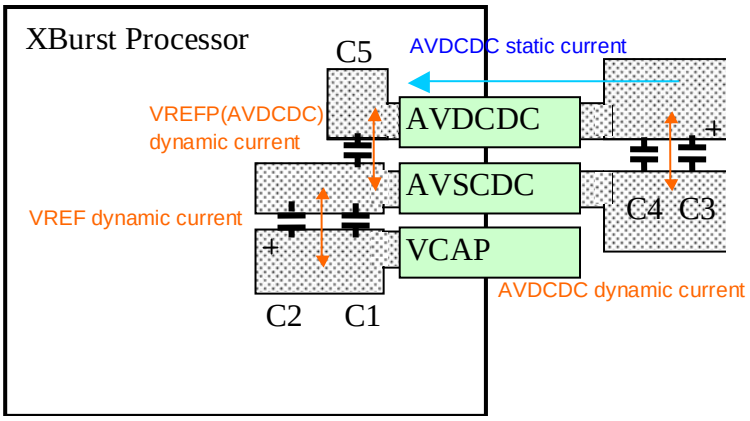


Figure 12-15 the bottom corner of chip PCB Layer

This is just an example reference diagram. You should change and select the PCB layer and route with your design constraints.

12.21.5.1 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	100	nF

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	with C4).		
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

12.22 Analog characteristics

12.22.1 Line input to audio ADC path

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Input level	Full Scale, Gain GIDL, GIDR = 0dB (note 1)	1.89	2.12	2.39	Vpp
Input resistance		8.5			kOhm
Input capacitance	Includes 10pF for ESD, bonding and package pins capacitances			25	pF
Input bypass capacitor	Cbyline		1		uF

NOTE: The Full Scale input voltage scales with AVDCDC, equals to 0.85*VREF (typ).

12.22.2 Microphone input to audio ADC path

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Input level	Full Scale, Gain GIDL, GIDR = 0dB, boost gain GIM1,GIM2 = 20dB (note 1)	0.189	0.212	0.239	Vpp

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Input resistance (Differential mic configuration)	Boost gain GIM1,GIM2 = 0 dB	66	80	100	kOhm
	Boost gain GIM1,GIM2 = 20 dB	10	13	15	
Input resistance (single-ended mic configuration)	Boost gain GIM1,GIM2 = 0 dB	92	115	138	kOhm
	Boost gain GIM1,GIM2 = 20 dB	19	24	29	
Input capacitance	Includes 10pF for ESD, bonding and package pins capacitances			25	pF
Input bypass capacitor	Cbyline		1		uF

NOTE: The Full Scale input voltage scales with AVDCDC, equals to 0.085*VREF (typ).

12.22.3 Audio DAC to headphone output path

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96 kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
DAC playback on 16 Ohm HeadPhone					
Output level	Full Scale, Gain GOL, GOR = -3 dB, GODL, GODR=0dB, 16 Ohm load	1.33	1.5	1.69	Vpp
Maximum output power	Rl = 16 Ohm		17.6		mW
Output resistance	R1	16			Ohm
Output bypass capacitor	Cl (Rl = 16 Ohm)			220	uF
DAC playback to 10k Ohms lineout single					
Output level	Full Scale, Gain GOL, GOR = 0 dB, GODL, GODR=0dB (note 1)	1.89	2.12	2.39	Vpp
Output resistance	R1	10k			Ohm
Output bypass capacitor	Cl (Rl = 10 kOhm)			1	uF
Common characteristics					
Output capacitance (note 2)	Cp			200	pF

NOTES:

1 The Full Scale output voltage scales with AVDCDC, equals to 0.85*VREF. The minimum and maximum output levels are given with gain accuracy.

2 Output may oscillate above specified load capacitances. The capacitance is equivalent to a 2-meter cable.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.22.4 Audio DAC to mono line output path

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Output level	Full Scale, Gain GODL, GODR = 0dB (note 1)	3.78	4.25	4.78	Vpp
Output resistance		10			kOhm
Output capacitance	Cp			100	pF
Output bypass capacitor	Cl (Rl = 10 kOhm)			1	uF

NOTE: The Full Scale output voltage scales with AVDCDC, equals to 1.7*VREF (typ).

12.22.5 Line input to headphone output path (analog bypass)

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96 kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Input level	Full Scale	1.89	2.12	2.39	Vpp
Input resistance		8.5			kOhm
bypass on 16 Ohm HeadPhone					
Output level	Full Scale, Gain GOL, GOR = -3 dB, GIL, GIR=0dB, 16 Ohm load	1.33	1.5	1.69	Vpp
Output resistance	R1	16			Ohm
bypass to 10k Ohms lineout single					
Output level	Full Scale, Gain GOL, GOR = 0 dB, GIL, GIR=0 dB (note 1)	1.89	2.12	2.39	Vpp
Common characteristics					
Input capacitance	Includes 10pF for ESD, bonding and package pins capacitances			25	pF
Input bypass capacitor	Cbyline		1		uF

NOTE: The Full Scale output voltage scales with AVDCDC, equals to 1.7*VREF (typ).

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12.22.6 Microphone input to headphone output path (analog sidetone)

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Input level	Full Scale, Gain GOL, GOR = 0dB, boost gain GIM1,GIM2 = 20dB (note 1)	0.189	0.212	0.239	Vpp
Output level	Full Scale, Gain GOL,GOR= 0dB, boost gain GIM1,GIM2 = 0 to 20dB, 10kOhm load (note 2)	1.89	2.12	2.39	Vpp
	Full Scale, Gain GOL,GOR= -3 dB, boost gain GIM1,GIM2 = 0 to 20dB, 16Ohm load (note 2)	1.33	1.5	1.69	Vpp

NOTES:

1 The Full Scale input voltage scales with AVDCDC, equals to 0.085*VREF (typ).

2 The Full Scale output voltage scales with AVDCDC, equals to 0.85*VREF (typ).

12.22.7 Micbias and reference

Measurement conditions: T = 25°C, AVDCDC = AVDHP = VREFP = 2.5V, input sine wave with a frequency of 1kHz, Fmclk = 12MHz, Fs = 8 to 96kHz, measurement bandwidth 20Hz – 20kHz, unless otherwise specified.					
Parameter	Test conditions	Min.	Typ	Max.	Unit
Micbias output level	(note 1)		2.08 1.66		V
Micbias output current				4	mA
Micbias decoupling capacitor	Cmic	0.75	1	1.25	nF
VCAP voltage	(note 2)		2		V

NOTES:

1 Micbias output voltage scales with AVDCDC, equals to 5/6*VREF or 4/6*VREF (typ).

2 VCAP output voltage scales with AVDCDC, equals to 0.8*VREF (typ).

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 4

MEMORY INTERFACE

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

13 DDR Controller

13.1 Overview

DDRC (DDR Controller) is a general IP which provides an interface to DDR2, DDR3, mobile DDR(LPDDR) memory. The DDRC IP is designed for SOC usage and 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

13.1.1 Supported DDR SDRAM Types

- DDR2
- DDR3
- mobile DDR(LPDDR)

Row address width less than 16-bit & Column width less than 12-bit are supported.

13.1.2 Block Diagram

Following figure shows the functional block diagram of DDRC.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

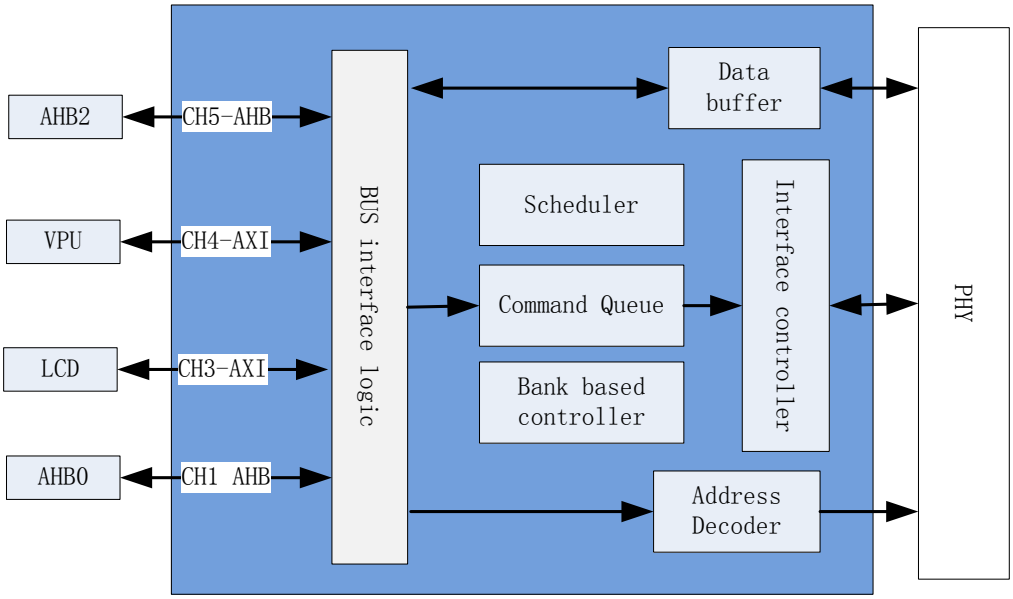


Figure 13-1 DDRC block diagram

13.2 Register Description

Following table lists the registers of DDR Controller. All of these registers are 32bit

All DDRC register can be accessed by channel 1: CH1_AHB.
The physical address base is 0x13010000.

Table 13-1 DDRC Register

Name	Address offset	Width	Access	Description
DSTATUS	0x00	32	RW	DDR Status Register
DCFG	0x04	32	RW	DDR Configure Register
DCTRL	0x08	32	RW	DDR Control Register
DLMR	0x0C	32	RW	DDR Load-Mode-Register
DTIMING1	0x60	32	RW	DDR Timing Configure Register 1
DTIMING2	0x64	32	RW	DDR Timing Configure Register 2
DTIMING3	0x68	32	RW	DDR Timing Configure Register 3
DTIMING4	0x6C	32	RW	DDR Timing Configure Register 4

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

DTIMING5	0x70	32	RW	DDR Timing Configure Register 5
DTIMING6	0x74	32	RW	DDR Timing Configure Register 6
DREFCNT	0x18	32	RW	Auto-Refresh Counter
DMMAP0	0x24	32	RW	DDR Memory CS0 Map Configure Register
DMMAP1	0x28	32	RW	DDR Memory CS1 Map Configure Register
DDLP	0xBC	32	RW	DFI low power handshake register
DREMAP1	0x9C	32	RW	DDR address remapping register1
DREMAP2	0xA0	32	RW	DDR address remapping register2
DREMAP3	0xA4	32	RW	DDR address remapping register3
DREMAP4	0xA8	32	RW	DDR address remapping register4
DREMAP5	0xAC	32	RW	DDR address remapping register5
DSTRB	0x34	32	RW	Multi-media stride register
WCMDCTRL1	0x100	32	RW	Write command reorder & grouping (Performance control)
RCMDCTRL0	0x104	32	RW	Read Channel mode control (Performance control)
RCMDCTRL1	0x108	32	RW	Read Channel mode control (Performance control)
WDATTHD0	0x114	32	RW	Wdata Channel mode control (Performance control)
WDATTHD1	0x118	32	RW	Wdata Channel mode control (Performance control)
IPORTPRI	0x128	32	RW	Configuration of Internal Priority (Performance control)
IPORTWPRI	0x240	32	RW	Configuration of Internal Priority for write channel (Performance control)
IPORTRPRI	0x244	32	RW	Configuration of Internal Priority for read channel (Performance control)
CHxWDOS	0x200 0x204	32	RW	WQoS configure for each channel (Performance control)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	0x208 0x20C 0x210 0x214 0x218			
CHxRDOS	0x220 0x224 0x228 0x22C 0x230 0x234 0x238	32	RW	RQoS configure for each channel (Performance control)
CPM_DRCG	**0xB0000 0D0**	32	RW	CPM register, DLL & Clock control for DDR

13.2.1 DSTATUS

	DSTATUS																															
																	0x13010000															
Bit	31	30	29	28	27	26	25	24	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	Reserved	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	0

Bits 31~8,1: Reserved. Write has no effect, read value not need to care.
ENDIAN: Read-only, indicate the data endian status.

Bit [7]	Description	Remark
0	Little data Endian.	(reset value)
1	Big data Endian.	

MISS: Indicate the accessed address is out of memory mapping range. (This bit is writable)

Bit [6]	Description	Remark
0	No operation miss DDRC memory mapping.	(reset value)
1	At last one operation miss DDRC memory mapping.	

DPDN: Indicate the deep-power-down status of DDR memory.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bit [5]	Description	Remark
0	DDR memory is NOT in deep-power-down state.	(reset value)
1	DDR memory is in deep-power-down state.	

PDN: Indicate the power-down status of DDR memory.

Bit [4]	Description	Remark
0	DDR memory is NOT in power-down state.	(reset value)
1	DDR memory is in power-down state.	

AREF: Indicate the auto-refresh status of DDR memory.

Bit [3]	Description	Remark
0	DDR memory is NOT in auto-refresh state.	(reset value)
1	DDR memory is in auto-refresh state.	

SREF: Indicate the self-refresh status of DDR memory.

Bit [2]	Description	Remark
0	DDR memory is NOT in self-refresh state.	(reset value)
1	DDR memory is in self-refresh state.	

CKE1: not support in this version.

Bit [1]	Description	Remark
0	CKE1 Pin is low.	(reset value)
1	CKE1 Pin is high.	

CKE0: Indicate the CKE0 Pin status of DDR memory.

Bit [0]	Description	Remark
0	CKE0 Pin is low.	(reset value)
1	CKE0 Pin is high.	

13.2.2 DCFG

Configure the external memory, static configuration only. ie. This register can NOT be changed on-the-fly.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DCFG																0x13010004													
Bit	31	30	29	28	27	26	25	24	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	COL1			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 31, 30 , 20,14: Reserved. Writing has no effect, read as zero.

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.

BSL: Burst length for DDR chips

1: 8 burst

0: 4 burst

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 . 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;

TYPE: Select external memory device type.

Bit [19:17]	Description	Remark
000	Normal SDR (Single-Data-Rate) SDRAM(Not support).	(reset value)
001	Mobile SDR(Not support).	
010	Normal DDR1 (Not support)	
011	Mobile DDR(LPDDR).	
100	Normal DDR2.	
101	LPDDR2(Not support)	
110	Normal DDR3	
111	LPDDR3 (Not support).	

ROW0/1: Row Address width. Specify the row address width of external DDR.

	Description	Remark
000	12-bit row address	(reset value)
001	13-bit row address	
010	14-bit row address	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

011	15-bit row address	
100	16-bit row address	
Reserved		

COL0/1: Column Address width. Specify the Column address width of external DDR.

	Description	Remark
000	8-bit Column address.	(reset value)
001	9-bit Column address	
010	10-bit Column address	
011	11-bit Column address	
100	12-bit Column address	
Reserved		

CS1EN: DDR Chip-Select-1 Enable.

If an DDR memory chip is connected to ddr pin cs1, set CS1EN=1.

Bit [7]	Description	Remark
0	DDR Pin CS1 is not in use.	(reset value)
1	DDR Pin CS1 is in use.	

CS0EN: DDR Chip-Select-0 Enable.

If DDR memory is connected to ddr pin cs0, set CS0EN=1.

Bit [6]	Description	Remark
0	DDR Pin CS0 is not in use.	(reset value)
1	DDR Pin CS0 is in use.	

CL: CAS Latency.

Bit [5:2]	Description	Remark
Reserved	Not in use for this version	

BA0/1: Bank Address width of DDR memory.

Bit [1]	Description	Remark
0	4 bank device, Pin ba[1:0] is valid, ba[2] un-used.	(reset value)
1	8 bank device, Pin ba[2:0] is valid.	

DW: External DDR Memory Data Width.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Specify the external DDR memory data width.

Bit [0]	Description	Remark
0	External memory data width is 16-bit.	(reset value)
1	External memory data width is 32-bit.	

13.2.3 DCTRL

On the positive edge of START, the command selected by CMD field will be performed.

	DCTRL																0x13010008														
Bit	31	30	29	28	27	26	25	24	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

Bit 31~24, 19,18,16,10~7,2: Reserved. Writing has no effect, read as zero.
CFG_RST: Reserved, please keep the default value.
CTL_RST: Reserved, please keep the default value.
DLL_RST: Reserved, please keep the default value.
DFI_RST: Reserved, please keep the default value.
KEEPSR: inner usage, keep to 0.
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 actived
ACTPD: Active Power-Down.
Some SDRAM devices support Active-Power-Down.
By default, ACTPD=0, hardware will percharge all active banks before entering Power-Down
mode, so called Precharge-Power-Down.
By setting ACTPD=1, hardware drives SDRAM into Power-Down mode without precharge all
active banks, some banks are still active in Power-Down mode, so called Active-Power-Down.

Bit [15]	Description	Remark
0	Precharge all banks before entering power-down.	(reset value)
1	Do not precharge all banks before entering power-down.	

PDT: Power-Down Timer.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

Bit [14:13]	Description	Remark
000	power-down disabled, hardware never drive SDRAM into power-down mode.	(reset value)
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.	

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 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!!**

Bit [5]	Description	Remark
0	Drive external DDR device entering self-refresh mode.	(reset value)
1	Drive external DDR device exiting self-refresh mode.	

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

UNALIGN: Enable unaligned transfer on AXI BUS.

Bit [4]	Description	Remark
Reserved	Not use in this version	

ALH: Advanced Latency Hiding.

This is a test-oriented register.

Some latency is reduced in special cases.

Bit [3]	Description	Remark
0	Disable ALH.	(reset value)
1	Enable ALH.	

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.

Bit [1]	Description	Remark
0	Not set CKE Pin High.	(reset value)
1	Set CKE Pin to HIGH.	

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.

Bit [0]	Description	Remark
0	End resetting ddr_controller.	(reset value)
1	Resetting ddr_controller.	

13.2.4 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DLMR																													0x1301000C
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDR_ADDR																				Reserved	BA			Reserved		CMD			Reserved		START
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit 7~6, 2~1: Reserved. Writing has no effect, read as zero.
DDR_ADDR: When performing a DDR command, DDR_ADDR[13:0] corresponding to
external DDR address Pin A[13:0]; DDR_ADDR[15:14] are reserved.

Bit [31:12]	Description	Remark
0000_0000	{MA[9:0],OP[9:0]} for LPDDR2(not support in this version) For other DDR the low bit 14 bit corresponding to external DDR address Pin A[13:0].	(reset value)

BA: Bank Address.
When performing a DDR command, BA[2:0] corresponding to external DDR address Pin BA[2:0].

Bit [10:8]	Description	Remark
000	corresponding to external DDR address Pin BA[2:0].	(reset value)

CMD: Select command to process when setting START from low to high.
On the positive edge of START, one of the following commands will be performed.

Bit [5:3]	Description	Remark
000	Precharge one bank / All banks. (dependent field : BA, DDR_ADDR)	(reset value)
001	Auto-Refresh.	
010	Load Mode Register. (dependent field : BA, DDR_ADDR)	
010	ZQCS for DDR3	
100	ZQCL for DDR3	
101~111	Reserved.	

START: Start to perform a command to external DDR memory.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The command is performed on the positive edge of START; Hardware will clear START bit to zero when command issued out to external DDR memory.

Write 0 to START will be ignored and take no effect;

START=1 means hardware is busy executing current command and can NOT accept new command;

Software must check START=0 before writing 1 to START.

Bit [0]	Description	Remark
0	No command is performed.	(reset value)
1	On the positive edge of START, perform a command defined by CMD field.	

13.2.5 DTIMING1,2,3,4,5,6 (DDR Timing Configure Register)

The timing parameters are identical to the JEDEC DDR Specification.

	DTIMMING1																													0x13010060
Bit	31	30	29	28	27	26	25	24	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

	DTIMMING2																													0x13010064
Bit	31	30	29	28	27	26	25	24	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	2	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

	DTIMMING3																													0x13010068
Bit	31	30	29	28	27	26	25	24	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

	DTIMING4																													0x1301006C
Bit	31	30	29	28	27	26	25	24	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DTIMMING5																													0x13010070
Bit	31	30	29	28	27	26	25	24	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

	DTIMMING6																													0x13010074
Bit	31	30	29	28	27	26	25	24	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

If the DDR clock period are tCK.

All these registers's value equal to the number tCK cycle. For example, if tRTP=2, it means 2 tCK between READ and PRECHARGE;

tRTP: READ to PRECHARGE command period.

tWTR: WRITE to READ command delay.

tWR: WRITE Recovery Time defined by register MR of DDR2 DDR3 memory.

tWL: Write latency, please notice that this version only support AL=0.

tCCD: CAS# to CAS# command delay

tRAS: ACTIVE to PRECHARGE command period.

tRAS defines the ACTIVE to PRECHARGE command period to the same bank.

tRCD: ACTIVE to READ or WRITE command period.

tRCD defines the ACTIVE to READ/WRITE command period to the same bank.

tRL: Read latency, please notice that this version only support AL=0.

ONUM: Keep to 4 in this version.

tRP: PRECHARGE command period

tRP defines the PRECHARGE to next command period to the same bank.

tRRD: ACTIVE bank A to ACTIVE bank B command period.

tRRD defines the ACTIVE to ACTIVE command period to **different** banks.

tRC: ACTIVE to ACTIVE command period.

tRC defines the ACTIVE to ACTIVE command period to the same bank.

tWDLAT: tWL-1(when use LPDDR2, set tWDLAT=tWL).

tRDLAT: tRL-2. (when use LPDDR2, set tRDLAT=tRL).

tFAW: 4-active command window.

tCFGW: Write PHY configure registers to other commands delay. Not need to change.

tCFGR: Ready PHY configure registers to other commands delay. Not need to

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

change.

tRTW: Read to Write latency.

tCTLUPD: Inner usage. Not need to change.

tEXTRW: keep the default value.

tRWCOV: keep the default value.

All these registers' value has different rate between the tCK cycle

tXSRD: exit self-refresh to READ delay.

Delay time is tXSRD*4 (tCK)

Note: You can set this registers as tXS(or tXSNR), but must make sure the system do not read ddr before tXSRD in wakeup.

tCKSRE: Valid clock after enter self-refresh(tCKSRX = tCKSRE in this version).

Delay time is tCKSRE*8 (tCK)

tCKE: minimum CKE pulse width.

tCKE define the minimum CKE pulse width, include high level and low level.

Bit [18:16]	Description	Remark
000	1 tCK.	(reset value)
001	2 tCK.	
010	3 tCK.	
011	4 tCK.	
100	5 tCK.	
101	6 tCK.	
110	7 tCK.	
111	8 tCK.	

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).**

Bit [29:24]	Description	Remark
000000	1 tCK.	(reset value)
000001	3 tCK.	
000010	5 tCK.	
000011	7 tCK.	
… …	**… 2 * (tRFC + 1) …**	
111101	125 tCK.	
111110	127 tCK.	
111111	129 tCK.	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

* tCK – one DDR memory clock cycle, typical tCK value is 7.5 ns (133MHz clock).

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.

Bit [11:8]	Description	Remark
0000	1*8 + 1 tCK.	(reset value)
0001	2*8 + 1 tCK.	
0010	3*8 + 1 tCK.	
0011	4*8 + 1 tCK.	
… …	**… tMINSR * 8 + 1 …**	
1101	14*8 + 1 tCK.	
1110	15*8 + 1 tCK.	
1111	16*8 + 1 tCK.	

tXP: EXIT-POWER-DOWN to next valid command period.

tXP defines the EXIT-POWER-DOWN to next valid command period to all banks.

Bit [6:4]	Description	Remark
000	1 tCK.	(reset value)
001	1 tCK.	
010	2 tCK.	
011	3 tCK.	
100	4 tCK.	
101	5 tCK.	
110	6 tCK.	
111	7 tCK.	

tMRD: Load-Mode-Register to next valid command period.

tMRD defines the Load-Mode-Register to next valid command period.

Bit [1:0]	Description	Remark
00	1 tCK.	(reset value)
01	2 tCK.	
10	3 tCK.	
11	4 tCK.	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.2.6 DREFCNT (DDR Auto-Refresh Counter)

	DREFCNT																												0x13010018	
Bit	31	30	29	28	27	26	25	24	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 31~24, 7-4: Reserved. Writing has no effect, read as zero.

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.

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.

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 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.

Bit [3:1]	Description	Remark
000	dclk / 16.	(reset value)
001	dclk / 32.	
010	dclk / 64.	
011	dclk / 128.	
100	dclk / 256.	
101	dclk / 512.	
110	dclk / 1024.	
111	--	

REF_EN: Enable Refresh Counter.

Software set REF_EN=1 right after initialize ddr memory.

Bit [29]	Description	Remark
0	Enable auto-refresh counter.	(reset value)
1	Disable auto-refresh counter.	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.2.7 DMMAP0,1 (DDR Memory Map Configure Register)

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.

	DMMAP0,1																													0x13010024, 0x13010028							
Bit	31	30	29	28	27	26	25	24	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 31~16: Reserved. Writing has no effect, read as zero.

BASE: base address.

MASK: address mask.

Examples:

1 DDR address space in system memory : 0x2000_0000 ~ 0x2FFF_FFFF (256MB)

BASE=0x20 MASK=0xF0.

2 DDR address space in system memory : 0x5000_0000 ~ 0x57FF_FFFF (128MB)

BASE=0x50 MASK=0xF8.

NOTE: If DDRC is disabled, please set DMMAP=0x0000_FF00 (reset value).

13.2.8 DDLP (DDR DFI low power handshake control register)

	DDLP																													0x130100BC							
Bit	31	30	29	28	27	26	25	24	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 31~19: Reserved. Writing has no effect, read as zero.

Bits 18~16: Inner usage, keep to 0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

LPEN: DFI low power(lp) interface enable

FPD: 1: Use fast lp handshake process during power down.
0: Use slow lp handshake process during power down.

FSR: 1: Use fast lp handshake process during self-refresh.
0: Use slow lp handshake process during self-refresh.

SLP, FLP: slow LP handshake and fast LP handshake timing register, Just keep it as reset value.

13.2.9 DREMAP1,2,3,4,5 (DDR Address Remapping Register 1,2,3,4,5)

	DREMAP1																													0x1301009C	
Bit	31	30	29	28	27	26	25	24	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

	DREMAP2																													0x130100A0	
Bit	31	30	29	28	27	26	25	24	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																													0x130100A4	
Bit	31	30	29	28	27	26	25	24	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																													0x130100A8	
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DREMAP5																													0x130100AC
Bit	31	30	29	28	27	26	25	24	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

13.2.10 WCMDCTRL1 (Performance wcmd reorder & grouping)

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																													0x13010100	
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	Reset
[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.	0x8
[26:19]	wpage_hit_max	Wpage-hit affects the reorder of wcmd, the wcmd with same page should be bonded 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.	0x8
[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. 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.**	0x100
[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**	0xb
[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.**	0
[0]	msk_empty_wfifo	0: wcmd is involved in arbitration immediately. 1: wcmd is not involved in arbitration until receives enough wdata.	0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.2.11 RCMDCTRL0 (Performance rcmd request control)

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																0x13010104													
Bit	31	30	29	28	27	26	25	24	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						Reserved								ch1_rcmd_igr_cflit	ch1_rfifo_thd_en	ch1_rfifo_thd						Reserved							
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

Bits	Name	Description	Reset
[31]	ch3_rcmd_igr_cflit	1: enable. Consistence between wcmd and rcmd will not be guaranteed. 0: disable.	0x0
[30]	ch3_fifo_thd_en	0: disable rfifo_thd (recommanded) 1: enable rfifo_thd	0x0
[29:24]	ch3_rfifo_thd	If Ch3 rfifo level threshold higher than this value, rcmd request is masked	0x8
[23]	reserved		0x0
[22]	reserved		0x0
[21:16]	reserved		0x8
[15]	ch1_rcmd_igr_cflit	1: enable. Consistence between wcmd and rcmd will not be guaranteed. 0: disable.	0x0
[14]	ch1_fifo_thd_en	0: disable rfifo_thd (recommanded) 1: enable rfifo_thd	0x1
[13:8]	ch1_rfifo_thd	If Ch3 rfifo level threshold higher than this value, rcmd request is masked	0x8
[7]	reserved		0x0
[6]	reserved		0x1
[5:0]	reserved		0x8

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

13.2.12 RCMDCTRL1 (Performance rcmd request control)

	RCMDCTRL1																0x13010108													
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	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	Reset
[15]	ch5_rcmd_igr_cflit	0: disable rfifo_thd (recommended) 1: enable rfifo_thd	0x0
[14]	ch5_fifo_thd_en	1: enable. consistence between wcmd and rcmd will not be guaranteed. 0: disable.	0x1
[13:8]	ch5_rfifo_thd	If Ch1 rfifo level threshold higher than this value, rcmd request is masked	0x8
[7]	ch4_rcmd_igr_cflit	0: disable rfifo_thd (recommended) 1: enable rfifo_thd	0x0
[6]	ch4_fifo_thd_en	1: enable. consistence between wcmd and rcmd will not be guaranteed. 0: disable.	0x0
[5:0]	ch4_rfifo_thd	If Ch1 rfifo level threshold higher than this value, rcmd request is masked	0x8

13.2.13 WDATTHD0 (performance wcmd request control)

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																0x13010114													
Bit	31	30	29	28	27	26	25	24	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								Reserved	Ch1_wfifo_thd_en	Ch1_wfifo_thd						Reserved							
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
30	Ch3_wfifo_thd_en	1: Enable 1: Disable (recommended)	0x0
[29:24]	Ch3_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	0x8
[22]	Ch2_wfifo_thd_en	1: Enable 1: Disable (recommended)	0x0
[21:16]	Ch2_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	0x8
[14]	Ch1_wfifo_thd_en	1: Enable 1: Disable (recommended)	0x0
[13:8]	Ch1_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	0x8
[6]	Reserved		0x0
[5:0]	Reserved		0x8

13.2.14 WDATTHD1 (performance wcmd request control)

WDATTHD1																	0x13010118													
Bit	31	30	29	28	27	26	25	24	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																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
[14]	Ch5_wfifo_thd_en	1: Enable 1: Disable (recommended)	0x1
[13:8]	Ch5_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	0x8
[6]	Ch4_wfifo_thd_en	1: Enable 1: Disable (recommended)	0x0
[5:0]	Ch4_wfifo_thd	Wcmd is involved in arbitration before receiving all wdata.	0x8

13.2.15 IPORTPRI (performance priority control)

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	IPORTWPRI																													0x13010240
Bit	31	30	29	28	27	26	25	24	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								Ch5_iport_wpri_en	Reserved	Ch5_iport_wpri		Ch4_iport_wpri_en	Reserved	Ch4_iport_wpri		Ch3_iport_wpri_en	Reserved	Ch3_iport_wpri		Reserved				Ch1_iport_wpri_en	Reserved	Ch1_iport_wpri		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
	IPORTRPRI																													0x13010244
Bit	31	30	29	28	27	26	25	24	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								Ch5_iport_rpri_en	Reserved	Ch5_iport_rpri		Ch4_iport_rpri_en	Reserved	Ch4_iport_rpri		Ch3_iport_rpri_en	Reserved	Ch3_iport_rpri		Reserved				Ch1_iport_rpri_en	Reserved	Ch1_iport_rpri		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
[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	0x0
[21:20]	Ch5_iport_wpri Ch5_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	0x0
[19]	Ch4_iport_wpri_en Ch4_iport_rpri_en	Enable iport_pri 0: disable, use external priority 1: enable, use the max priority of external and internal priority	0x0
[17:16]	Ch4_iport_wpri Ch4_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	0x0
[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	0x0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

[13:12]	Ch3_iport_wpri Ch3_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	0x0
[11]	reserved		0x0
[9:8]	reserved		0x0
[7]	Ch1_iport_wpri_en Ch1_iport_rpri_en	Enable iport_pri 0: disable, use external priority 1: enable, use the max priority of external and internal priority	0x0
[5:4]	Ch1_iport_wpri Ch1_iport_rpri	Priority of channel 0x3: the highest priority ... 0x0: the lowest priority	0x0
[3]	reserved		0x0
[1:0]	reserved		0x0

13.2.16 CHxQOS0,1,2,3,4,5 (performance QoS control)

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.

	CHxWQOS0,1,2,3,4,5,6																0x13010200,204,208,20C,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																0x13010300,304,308,30C,310,314,318													
Bit	31	30	29	28	27	26	25	24	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	Rrset
[31]	Qos_en	enable	0x1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

[17:16]	Max	Up-limit of priority	0x3
[15:0]	Con	Threshold of timer	0x100

13.2.17 CPM_DRCG

	CPM_DRCG																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	Rrset
[31:2]	Reserved	Inner usage, keep in default value	0x0
[1]	DLLRESET	1: reset the DLL in DDR PHY 0: disable the reset.	0x0
[0]	CFGCLKEN	Inner usage, keep to 1	0x1

13.3 Functional Description

13.3.1 DDRC and DDR2 Memory Initialization Sequence

The following content is just an example for initialization, the sequence is already implementation in our BSP。

13.3.1.1 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.
//----------

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

```
//   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
//-----------------------------------------------------------------------------------------------------------
```

13.4 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.

13.4.1 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.

13.4.2 CPM driven SELF-REFRESH Mode

CPM will auto drive DDRC to self-refresh mode, when use ddr2, ddr3.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

To change clock frequency, please refer CPM spec.

13.4.3 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.

Step1, Configure DLL into reset mode to guarantee a driving clock.

(0xB00000D0) CPM_DRCG[1] <= 1;

Step2, Configure DLL into bypass mode.

(0xB3011014) DDR_ACDLLCR[0] <= 1;

Then, (0xB00000D0) CPM_DRCG[1] <= 0;

13.5 Data Endian

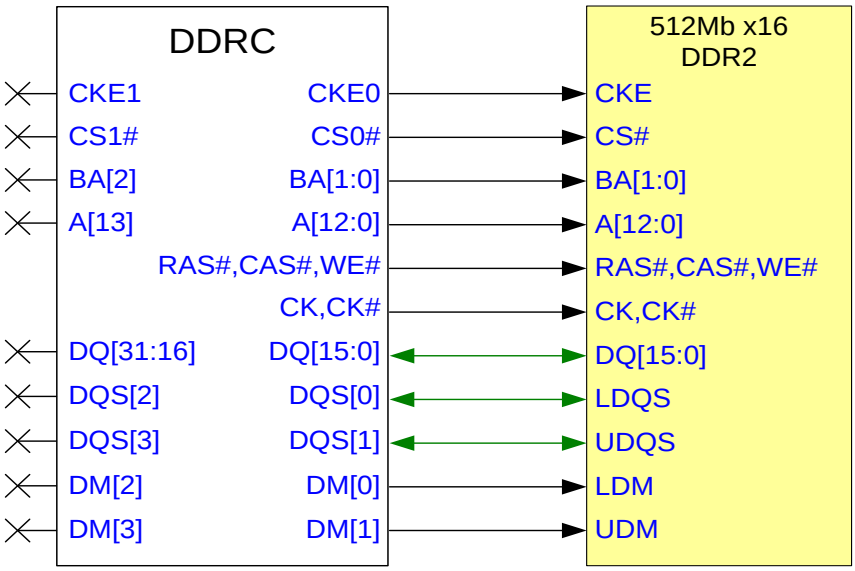
Fix to little Endian.

13.6 DDR Connection Diagrams

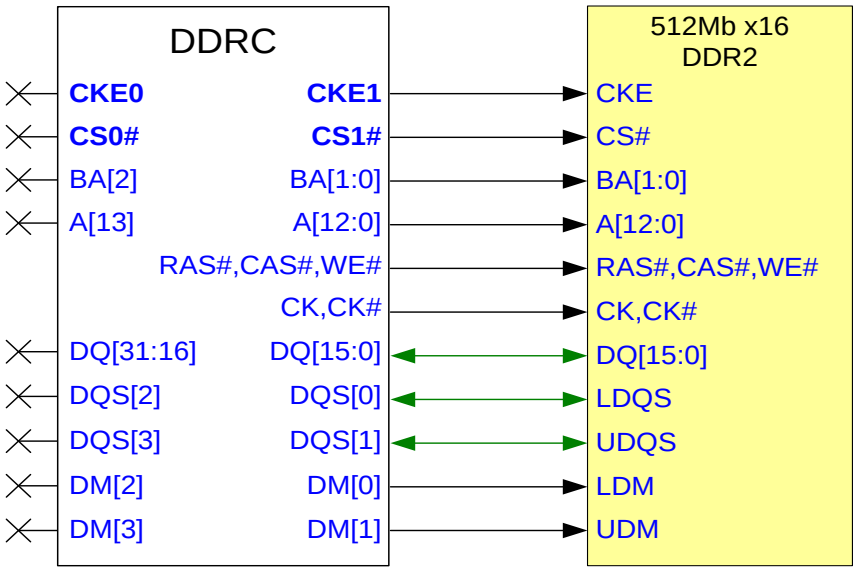
The following diagrams give examples on the connection to external DDR2 devices.

Note not all the possible connections are listed.

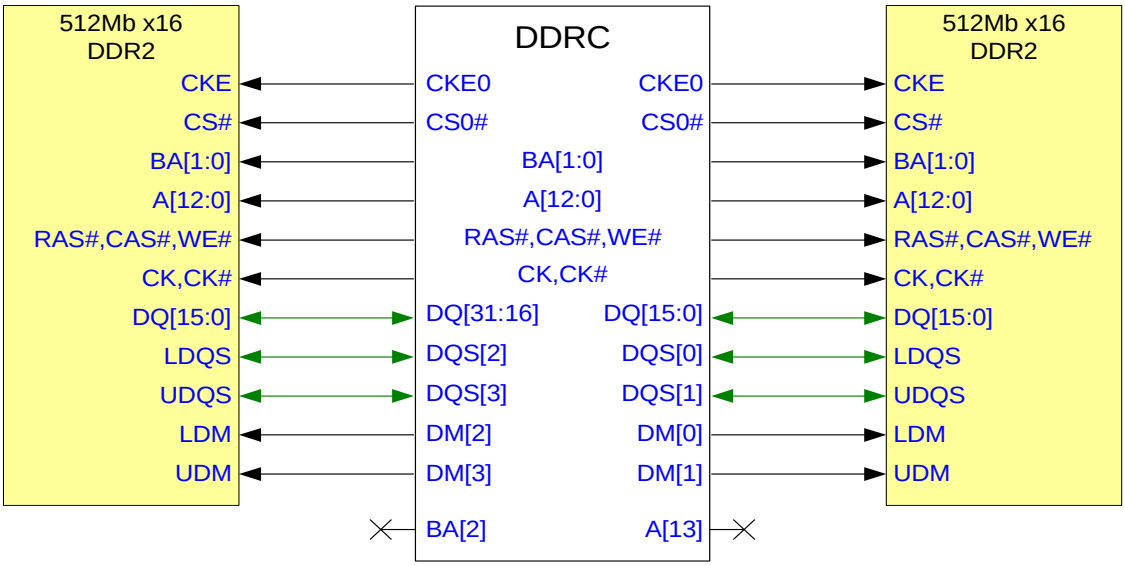
13.6.1 Connection to one 512Mb x16 DDR2 devic



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



13.6.2 Connection to two 512Mb x16 DDR2 devices



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14 External NAND Memory Controller

14.1 Overview

The External NAND Memory Controller (NEMC) 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(8bit and 16bit bus width), Toggle NAND flash memory(ONLY 8bit bus width), 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 8/16-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
  - 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

14.2 Pin Description

Following table list the NEMC pins.

Table 14-1 NEMC Pin Description

Pin Name	I/O	Signal	Description
Data Bus	I/O	SD15 – SD0	Data I/O.
Address bus	O	SA5–SA0	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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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. (When Nand flash boot, GPC30 is used as FRB# of CS1#)
Toggle NAND flash data strobe	I/O	DQS	Toggle NAND data output is aligned with DQS falling/rising edge, and data inputs at DQS falling/rising.

14.3 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

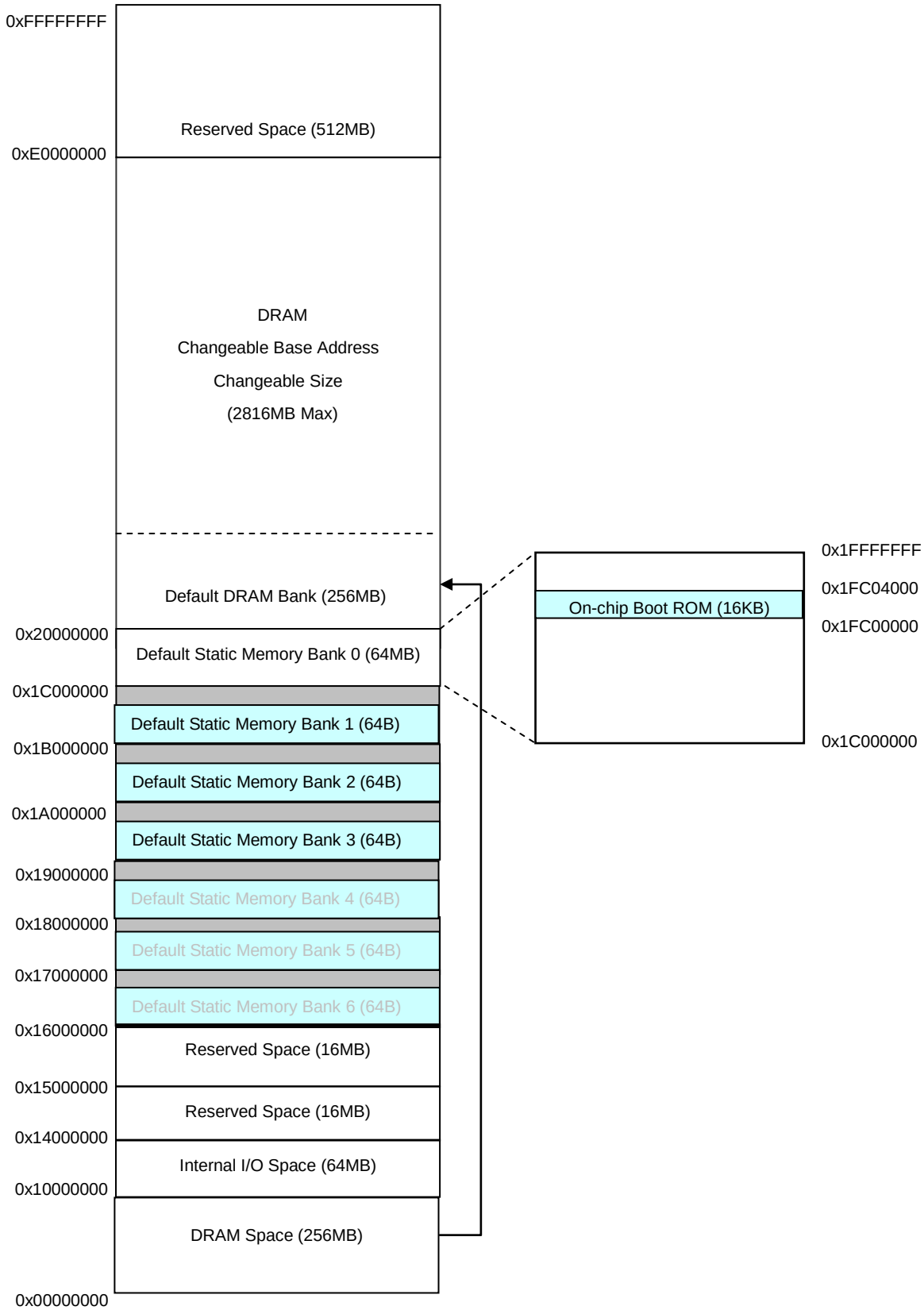


Figure 14-1 Physical Address Space Map

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 14-2 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	Static Memory, CS6#	64B
0x16000040	0x16FFFFFF	Reserved	
0x17000000	0x1700003F	Static Memory, CS5#	64B
0x17000040	0x17FFFFFF	Reserved	
0x18000000	0x1800003F	Static Memory, CS4#	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 (16kB)	16KB
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 14-3 Default Configuration of NEMC 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 The 16KB address space from H'1FC00000 to H'1FC03FFF in bank 0 is mapped to on-chip boot ROM. The other memory spaces in bank 0 are not used.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.4 Static Memory Interface

NEMC provides a glueless interface to normal static memory which don't need byte control like SRAM, memory interface IO devices, etc.. It can directly control up to 3 devices using six chip select lines. Additional devices may be supported through external decoding of the address bus.

Each chip select can directly access memory or IO devices that are 8/16-bits wide. Each device connected to a chip select line has 2 associated registers that control its operation and the access timing to the external device. The Static Memory Control Register SMCRn specifies various configurations for the device. The Static Memory Address Configuration Register SACRn specifies the base address and size for each device, enabling any device to be located anywhere in the physical address range.

The static memory interface includes the following signals:

- Three chip selects, CS3~1#
- Six address signals, SA5-SA0
- One read enable, RD#
- One write enable, WE#
- One wait pin, WAIT#

The SMT field in SMCRn registers specifies the type of memory and BW field specifies the bus width. BOOT_SEL[1:0] pin defines whether system boot from Nor or Nand flash and the page size when boot from Nand flash.

14.4.1 Register Description

Table 14-4 Static Memory Interface Registers

Name	Description	RW	Reset Value	Address	Access Width
SMCR1	Static memory control register 1	RW	0x0FFF7700	0x13410014	32
SMCR2	Static memory control register 2	RW	0x0FFF7700	0x13410018	32
SMCR3	Static memory control register 3	RW	0x0FFF7700	0x1341001C	32
SMCR4	Static memory control register 4	RW	0x0FFF7700	0x13410020	32
SMCR5	Static memory control register 5	RW	0x0FFF7700	0x13410024	32
SMCR6	Static memory control register 6	RW	0x0FFF7700	0x13410028	32
SACR1	Static memory bank 1 address configuration register	RW	0x00001AFE	0x13410034	32
SACR2	Static memory bank 2 address configuration register	RW	0x000018FE	0x13410038	32
SACR3	Static memory bank 3 address configuration register	RW	0x000017FF	0x1341003C	32
SACR4	Static memory bank 4 address configuration register	RW	0x000016FF	0x13410040	32
SACR5	Static memory bank 5 address configuration register	RW	0x000015FF	0x13410044	32
SACR6	Static memory bank 6 address configuration register	RW	0x000014FF	0x13410048	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.4.1.1 Static Memory Control Register (SMCR1~6)

SMCR1~6 are 32-bit read/write registers that contain control bits for static memory. On reset, SMCR1~6 are initialized to 0x0FFF7700.

NOTE. If use the conventional type NAND (i.e. SDR NAND), all of the Command Latch Cycle/Address Latch Cycle/Data input Timing/Data output Timing/Read ID Operation/Read Status Cycle will be configured by SMCRn.

But, if use the high-speed Toggle DDR NAND, only the Command Latch Cycle/Address Latch Cycle will be configured by SMCRn, and Data input Timing/Data output Timing/Read ID Operation/Read Status Cycle are configured by Toggle NAND Control Register(TGCRn).

	SMCR1																0x13410014													
	SMCR2																0x13410018													
	SMCR3																0x1341001C													
	SMCR4																0x13410020													
	SMCR5																0x13410024													
	SMCR6																0x13410028													
Bit	31	30	29	28	27	26	25	24	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		STRV						TAW				TBP				TAH				TAS				BW		Reserved			BL		SMT
RST	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	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	STRV	**Static Memory Recovery Time.** Its value is the number of idle cycles (0~63 cycles) inserted between bus cycles when switching from one bank to another bank or between a read access to a write access in the same bank. Its initial value is 0xF (15 cycles).	RW
23:20	TAW	**Access Wait Time.** For normal memory, these bits specify the number of wait cycles to be inserted in read strobe time. For burst ROM, these bits specify the number of wait cycles to be inserted in first data read strobe time. **TAW3~0 Wait cycle Wait# Pin** 0000 0 cycle Ignored 0001 1 cycle Enabled 0010 2 cycles Enabled 0011 3 cycles Enabled 0100 4 cycles Enabled 0101 5 cycles Enabled	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0110 6 cycles Enabled 0111 7 cycles Enabled 1000 8 cycles Enabled 1001 9 cycles Enabled 1010 10 cycles Enabled 1011 12 cycles Enabled 1100 15 cycles Enabled 1101 20 cycles Enabled 1110 25 cycles Enabled 1111 31 cycles Enabled (Initial Value)	
19:16	TBP	**Burst Pitch Time.** For burst ROM, these bits specify the number of wait cycles to be inserted in subsequent access. For normal memory, these bits specify the number of wait cycles to be inserted in write strobe time. **TBP3~0 Wait cycle Wait# Pin** 0000 0 cycle Ignored 0001 1 cycle Enabled 0010 2 cycles Enabled 0011 3 cycles Enabled 0100 4 cycles Enabled 0101 5 cycles Enabled 0110 6 cycles Enabled 0111 7 cycles Enabled 1000 8 cycles Enabled 1001 9 cycles Enabled 1010 10 cycles Enabled 1011 12 cycles Enabled 1100 15 cycles Enabled 1101 20 cycles Enabled 1110 25 cycles Enabled 1111 31 cycles Enabled (Initial Value)	RW
15:12	TAH	**Address Hold Time.** These bits specify the number of wait cycles to be inserted from negation of read/write strobe to address. **TAH2~0 Wait cycle** 0000 0 cycle 0001 1 cycle 0010 2 cycles 0011 3 cycles 0100 4 cycles 0101 5 cycles 0110 6 cycles 0111 7 cycles (Initial Value)	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1000 8 cycles 1001 9 cycles 1010 10 cycles 1011 11 cycles 1100 12 cycles 1101 13 cycles 1110 14 cycles 1111 15 cycles	
11:8	TAS	**Address Setup Time.** These bits specify the number of wait cycles (0~15 cycles) to be inserted from address to assertion of read/write strobe. **TAS2~0 Wait cycle** 000 0 cycle 001 1 cycle 010 2 cycles 011 3 cycles 100 4 cycles 101 5 cycles 110 6 cycles 111 7 cycles (Initial Value) 1000 8 cycles 1001 9 cycles 1010 10 cycles 1011 11 cycles 1100 12 cycles 1101 13 cycles 1110 14 cycles 1111 15 cycles	RW
7:6	BW	**Bus Width.** These bits specify the bus width. This filed is writeable and are initialized to 0 by a reset. **BW1~0 Bus Width** 00 8 bits (Initial Value) 01 Reserved 10 Reserved 11 Reserved	RW
5:3	Reserved	Writing has no effect, read as zero. **NOTE:** Don't write Bit3 to 1.	R
2:1	BL	**Burst Length (BL1, BL0).** When Burst ROM is connected; these bits specify the number of burst in an access. These bits are only valid when SMT is set to 1. **BL1~0 Burst Length** 00 4 consecutive accesses. Can be used with 8-bit bus	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		width (Initial Value) 01 8 consecutive accesses. Can be used with 8-bit bus width 10 16 consecutive accesses. Can only be used with 8-bit bus width 11 32 consecutive accesses. Can only be used with 8-bit bus width	
0	SMT	**Static Memory Type (SMT).** This bit specifies the type of static memory. **SMT** **Description** 0 Normal Memory (Initial Value) 1 Burst ROM	RW

14.4.1.2 Static Bank Address Configuration Register (SACR1~6)

SACR1~6 defines the physical address for static memory bank 1 to 6, 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.

	SACR1																													**0x13410034**
	SACR2																													**0x13410038**
	SACR3																													**0x1341003C**
	SACR4																													**0x13410040**
	SACR5																													**0x13410044**
	SACR6																													**0x13410048**
Bit	31	30	29	28	27	26	25	24	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							
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
RST4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1	1	1	1
RST5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1
RST6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:16	Reserved	Writing has no effect, read as zero.	R
15:8	BASE	**Address Base:** Defines the base address of Static Bank n (n = 1 to 6). The initial values are: SACR1.BASE 0x1B SACR2.BASE 0x1A SACR3.BASE 0x19 SACR4.BASE 0x18 SACR5.BASE 0x17 SACR6.BASE 0x16	RW
23:20	MASK	**Address Mask**: Defines the mask of Static Bank n (n = 1 to 6). The initial values are: SACR1.MASK 0xFF SACR2.MASK 0xFF SACR3.MASK 0xFF SACR4.MASK 0xFF SACR5.MASK 0xFF SACR6.MASK 0xFF	RW

14.4.2 Example of Connection

Following figures shows examples of connection to 8-bit/16-bit data width normal memory.

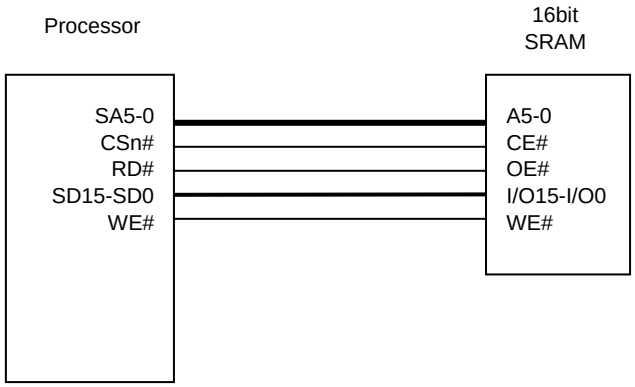


Figure 14-2 Example of 16-Bit Data Width SRAM Connection

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

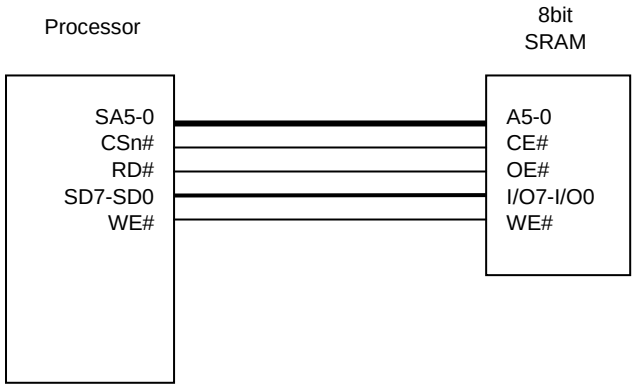


Figure 14-3 Example of 8-Bit Data Width SRAM Connection

14.4.3 Basic Interface

When SMT field in SMCRn (n = 1 to 6) is 0, normal memory (non-burst ROM, Flash, normal SRAM or memory-like device) is connected to bank n. When bank n (n = 1 to 6) is accessed, CSn# is asserted as soon as address is output. In addition, the RD# signal, which can be used as OE#, and write control signals WE# is asserted.

The TAS field in SMCRn is the latency from CSn# to read/write strobe. The TAW3 field is the delay time of RD# in read access. TBP3~0 field is the delay time of WE# and WEn# in write access. In addition, any number of waits can be inserted by means of the external pin (WAIT#). The TAH field is the latency from RD# and WEn# negation to CSn# negation, also the hold time to address and write data.

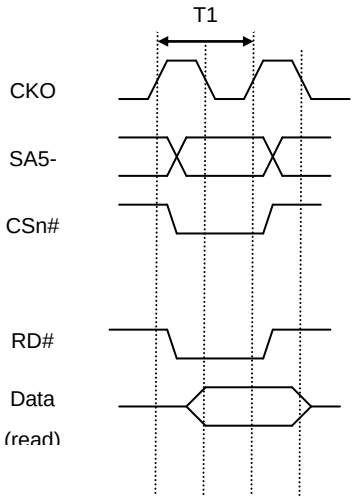
All kinds of normal memories (non-burst ROM, normal SRAM and Flash) have the same read and write timing. There are some requirements for writes to flash memory. Flash memory space must be un-cacheable and un-buffered. Writes must be exactly the width of the populated Flash devices on the data bus (no byte writes or word writes to a 16-bit bus, and so on). Software is responsible for partitioning commands and data, and writing them out to Flash in the appropriate sequence.

Glossary

Th – hold cycle
Tw – wait cycle
Ts – setup cycle
T1 – read inherent cycle or first write inherent cycle
T2 – last write inherent cycle
Tb – burst read inherent cycle

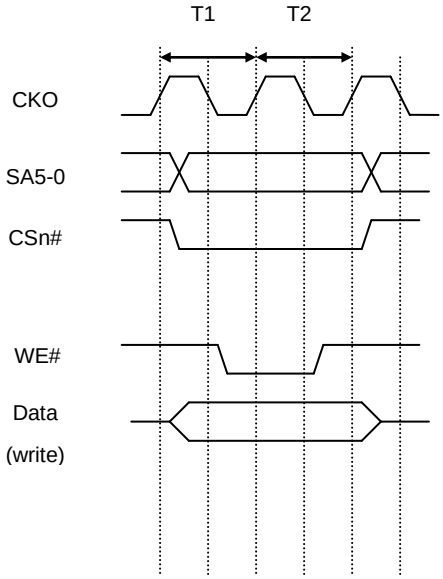
Following figures show the timing of normal memory. A no-wait read access is completed in one cycle and a no-wait write access is completed in two cycles. Therefore, there is no negation period in case of access at minimum pitch.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



*In this example, SMCRn:MT = 0, TAS = 0, TAW = 0, TAH = 0

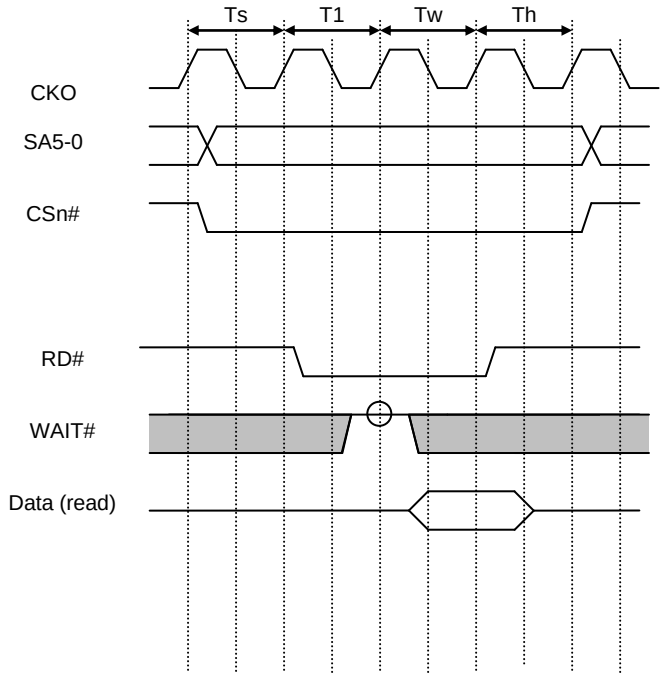
Figure 14-4 Basic Timing of Normal Memory Read



*In this example, SMCRn: SMT = 0, TAS = 0, TBP = 0, TAH = 0

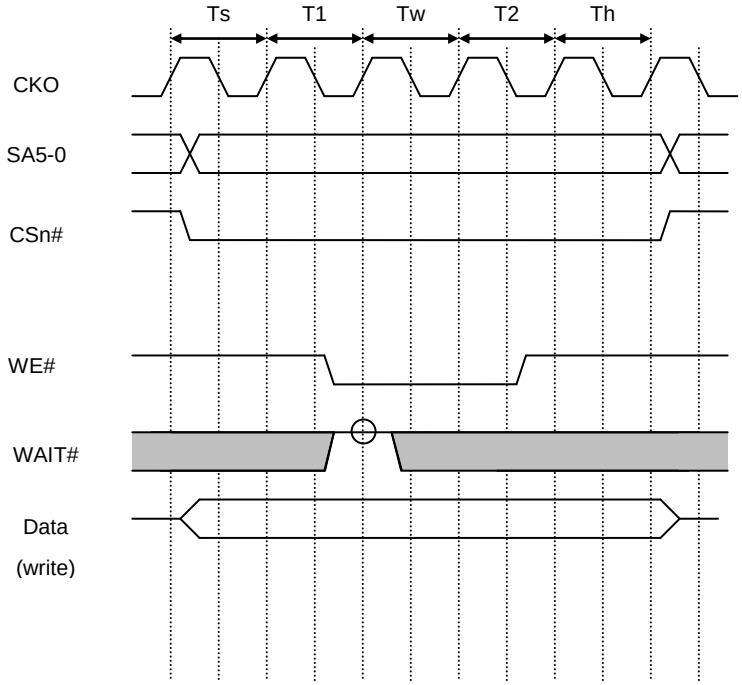
Figure 14-5 Basic Timing of Normal Memory Write

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



*In this example, SMCRn: SMT = 0, TAS = 1, TAW = 1, TAH = 1

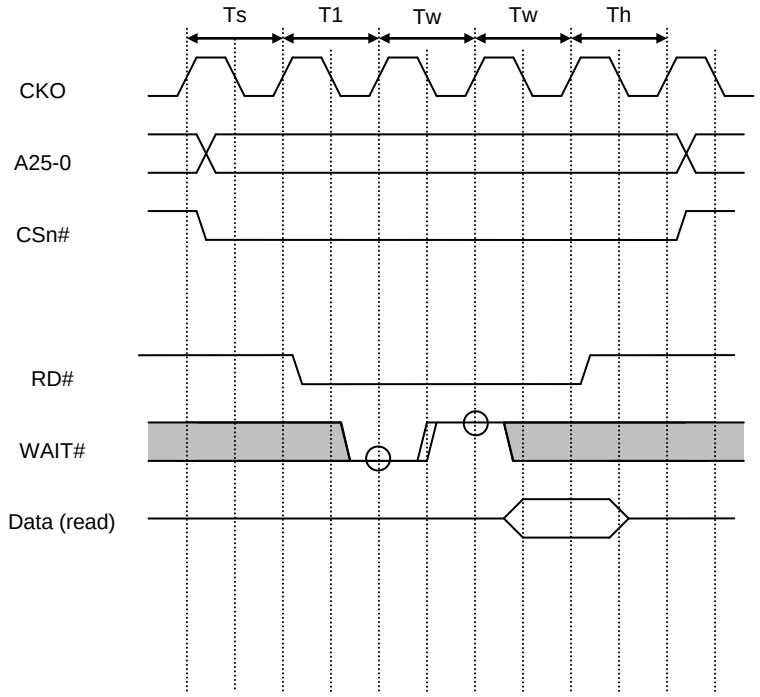
Figure 14-6 Normal Memory Read Timing With Wait (Software Wait Only)



*In this example, SMCRn: SMT = 0, TAS = 1, TBP = 1, TAH = 1

Figure 14-7 Normal Memory Write Timing With Wait (Software Wait Only)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



*In this example, SMCRn: SMT = 0, TAS = 1, TAW = 1, TAH=1

Figure 14-8 Normal Memory Read Timing With Wait (Wait Cycle Insertion by WAIT# pin)

14.4.4 Burst ROM Interface

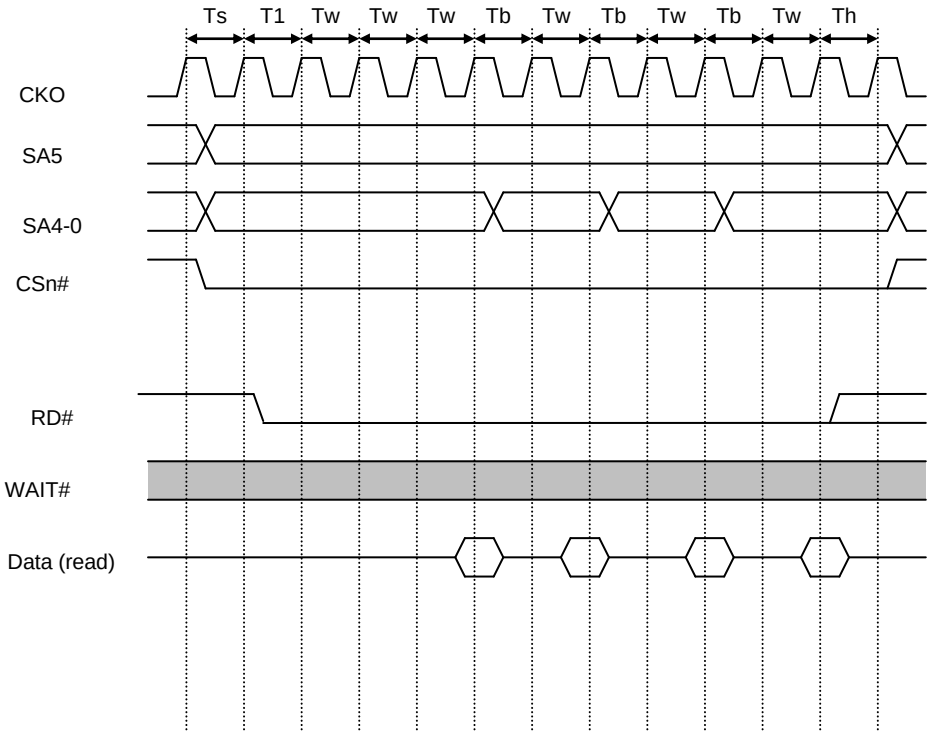
Setting SMT to 1 in SMCRn allows burst ROM to be connected to bank n (n = 1 to 6). The burst ROM interface provides high-speed access to ROM that has a nibble access function. Basically, access is performed in the same way as for normal memory, but when the first cycle ends, only the address is changed before the next access is executed. When 8-bit burst ROM is connected, the number of consecutive accesses can be set as 4, 8, 16, or 32 with bits BL1~0. When 16-bit ROM is connected, 4, 8, or 16 can be set in the same way.

For burst ROM read, TAW sets the delay time from read strobe to the first data, TBP sets the delay time from consecutive address to data. Burst ROM writes have the same timing as normal memory except TAW instead of TBP is used to set the delay time of write strobe.

WAIT# pin sampling is always performed when one or more wait states are set.

Following figures show the timing of burst ROM.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



*In this example, SMT = 1, BL = 0, TAS = 1, TAW = 3,TBP = 1,TAH = 1

Figure 14-9 Burst ROM Read Timing (Software Wait Only)

14.5 NAND Flash Interface

NAND flash can be connected to static memory bank 6~ band 1. Both 8-bit and 16-bit NAND flashes (Toggle NAND or conventional NAND) are supported. A mechanism for booting from NAND flash is also supported.

14.5.1 Register Description

Table 14-5 NAND Flash Interface Registers

Name	Description	RW	Reset Value	Address	Access Width
NFCSR	NAND flash control/status register	RW	0x00000000	0x13410050	32
PNCR	NAND PN control register	RW	0x00000000	0x13410100	32
PNDR	NAND PN data register	RW	0x00005AA5	0x13410104	32
BITCNT	NAND bit counter	R	0x00000000	0x13410108	32
TGWE	Toggle NAND Data Write Access	RW	0x00010000	0x1341010C	32
TGCR1	Toggle NAND Control Register 1	RW	0x0FFF770A	0x13410110	32
TGCR2	Toggle NAND Control Register 2	RW	0x0FFF770A	0x13410114	32
TGCR3	Toggle NAND Control Register 3	RW	0x0FFF770A	0x13410118	32
TGCR4	Toggle NAND Control Register 4	RW	0x0FFF770A	0x1341011C	32
TGCR5	Toggle NAND Control Register 5	RW	0x0FFF770A	0x13410120	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

TGCR6	Toggle NAND Control Register 6	RW	0x0FFF770A	0x13410124	32
TGSR	Toggle NAND RD# to DQS and DQ Delay Register	RW	0x14A5294A	0x13410128	32
TGFL	Toggle NAND ALE Fall to DQS Rise (bank 1/2/3)	RW	0x00FFFFFF	0x1341012C	32
TGFH	Toggle NAND ALE Fall to DQS Rise (bank 4/5/6)	RW	0x00FFFFFF	0x13410130	32
TGCL	Toggle NAND CLE to RD# Low (bank 1/2/3)	RW	0x00646464	0x13410134	32
TGCH	Toggle NAND CLE to RD# Low (bank 4/5/6)	RW	0x00646464	0x13410138	32
TGPD	Toggle NAND Data Postamble Hold Time Done	R	0x0000003F	0x1341013C	32
TGSL	Toggle NAND DQS Setup Time for Data Input Start (bank 1/2/3)	RW	0x00252525	0x13410140	32
TGSH	Toggle NAND DQS Setup Time for Data Input Start (bank 4/5/6)	RW	0x00252525	0x13410144	32
TGRR	Toggle NAND Timer for Random Data Input and Register Read Out	RW	0x00010000	0x13410148	32
TGDR	Toggle NAND DQS Delay Control Register	RW	0x03000016	0x1341014C	32
TGHL	Toggle NAND DQS Hold Time for Data Input Finish(bank 1/2/3)	RW	0x01252525	0x13410150	32
TGHH	Toggle NAND DQS Hold Time for Data Input Finish(bank 4/5/6)	RW	0x00252525	0x13410154	32

14.5.1.1 NAND Flash Control/Status Register (NFCSR)

NFCSR is a 32-bit read/write register that is used to configure NAND flash. It is initialized by any reset.

	NFCSR																													0x13410050
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAEC	Reserved									TNFE6	TNFE5	TNFE4	TNFE3	TNFE2	TNFE1	Reserved				NFCE6	NFE6	NFCE5	NFE5	NFCE4	NFE4	NFCE3	NFE3	NFCE2	NFE2	NFCE1	NFE1
RST	0	0	0	0	0	0	0	0	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	DAEC	**Toggle NAND Data Access Enabel Clear:** write 1 to clear TGWE.DAE.	W1
30:22	Reserved	Writing has no effect, read as zero.	R
21:16	TNFEn (n=1,2,3, 4,5,6)	**Toggle NAND Flash Enable:** Specifies wither Toggle NAND flash or conventional NAND flash is connected to static bank n. When system is configured to boot from Toggle NAND flash, this bit is initialized to 1. **TNFE** **Description** 0 Static bank n is used as conventional NAND flash 1 Static bank n is used as Toggle NAND flash	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:12	Reserved	Writing has no effect, read as zero.	R
1/3/5/7/9/11	FCEn (n=1,2,3,4,5,6)	**NAND Flash FCE# Assertion Control:** Controls the assertion of NAND Flash FCEn#. When set, FCEn# is always asserted until this bit is cleared. When the NAND flash (Toggle NAND flash or conventional NAND flash) require FCEn# to be asserted during read busy time, this bit should be set. NOTE: Before set FCEn to 1, DPHTDn MUST be checked by software. When the last Toggle NAND data write or read access done, software MUST clear the relevant FCEn. **FCE** **Description** 0 FCEn# is asserted as normal static chip enable(Initial value) 1 FCEn# is always asserted	RW
0/2/4/6/8/10	NFEn (n=1,2,3,4,5,6)	**NAND Flash Enable:** Specifies if NAND flash (Toggle NAND flash or conventional NAND flash) is connected to static bank n. When system is configured to boot from NAND flash, this bit is initialized to 1. **NFE** **Description** 0 Static bank n is not used as NAND flash 1 Static bank n is used as NAND flash	RW

14.5.1.2 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																												0x13410100					
Bit	31	30	29	28	27	26	25	24	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																										BIT_RST	BIT_SEL	BIT_EN	Reserved	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

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5	BIT_RST	**NAND 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
4	BIT_SEL	**NAND 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
3	BIT_EN	**NAND BIT Counter Enable:** Enable/disable bit counter counting.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		BIT_EN Description 0 Bit counting is disabled 1 Bit counting is enabled	
2	Reserved	Writing has no effect, read as zero.	R
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

14.5.1.3 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																															0x13410104
Bit	31	30	29	28	27	26	25	24	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

14.5.1.4 NAND Bit Counter (BITCNT)

BITCNT is a 32-bit read/write 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.

	BITCNT																															0x13410108
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BITCNT																															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

14.5.1.5 Toggle NAND Data Write Access (TGWE)

DQS Setup Time for Data Write Start (TGSL.CDQSS and TGSH.CDQSS) is defined from the last data input condition of the control signals such as CE, CLE and ALE. Before any Toggle NAND data write access, software MUST write 1 to corresponding SDEn, and WCD shall be checked prior to a write data access.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

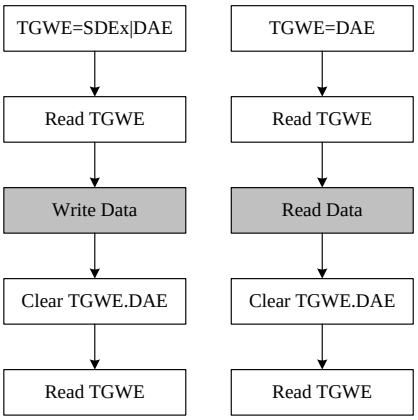


Figure 14-10 Toggle NAND Page Write/Read Operation

	TGWE																0x1341010C													
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAE	FIDLE	Reserved													WCD	Reserved										SDE6	SDE5	SDE4	SDE3	SDE2	SDE1
RST	0	1	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

Bits	Name	Description	RW
31	DAE	**Toggle NAND Data Access Enabel:** Before Data Access (Data Write & Data Read) this bit MUST set to 1, and after Data Access DAE MUST clear by software (before next Command or Address cycle). **NOTE.** DAE can be cleared by set **NFCSR.DAEC**=1.	WR
30	FIDLE	**Internal FSM IDLE**	R
29:17	Reserved	Writing has no effect, read as zero.	R
16	WCD	**DQS Setup Time for data input start Done:** After set SDEn, this bit will be low CDQSS cycles. When WCD return to 1 data can be write to Toggle NAND. **WCD Description** 0 DQS Setup Time for Data Write Start not pass 1 DQS Setup Time for Data Input Start Pass	R
15:6	Reserved	Writing has no effect, read as zero.	R
5	SDE6	**Set DQS output enable (bank6)**: Before data write access, software MUST write 1 to SDE6. After CDQSS cycles, controller will goes into data-write-mode.	W
4	SDE5	**Set DQS output enable (bank5)**: Before data write access, software MUST write 1 to SDE5. After CDQSS cycles, controller will goes into data-write-mode.	W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3	SDE4	**Set DQS output enable (bank4)**: Before data write access, software MUST write 1 to SDE4. After CDQSS cycles, controller will goes into data-write-mode.	W
2	SDE3	**Set DQS output enable (bank3)**: Before data write access, software MUST write 1 to SDE3. After CDQSS cycles, controller will goes into data-write-mode.	W
1	SDE2	**Set DQS output enable (bank2)**: Before data write access, software MUST write 1 to SDE2. After CDQSS cycles, controller will goes into data-write-mode.	W
0	SDE1	**Set DQS output enable (bank1)**: Before data write access, software MUST write 1 to SDE1. After CDQSS cycles, controller will goes into data-write-mode.	W

14.5.1.6 Toggle NAND Control Register (TGCR1~6)

TGCR1~6 are 32-bit read/write registers that contain control bits for Toggle NAND flash. On reset, TGCR1~6 are initialized to 0x0FFF770A.

NOTE:

(1) If the conventional type NAND (i.e. SDR NAND) enable, all of the Command Latch Cycle/Address Latch Cycle/Data input Timing/Data output Timing/Read ID Operation/Read Status Cycle will be configured by SMCRn. And Toggle NAND Control Register (TGCRn) will be ignored.

(2) However, if the high-speed Toggle DDR NAND enable, only the Command Latch Cycle/Address Latch Cycle are configured by SMCRn, but the Data input Timing/Data output Timing/Read ID Operation/Read Status Cycle are configured by Toggle NAND Control Register(TGCRn).

	TGCR1 **0x13410110** **TGCR2** **0x13410114** **TGCR3** **0x13410118** **TGCR4** **0x1341011C** **TGCR5** **0x13410120** **TGCR6** **0x13410124**																													
Bit	31	30	29	28	27	26	25	24	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		TRV						TRW				TWW				TAH				TAS				Reserved				DPHT			
RST	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	0	0	0	1	0	1	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:24	TRV	**Toggle NAND Flash Recovery Time.** Its value is the number of idle cycles (0~63 cycles) inserted between bus cycles when switching from one bank to another bank or between a read	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		access to a write access in the same bank. Its initial value is 0xF (15 cycles).	
23:20	TRW	**Toggle NAND Flash Read Wait Time.** These bits specify the number of wait cycles to be inserted in read strobe time.	RW
19:16	TWW	**Toggle NAND Flash Write Wait Time.** These bits specify the number of wait cycles to be inserted in write strobe time.	RW
15:12	TAH	**Toggle NAND Flash Address Hold Time.** These bits specify the number of wait cycles to be inserted from negation of read/write strobe to address.AHB cycles=TAH+1	RW
11:8	TAS	**Toggle NAND Flash Address Setup Time.** These bits specify the number of wait cycles (1~15 cycles) to be inserted from address to assertion of read/write strobe. **NOTE. TAS can NOT be 0.AHB cycles=TAS+1**	RW
7:4	Reserved	Writing has no effect, read as zero.	R
3:0	DPHT	**Toggle NAND Flash Data Postamble Hold Time.** CE# is once deasserted, it shall stay high over at least DPHT cycles before it is asserted again. Therefore, software MUST wait corresponding DPHTDn return to 1 before set FCEn. **DPHT** **Wait cycle** **Wait# Pin** 0010 2 cycles Enabled 0011 3 cycles Enabled 0100 4 cycles Enabled 0101 5 cycles Enabled 0110 6 cycles Enabled 0111 7 cycles Enabled 1000 8 cycles Enabled 1001 9 cycles Enabled 1010 10 cycles Enabled(Initial Value) 1011 12 cycles Enabled 1100 15 cycles Enabled 1101 20 cycles Enabled 1110 25 cycles Enabled 1111 31 cycles Enabled **NOTE: DPHT MUST >= 2.**	RW

14.5.1.7 Toggle NAND RD# to DQS and DQ delay Register (TGSR)

TGSR is 32-bit read/write registers that specify the number of delay cycles (0~31 cycles) from RD# to DQS for Toggle NAND flash. On reset, TGSR is initialized to 0x14A5294A. See Figure 14-12.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TGSR																														0x13410128
Bit	31	30	29	28	27	26	25	24	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		DQSRE5					DQSRE5					DQSRE4				DQSRE3						DQSRE2					DQSRE1				
RST	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	1	0	0	1	0	1	0

Bits	Name	Description	RW
31:30	Reserved	Writing has no effect, read as zero.	R
29:25	DQSRE6	**Toggle NAND Flash RD# to DQS and DQ delay (Bank6).** 10 cycles is the initial value.	RW
24:20	DQSRE5	**Toggle NAND Flash RD# to DQS and DQ delay (Bank5).** 10 cycles is the initial value.	RW
19:15	DQSRE4	**Toggle NAND Flash RD# to DQS and DQ delay (Bank4).** 10 cycles is the initial value.	RW
14:10	DQSRE3	**Toggle NAND Flash RD# to DQS and DQ delay (Bank3).** 10 cycles is the initial value.	RW
9:5	DQSRE2	**Toggle NAND Flash RD# to DQS and DQ delay (Bank2).** 10 cycles is the initial value.	RW
4:0	DQSRE1	**Toggle NAND Flash RD# to DQS and DQ delay (Bank1).** 10 cycles is the initial value.	RW

14.5.1.8 Toggle NAND ALE Fall to DQS Rise (bank 1/2/3 TGFL)

TGFL is 32-bit read/write registers that specify the number of wait cycles (0~255 cycles) from ALE fall to DQS rise for bank 1/2/3. On reset, TGFL is initialized to 0x00FFFFFF.

	TGFL																														0x1341012C
Bit	31	30	29	28	27	26	25	24	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								FDA3								FDA2								FDA1							
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
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	FDA 3	**Toggle NAND Flash ALE Fall to DQS Rise (Bank3).** 255 cycles is the initial value.	RW
15:8	FDA 2	**Toggle NAND Flash ALE Fall to DQS Rise (Bank2).** 255 cycles is the initial value.	RW
7:0	FDA 1	**Toggle NAND Flash ALE Fall to DQS Rise (Bank1).** 255 cycles is the initial value.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.5.1.9 Toggle NAND ALE Fall to DQS Rise (bank 4/5/6 TGFH)

TGFH is 32-bit read/write registers that specify the number of wait cycles (0~255 cycles) from ALE fall to DQS rise for bank 4/5/6. On reset, TGFH is initialized to 0x00FFFFFF.

	TGFH																0x13410130													
Bit	31	30	29	28	27	26	25	24	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								FDA6								FDA5								FDA4							
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
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	FDA 6	**Toggle NAND Flash First ALE Fall to DQS Rise (Bank6).** 255 cycles is the initial value.	RW
15:8	FDA 5	**Toggle NAND Flash First ALE Fall to DQS Rise (Bank5).** 255 cycles is the initial value.	RW
7:0	FDA 4	**Toggle NAND Flash First ALE Fall to DQS Rise (Bank4).** 255 cycles is the initial value.	RW

14.5.1.10 Toggle NAND CLE to RD# Low (bank 1/2/3 TGCL)

TGCL is 32-bit read/write registers that specify the number of wait cycles (0~255 cycles) from CLE to RD# low for bank 1/2/3. On reset, TGCL is initialized to 0x00646464.

	TGCL																0x13410134													
Bit	31	30	29	28	27	26	25	24	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								CLR3								CLR2								CLR1							
RST	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CLR 3	**Toggle NAND Flash CLE to RE_n Low (Bank3).** 100 cycles is the initial value. NOTE. CLR MUST smaller than FDA3 , and >=2.	RW
15:8	CLR 2	**Toggle NAND Flash CLE to RE_n Low (Bank2).** 100 cycles is the initial value.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		NOTE. CLR MUST smaller than FDA2, and >=2.	
7:0	CLR 1	**Toggle NAND Flash CLE to RE_n Low (Bank1).** 100 cycles is the initial value. NOTE. CLR MUST smaller than FDA1, and >=2.	RW

14.5.1.11 Toggle NAND CLE to RD# low (bank 4/5/6 TGCH)

TGCH is 32-bit read/write registers that specify the number of wait cycles (0~255 cycles) from CLE to RD# low for bank 4/5/6. On reset, TGCH is initialized to 0x00646464.

	TGCH																0x13410138													
Bit	31	30	29	28	27	26	25	24	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								CLR6								CLR5								CLR4							
RST	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CLR 6	**Toggle NAND Flash CLE to RE_n Low (Bank3).** 100 cycles is the initial value. NOTE. CLR MUST smaller than FDA6, and >=2.	RW
15:8	CLR 5	**Toggle NAND Flash CLE to RE_n Low (Bank2).** 100 cycles is the initial value. NOTE. CLR MUST smaller than FDA5, and >=2.	RW
7:0	CLR 4	**Toggle NAND Flash CLE to RE_n Low (Bank1).** 100 cycles is the initial value. NOTE. CLR MUST smaller than FDA4, and >=2.	RW

14.5.1.12 Toggle NAND Data Postamble Hold Time Done (TGPD)

If CE# is once deasserted, it shall stay high over at least DPHT cycles before it is asserted again. Therefore, DPHTDn shall be checked prior to any chip re-enable (i.e. set FCEn).

	TGPD																										0x1341013C					
Bit	31	30	29	28	27	26	25	24	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																										DPHTD6	DPHTD5	DPHTD4	DPHTD3	DPHTD2	DPHTD1
RST	0	0	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:6	Reserved	Writing has no effect, read as zero.	R
5	DPHTD6	**Toggle NAND Flash Data Postamble Hold Time Done (Bank6).** **DPHTD6 Description** 0 Can NOT enable CE6# 1 Data Postamble Hold Time Done, FCE6 can be set to 1	R
4	DPHTD5	**Toggle NAND Flash Data Postamble Hold Time Done (Bank5).** **DPHTD5 Description** 0 Can NOT enable CE5# 1 Data Postamble Hold Time Done, FCE5 can be set to 1	R
3	DPHTD4	**Toggle NAND Flash Data Postamble Hold Time Done (Bank4).** **DPHTD4 Description** 0 Can NOT enable CE4# 1 Data Postamble Hold Time Done, FCE4 can be set to 1	R
2	DPHTD3	**Toggle NAND Flash Data Postamble Hold Time Done (Bank3).** **DPHTD3 Description** 0 Can NOT enable CE3# 1 Data Postamble Hold Time Done, FCE3 can be set to 1	R
1	DPHTD2	**Toggle NAND Flash Data Postamble Hold Time Done (Bank2).** **DPHTD2 Description** 0 Can NOT enable CE2# 1 Data Postamble Hold Time Done, FCE2 can be set to 1	R
0	DPHTD1	**Toggle NAND Flash Data Postamble Hold Time Done (Bank1).** **DPHTD1 Description** 0 Can NOT enable CE1# 1 Data Postamble Hold Time Done, FCE1 can be set to 1	R

14.5.1.13 Toggle NAND DQS Setup Time for Data Input Start (bank 1/2/3 TGSL)

DQS Setup Time for Data Write Start (i.e. TGSL.CDQSS) is defined from the last data input condition of the control signals such as CE, CLE and ALE.

	TGSL																													0x13410140	
Bit	31	30	29	28	27	26	25	24	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								CDQSS 3								CDQSS 2								CDQSS 1							
RST	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CDQSS 3	**DQS Setup Time for data input start (bank3).** (i.e. ale_fall – dqs_rise < tCDQSS3 and ale_fall – dqs_fall < tCDQSS3)	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15:8	CDQSS 2	**DQS Setup Time for data input start (bank2).** (i.e. ale_fall – dqs_rise < tCDQSS2 and ale_fall – dqs_fall < tCDQSS2)	RW
7:0	CDQSS 1	**DQS Setup Time for data input start (bank1).** (i.e. ale_fall – dqs_rise < tCDQSS1 and ale_fall – dqs_fall < tCDQSS1)	RW

14.5.1.14 Toggle NAND DQS Setup Time for Data Input Start (bank 4/5/6 TGSH)

DQS Setup Time for Data Write Start (TGSH.CDQSS) is defined from the last data input condition of the control signals such as CE, CLE and ALE.

	TGSH																													0x13410144
Bit	31	30	29	28	27	26	25	24	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								CDQSS 6								CDQSS 5								CDQSS 4							
RST	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CDQSS 6	**DQS Setup Time for data input start (bank6).** (i.e. ale_fall – dqs_rise < tCDQSS6 and ale_fall – dqs_fall < tCDQSS6)	RW
15:8	CDQSS 5	**DQS Setup Time for data input start (bank5).** (i.e. ale_fall – dqs_rise < tCDQSS5 and ale_fall – dqs_fall < tCDQSS5)	RW
7:0	CDQSS 4	**DQS Setup Time for data input start (bank4).** (i.e. ale_fall – dqs_rise < tCDQSS4 and ale_fall – dqs_fall < tCDQSS4)	RW

14.5.1.15 Toggle NAND Timer for Random Data Input and Register Read Out (TGRR)

In Random Data Input or Register Read Out mode, TGRR.CWAW specifies the number of wait cycles to be inserted between Command Write Cycle and Address Write Cycle. See 1.5.4 (Program Operation with Random Data Input).

	TGRR																													0x13410148
Bit	31	30	29	28	27	26	25	24	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															TD	Reserved								CWAW							
RST	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	0	0	0	0	0

Bits	Name	Description	RW
31:17	Reserved	Writing has no effect, read as zero.	R
16	TD	**Timer Done:** After set CWAW, this bit will be low CWAW cycles. When	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		TD return to high, address can be write to Toggle NAND. **TD Description** 0 Command Write Cycle to Address Write Cycle not pass 1 Command Write Cycle to Address Write Cycle Pass	
15:8	Reserved	Writing has no effect, read as zero.	R
7:0	CWAW	**Command Write Cycle to Address Write Cycle Time**: In Random Data Input or Register Read Out mode, these bits specify the number of wait cycles to be inserted between Command Write Cycle and Address Write Cycle. Before data write access, software MUST write 1 to SDE1. After CDQSS cycles, controller will goes into data-write-mode.	W

14.5.1.16 Toggle NAND DQS Delay Control Register (TGDR)

TGDR contains an on-chip DLL to control the DQS Delay for read data.

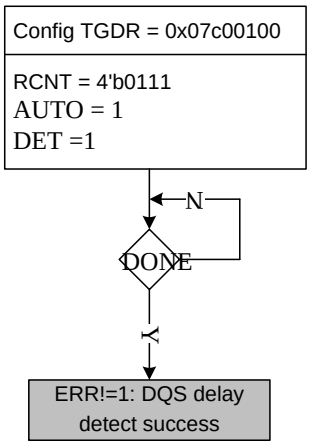


Figure 14-11 Example of DQS Delay Line Auto-Detect-Operation

	TGDR																														0x1341014C
Bit	31	30	29	28	27	26	25	24	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	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	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;	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		ERR is valid only when DONE=1.	
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**: 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, 31].	RW

14.5.1.17 Toggle NAND DQS Hold Time for Data Input Finish (bank 1/2/3 TGHL)

	TGHL																0x13410150													
Bit	31	30	29	28	27	26	25	24	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							CDQSD	CDQSH 3								CDQSH 2								CDQSH 1							
RST	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1

Bits	Name	Description	RW
31:25	Reserved	Writing has no effect, read as zero.	R
24	CDQSD	**DQS Hold Time for data input Finish**	R
23:16	CDQSH 3	**DQS Hold Time for data input (bank3).**	RW
15:8	CDQSH 2	**DQS Hold Time for data input (bank2).**	RW
7:0	CDQSH 1	**DQS Hold Time for data input (bank1).**	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14.5.1.18 Toggle NAND DQS Hold Time for Data Input Finish (bank 4/5/6 TGHH)

	TGHH																														0x13410154
Bit	31	30	29	28	27	26	25	24	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								CDQSH 6								CDQSH 5								CDQSH 4							
RST	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1

Bits	Name	Description	RW
31:24	Reserved	Writing has no effect, read as zero.	R
23:16	CDQSH 6	**DQS Hold Time for data input** (bank6).	RW
15:8	CDQSH 5	**DQS Hold Time for data input** (bank5).	RW
7:0	CDQSH 4	**DQS Hold Time for data input** (bank4).	RW

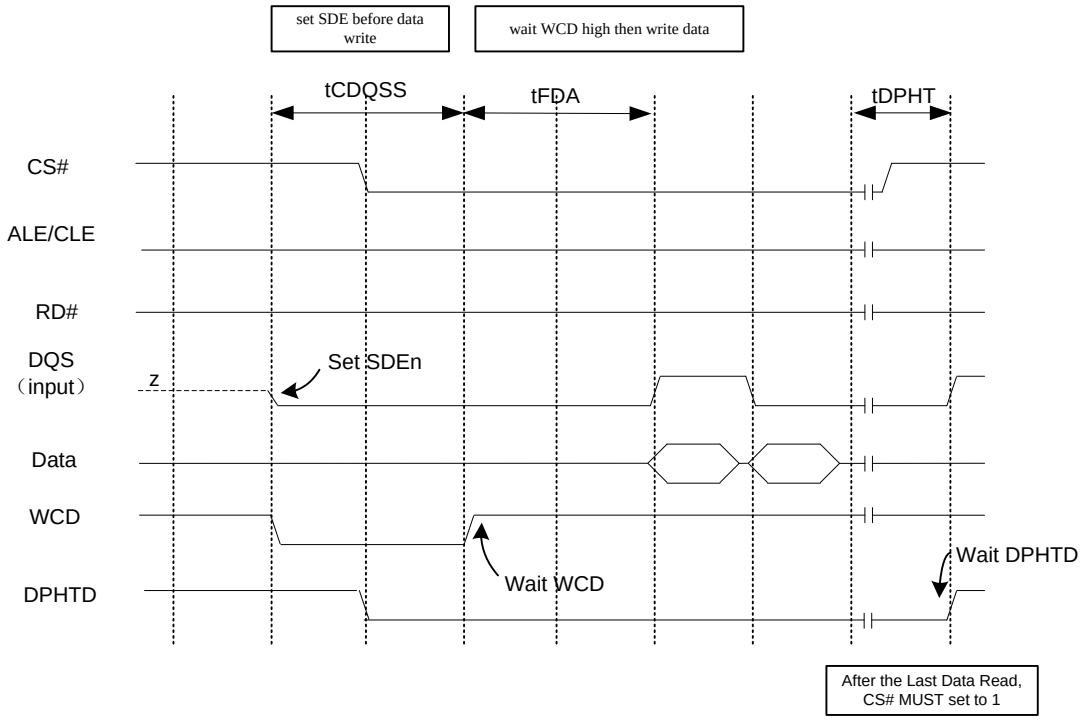


Figure 14-12 Basic Timing of Toggle NAND Write

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

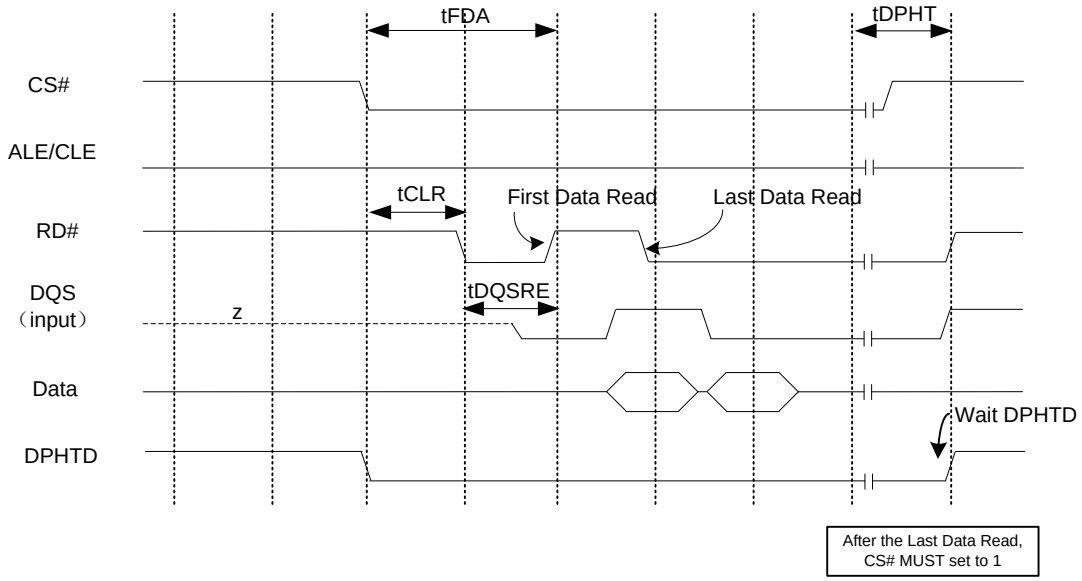


Figure 14-13 Basic Timing of Toggle NAND Read

NOTE.

a) tFDA MUST > tCLR;

b) tDQSRE have nothing to do with tFDA;

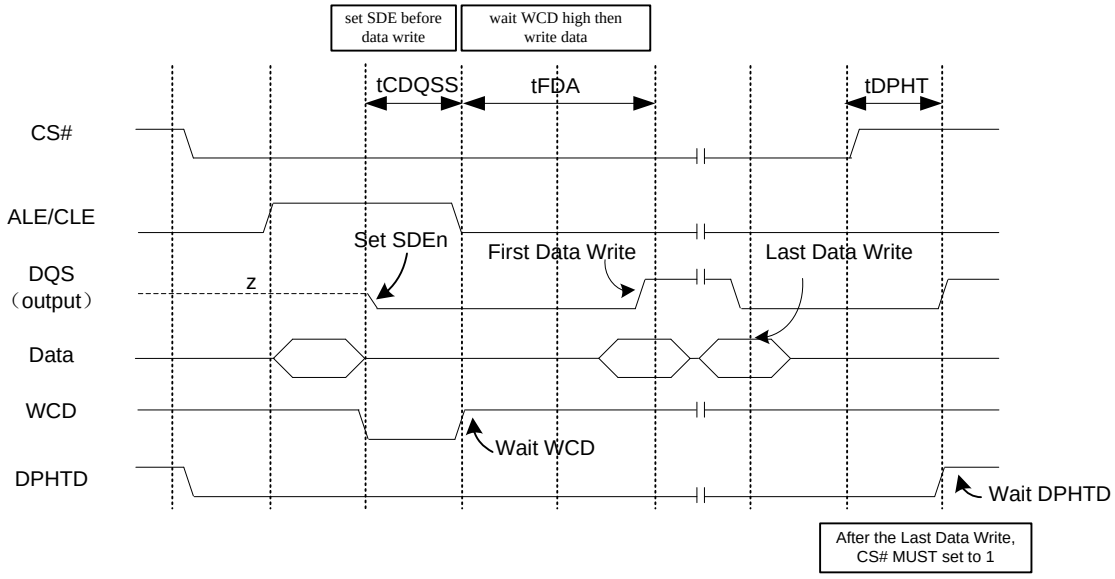


Figure 14-14 Basic Timing of Toggle NAND Page Write

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

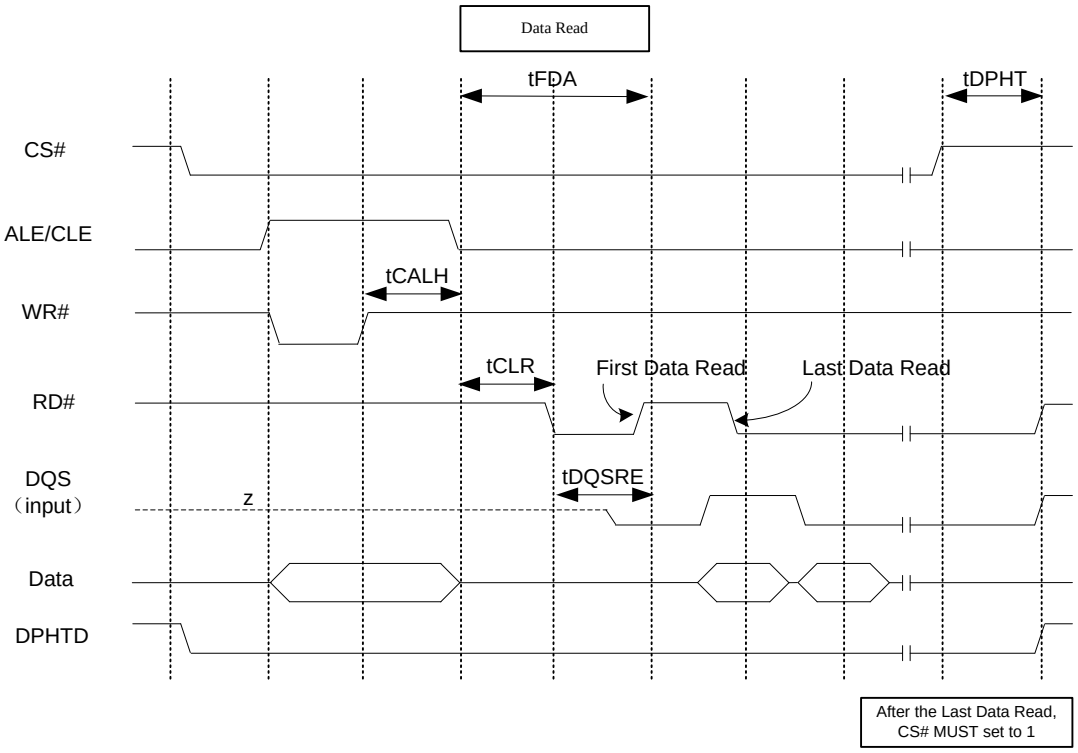


Figure 14-15 Basic Timing of Toggle NAND Page Read

14.5.2 NAND Flash Boot Loader

To support boot from NAND flash, 16KB on-chip Boot ROM is implemented. Following figure illustrates the structure of NAND Flash Boot Loader.

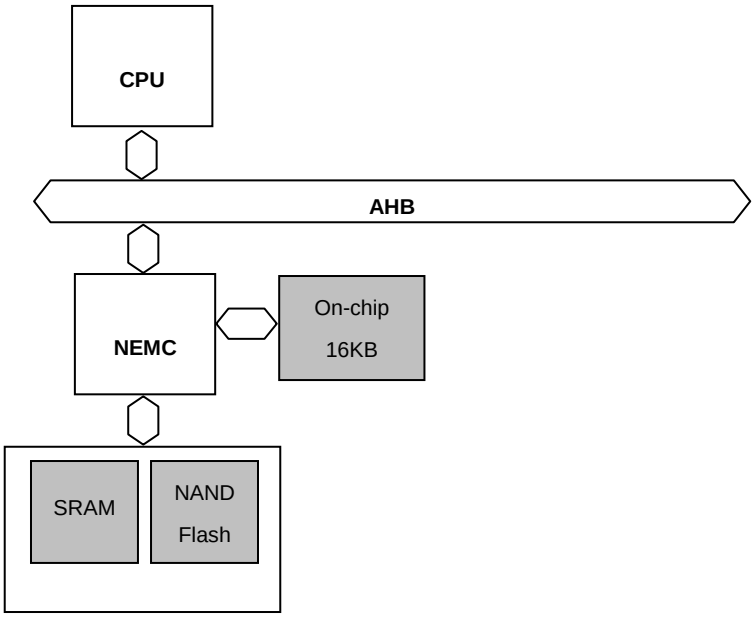


Figure 14-16 Structure of NAND Flash Boot Loader

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

14.5.3 NAND Flash Operation

Set NFEn bit of NAND Flash Control/Status Register (NFCSR) will enable access to NAND flash. 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.

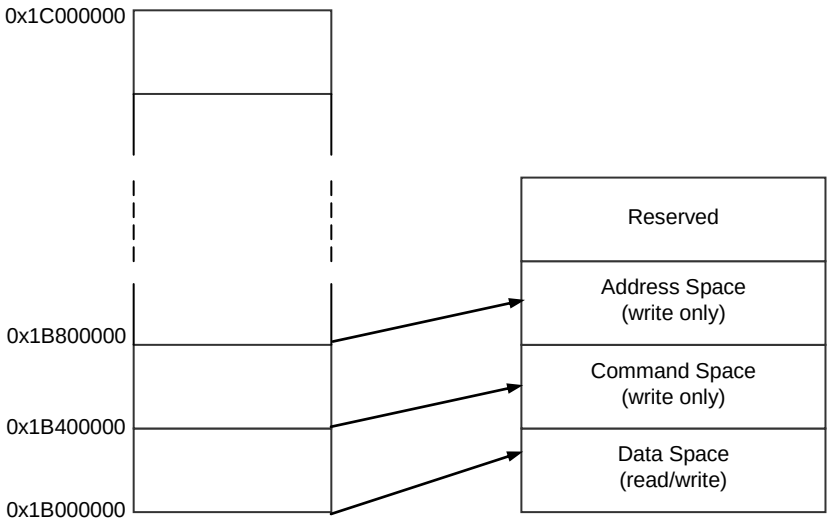


Figure 14-17 Static Bank 1 Partition When NAND Flash is Used (an example)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The timing of NAND flash access is configured by SMCRn and is same as normal static memory timing, except that CSn# is controlled by NFCE bit NFCSR. CSn# is always asserted when NFCE is 1. When NFCE is 0, CSn# is asserted as normal static memory access.

The control signals for direction connection of NAND flash are CSn#, FRE#, FWE#, FRB#(GPIO), A1 and A0. Following figure shows the connection between processor and NAND Flash.

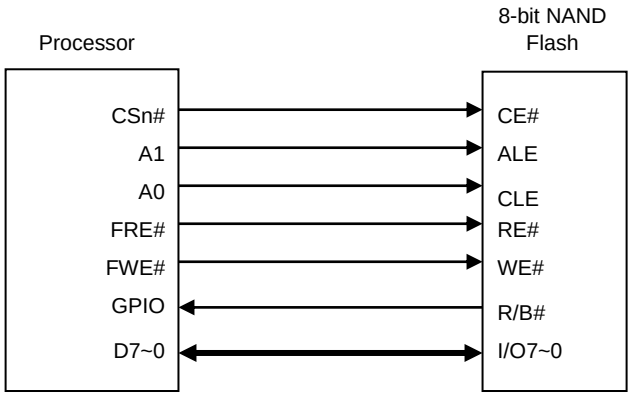


Figure 14-18 Example of 8-bit NAND Flash Connection

14.5.4 Example of Toggle NAND Flash Access

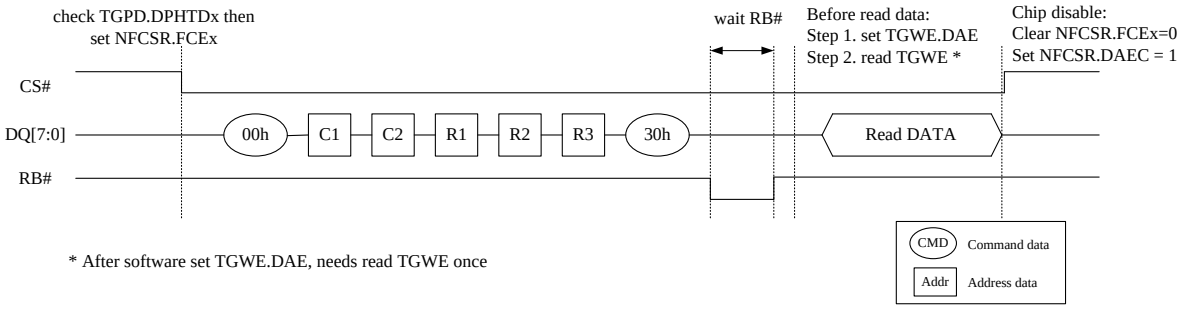


Figure 14-19 Toggle NAND Page Read Operation

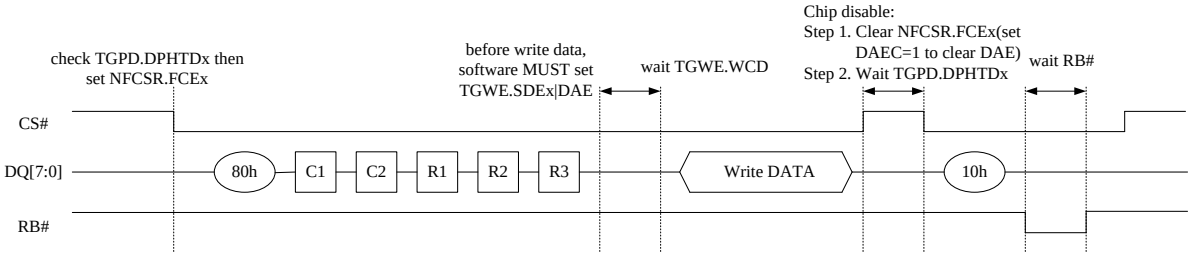


Figure 14-20 Toggle NAND Page Program Operation

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

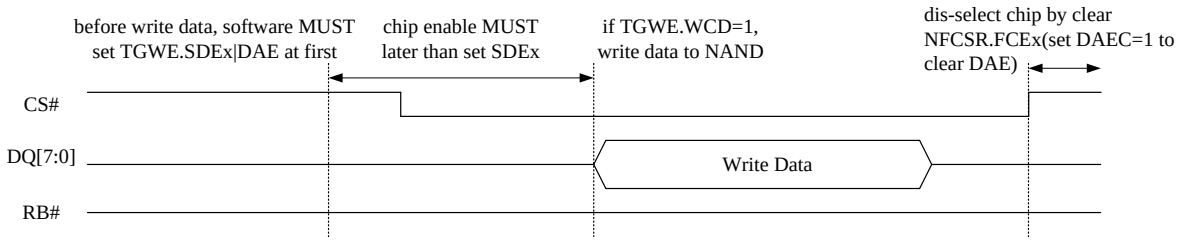


Figure 14-21 Toggle NAND Page Program Operation (Basic)

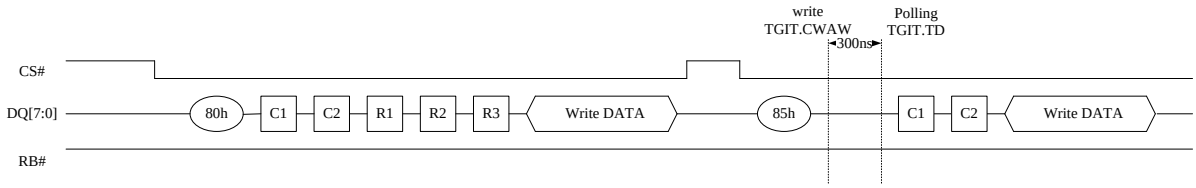


Figure 14-22 Program Operation with Random Data Input

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15 BCH Controller

15.1 Overview

The BCH Controller implements data ECC encoding and decoding.

Table15-1 BCH Registers

Name	Description	RW	Reset Value	Address	Access Width
BHCR	BCH Control register	RW	0x00000000	0x134D0000	32
BHCSR	BCH Control Set register	W	Undefined	0x134D0004	32
BHCCR	BCH Control Clear register	W	Undefined	0x134D0008	32
BHCNT	BCH ENC/DEC Count register	RW	0x00000000	0x134D000C	32
BHDR	BCH data register	W	Undefined	0x134D0010	32/16/8
BHPAR0	BCH Parity 0 register	R	0x00000000	0x134D0014	32
BHPAR1	BCH Parity 1 register	R	0x00000000	0x134D0018	32
BHPAR2	BCH Parity 2 register	R	0x00000000	0x134D001C	32
BHPAR3	BCH Parity 3 register	R	0x00000000	0x134D0020	32
BHPAR4	BCH Parity 4 register	R	0x00000000	0x134D0024	32
BHPAR5	BCH Parity 5 register	R	0x00000000	0x134D0028	32
BHPAR6	BCH Parity 6 register	R	0x00000000	0x134D002C	32
BHPAR7	BCH Parity 7 register	R	0x00000000	0x134D0030	32
BHPAR8	BCH Parity 8 register	R	0x00000000	0x134D0034	32
BHPAR9	BCH Parity 9 register	R	0x00000000	0x134D0038	32
BHPAR10	BCH Parity 10 register	R	0x00000000	0x134D003C	32
BHPAR11	BCH Parity 11 register	R	0x00000000	0x134D0040	32
BHPAR12	BCH Parity 12 register	R	0x00000000	0x134D0044	32
BHPAR13	BCH Parity 13 register	R	0x00000000	0x134D0048	32
BHPAR14	BCH Parity 14 register	R	0x00000000	0x134D004C	32
BHPAR15	BCH Parity 15 register	R	0x00000000	0x134D0050	32
BHPAR16	BCH Parity 16 register	R	0x00000000	0x134D0054	32
BHPAR17	BCH Parity 17 register	R	0x00000000	0x134D0058	32
BHPAR18	BCH Parity 18 register	R	0x00000000	0x134D005C	32
BHPAR19	BCH Parity 19 register	R	0x00000000	0x134D0060	32
BHPAR20	BCH Parity 20 register	R	0x00000000	0x134D0064	32
BHPAR21	BCH Parity 21 register	R	0x00000000	0x134D0068	32
BHPAR22	BCH Parity 22 register	R	0x00000000	0x134D006C	32
BHPAR23	BCH Parity 23 register	R	0x00000000	0x134D0070	32
BHPAR24	BCH Parity 24 register	R	0x00000000	0x134D0074	32
BHPAR25	BCH Parity 25 register	R	0x00000000	0x134D0078	32
BHPAR26	BCH Parity 26 register	R	0x00000000	0x134D007C	32
BHPAR27	BCH Parity 27 register	R	0x00000000	0x134D0080	32
BHERR0	BCH Error Report 0 register	R	0x00000000	0x134D0084	32
BHERR1	BCH Error Report 1 register	R	0x00000000	0x134D0088	32
BHERR2	BCH Error Report 2 register	R	0x00000000	0x134D008C	32
BHERR3	BCH Error Report 3 register	R	0x00000000	0x134D0090	32
BHERR4	BCH Error Report 4 register	R	0x00000000	0x134D0094	32
BHERR5	BCH Error Report 5 register	R	0x00000000	0x134D0098	32
BHERR6	BCH Error Report 6 register	R	0x00000000	0x134D009C	32
BHERR7	BCH Error Report 7 register	R	0x00000000	0x134D00A0	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

BHERR8	BCH Error Report 8 register	R	0x00000000	0x134D00A4	32
BHERR9	BCH Error Report 9 register	R	0x00000000	0x134D00A8	32
BHERR10	BCH Error Report 10 register	R	0x00000000	0x134D00AC	32
BHERR11	BCH Error Report 11 register	R	0x00000000	0x134D00B0	32
BHERR12	BCH Error Report 12 register	R	0x00000000	0x134D00B4	32
BHERR13	BCH Error Report 13 register	R	0x00000000	0x134D00B8	32
BHERR14	BCH Error Report 14 register	R	0x00000000	0x134D00BC	32
BHERR15	BCH Error Report 15 register	R	0x00000000	0x134D00C0	32
BHERR16	BCH Error Report 16 register	R	0x00000000	0x134D00C4	32
BHERR17	BCH Error Report 17 register	R	0x00000000	0x134D00C8	32
BHERR18	BCH Error Report 18 register	R	0x00000000	0x134D00CC	32
BHERR19	BCH Error Report 19 register	R	0x00000000	0x134D00D0	32
BHERR20	BCH Error Report 20 register	R	0x00000000	0x134D00D4	32
BHERR21	BCH Error Report 21 register	R	0x00000000	0x134D00D8	32
BHERR22	BCH Error Report 22 register	R	0x00000000	0x134D00DC	32
BHERR23	BCH Error Report 23 register	R	0x00000000	0x134D00E0	32
BHERR24	BCH Error Report 24 register	R	0x00000000	0x134D00E4	32
BHERR25	BCH Error Report 25 register	R	0x00000000	0x134D00E8	32
BHERR26	BCH Error Report 26 register	R	0x00000000	0x134D00EC	32
BHERR27	BCH Error Report 27 register	R	0x00000000	0x134D00F0	32
BHERR28	BCH Error Report 28 register	R	0x00000000	0x134D00F4	32
BHERR29	BCH Error Report 29 register	R	0x00000000	0x134D00F8	32
BHERR30	BCH Error Report 30 register	R	0x00000000	0x134D00FC	32
BHERR31	BCH Error Report 31 register	R	0x00000000	0x134D0100	32
BHERR32	BCH Error Report 32 register	R	0x00000000	0x134D0104	32
BHERR33	BCH Error Report 33 register	R	0x00000000	0x134D0108	32
BHERR34	BCH Error Report 34 register	R	0x00000000	0x134D010C	32
BHERR35	BCH Error Report 35 register	R	0x00000000	0x134D0110	32
BHERR36	BCH Error Report 36 register	R	0x00000000	0x134D0114	32
BHERR37	BCH Error Report 37 register	R	0x00000000	0x134D0118	32
BHERR38	BCH Error Report 38 register	R	0x00000000	0x134D011C	32
BHERR39	BCH Error Report 39 register	R	0x00000000	0x134D0120	32
BHERR40	BCH Error Report 40 register	R	0x00000000	0x134D0124	32
BHERR41	BCH Error Report 41 register	R	0x00000000	0x134D0128	32
BHERR42	BCH Error Report 42 register	R	0x00000000	0x134D012C	32
BHERR43	BCH Error Report 43 register	R	0x00000000	0x134D0130	32
BHERR44	BCH Error Report 44 register	R	0x00000000	0x134D0134	32
BHERR45	BCH Error Report 45 register	R	0x00000000	0x134D0138	32
BHERR46	BCH Error Report 46 register	R	0x00000000	0x134D013C	32
BHERR47	BCH Error Report 47 register	R	0x00000000	0x134D0140	32
BHERR48	BCH Error Report 48 register	R	0x00000000	0x134D0144	32
BHERR49	BCH Error Report 49 register	R	0x00000000	0x134D0148	32
BHERR50	BCH Error Report 50 register	R	0x00000000	0x134D014C	32
BHERR51	BCH Error Report 51 register	R	0x00000000	0x134D0150	32
BHERR52	BCH Error Report 52 register	R	0x00000000	0x134D0154	32
BHERR53	BCH Error Report 53 register	R	0x00000000	0x134D0158	32
BHERR54	BCH Error Report 54 register	R	0x00000000	0x134D015C	32
BHERR55	BCH Error Report 55 register	R	0x00000000	0x134D0160	32
BHERR56	BCH Error Report 56 register	R	0x00000000	0x134D0164	32
BHERR57	BCH Error Report 57 register	R	0x00000000	0x134D0168	32
BHERR58	BCH Error Report 58 register	R	0x00000000	0x134D016C	32
BHERR59	BCH Error Report 59 register	R	0x00000000	0x134D0170	32
BHERR60	BCH Error Report 60 register	R	0x00000000	0x134D0174	32
BHERR61	BCH Error Report 61 register	R	0x00000000	0x134D0178	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

BHERR62	BCH Error Report 62 register	R	0x00000000	0x134D017C	32
BHERR63	BCH Error Report 63 register	R	0x00000000	0x134D0180	32
BHINT	BCH Interrupt Status register	RW	0x00000000	0x134D0184	32
BHINTES	BCH Interrupt Set register	W	Undefined	0x134D0188	32
BHINTEC	BCH Interrupt Clear register	W	Undefined	0x134D018C	32
BHINTE	BCH Interrupt Enable register	RW	0x00000000	0x134D0190	32
BHTO	BCH User Tag Output register	R	Undefined	0x134D0194	32
BHEA	BCH Erase Analyzer Register	RW	0x00000000	0x134D0198	32

15.2 Register Description

15.2.1 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																												0x134D0000			
Bit	31	30	29	28	27	26	25	24	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

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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		8 8-bit correction 12 12-bit correction … 60 60-bit correction 64 64-bit correction	
3	Reserved	Writing has no effect, read as zero.	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

15.2.2 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 ignored.

	BHCSR																													0x134D0004	
Bit	31	30	29	28	27	26	25	24	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	0	0	0	0	0	0	0	0	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	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

15.2.3 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																													0x134D0008
Bit	31	30	29	28	27	26	25	24	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	0	0	0	0	0	0	0	0	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	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 BHCR.BSEL to 0.	W
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

15.2.4 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																													0x134D000C
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

15.2.5 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																0x134D0010													
Bit	31	30	29	28	27	26	25	24	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	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15.2.6 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 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	BHPAR0	0x134D0014
	BHPAR1	0x134D0018
	BHPAR2	0x134D001C
	BHPAR3	0x134D0020
	BHPAR4	0x134D0024
	BHPAR5	0x134D0028
	BHPAR6	0x134D002C
	…	
	BHPAR25	0x134D0078
	BHPAR26	0x134D007C
	BHPAR27	0x134D0080

Bit	31	30	29	28	27	26	25	24	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

15.2.7 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).

	BHERR0	0x134D0084
	BHERR1	0x134D0088
	BHERR2	0x134D008C
	BHERR3	0x134D0090
	BHERR4	0x134D0094
	BHERR5	0x134D0098
	BHERR6	0x134D009C
	BHERR7	0x134D00A0
	…	…
	BHERR61	0x134D0178
	BHERR62	0x134D017C
	BHERR63	0x134D0180

Bit	31	30	29	28	27	26	25	24	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,	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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).	
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

15.2.8 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																													0x134D0184
Bit	31	30	29	28	27	26	25	24	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	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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		block since the BPSO was asserted at the beginning of the correction operation.	
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) 1 All data (data + parity bytes) are 0xff	RW
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.2.9 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																													0x134D0188
Bit	31	30	29	28	27	26	25	24	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	0	0	0	0	0	0	0	0	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	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 BHINTE.ENCFE to 1.	W
0	ERRES	**Error Interrupt Enable Set:** It is used to set BHINTE.ERRE to 1.	W

15.2.10 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																													0x134D018C
Bit	31	30	29	28	27	26	25	24	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	0	0	0	0	0	0	0	0	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	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

15.2.11 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																												0x134D0190					
Bit	31	30	29	28	27	26	25	24	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

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

15.2.12 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																0x134D0194														
Bit	31	30	29	28	27	26	25	24	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	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

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

15.2.13 BCH Erase Analyzer Register (BHEA)

BHEA is a 32-bit read only register that is used to indicate the tag for the current corrected data block.

	BHEA																0x134D0198														
Bit	31	30	29	28	27	26	25	24	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	ZB							Reserved			TZB					Reserved															ZBE
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	Reserved	Writing has no effect, read as zero. NOTE. Software must **NOT** write 1 to this field.	R
30:24	ZB	**Zero bits in data and parity zone**	R
23:21	Reserved	Writing has no effect, read as zero. NOTE. Software must **NOT** write 1 to this field.	R
23:16	TZB	**Tolerable Zero bits in data and parity zone**	RW
15:1	Reserved	Writing has no effect, read as zero. NOTE. Software must **NOT** write 1 to this field.	R
0	ZBE	**Zero Bits Enable**	RW

15.3 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.

15.3.1 Encoding Sequence

BCH encoding can be operated by CPU or DMA.

15.3.1.1 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

15.3.1.2 DMA

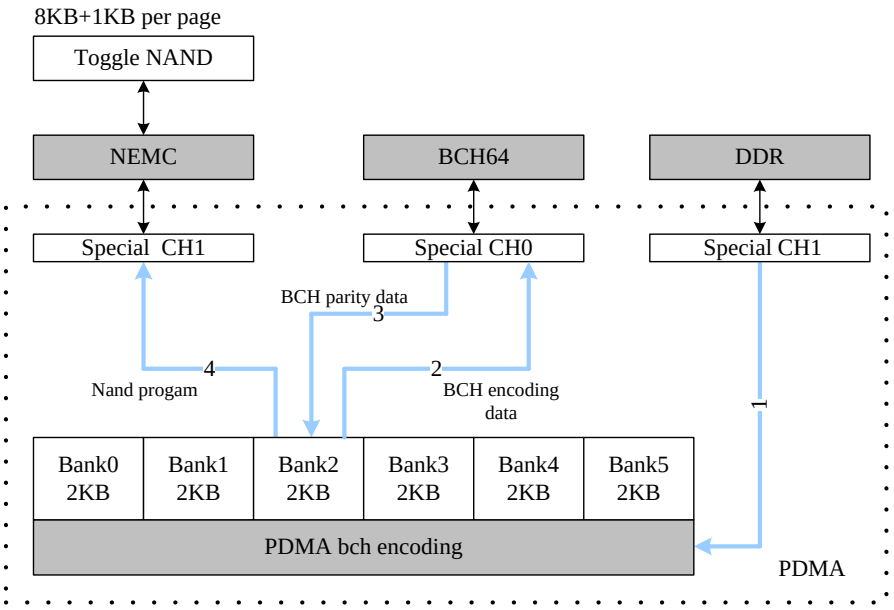


Figure 15-1 Block Diagram for BCH encoding

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.

15.3.2 Decoding Sequence

BCH decoding can be operated by CPU or DMA.

15.3.2.1 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 © 2005-2015 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.

15.3.2.2 DMA

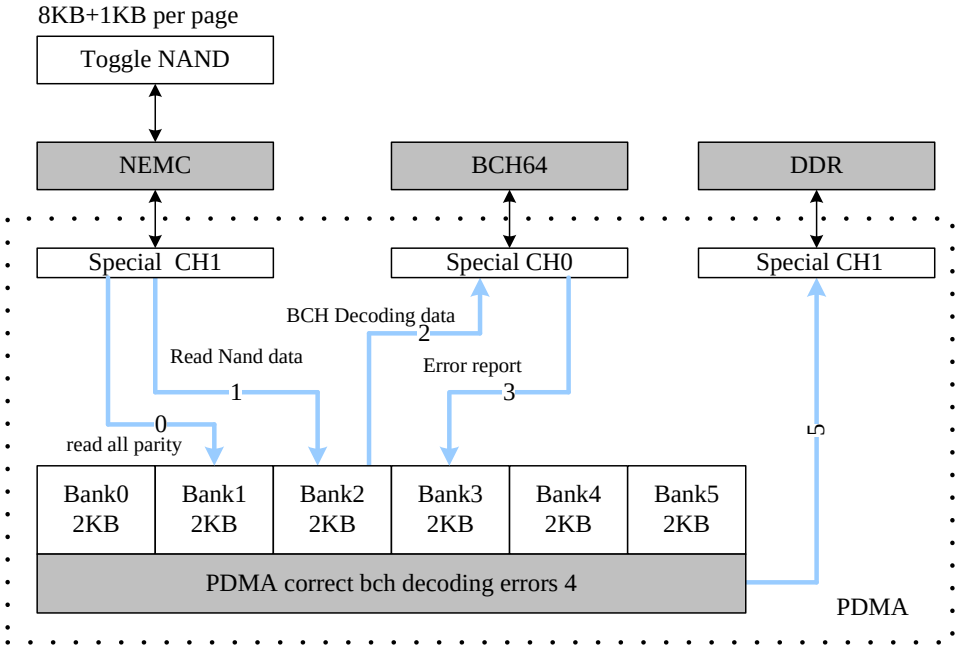


Figure 15-2 Block Diagram for BCH decoding

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 © 2005-2015 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.

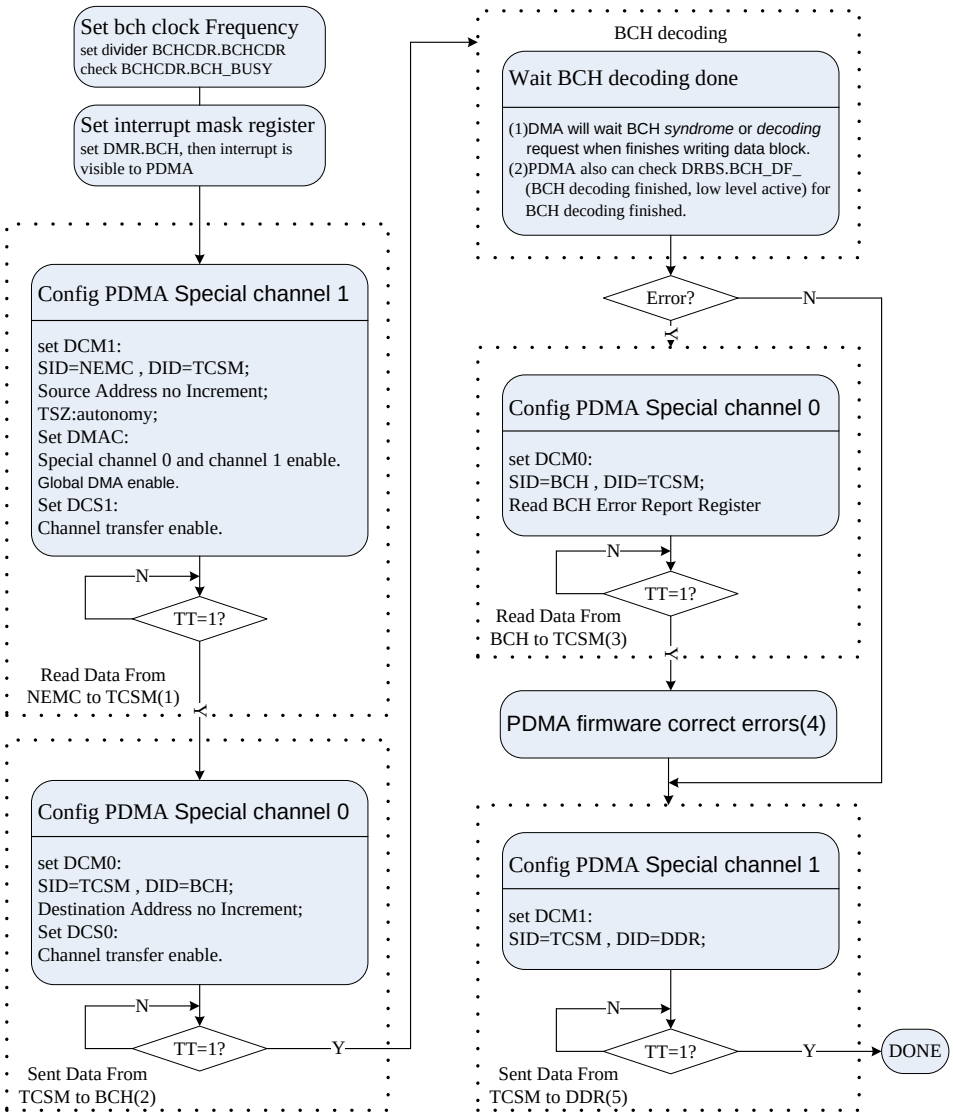


Figure 15-3 BCH decoding data flow

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 5

SYSTEM FUNCTIONS

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16 Clock Reset and Power Controller

16.1 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).

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 VPU, X2D module. When in Sleep mode, software may shut down J1. 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.

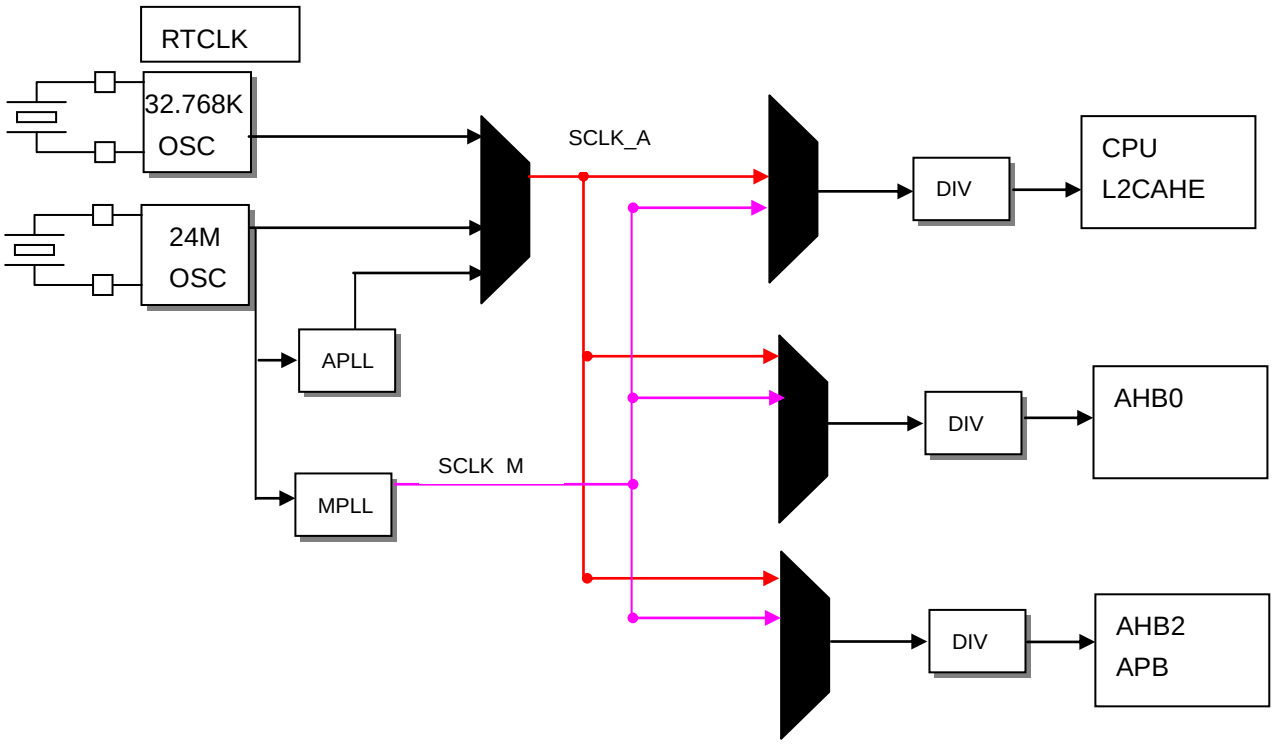
Feature:

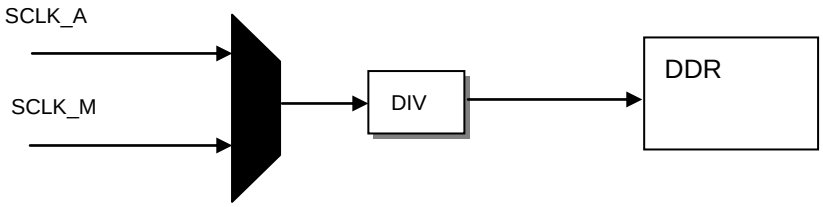
- On-chip 2MHz~27MHZ oscillator circuit
- On-chip 32.768KHZ oscillator circuit
- One two-chip phase-locked loops (PLL) with programmable multiplier
- CCLK, HHCLK, H2CLK, PCLK, H0CLK, DDR_CLK, VPU_CLK frequency can be changed separately for software by setting registers
- SSI clock supports 50M clock
- MSC clock supports 100M clock
- Functional-unit clock gating
- Shut down power supply for J1, VPU, X2D, L2CC

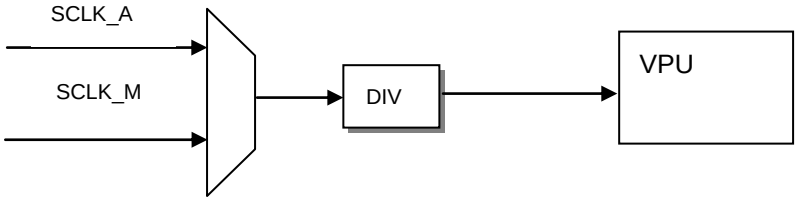
16.1.1 CGU Block Diagram

Following figure illustrates a block diagram of CGU.

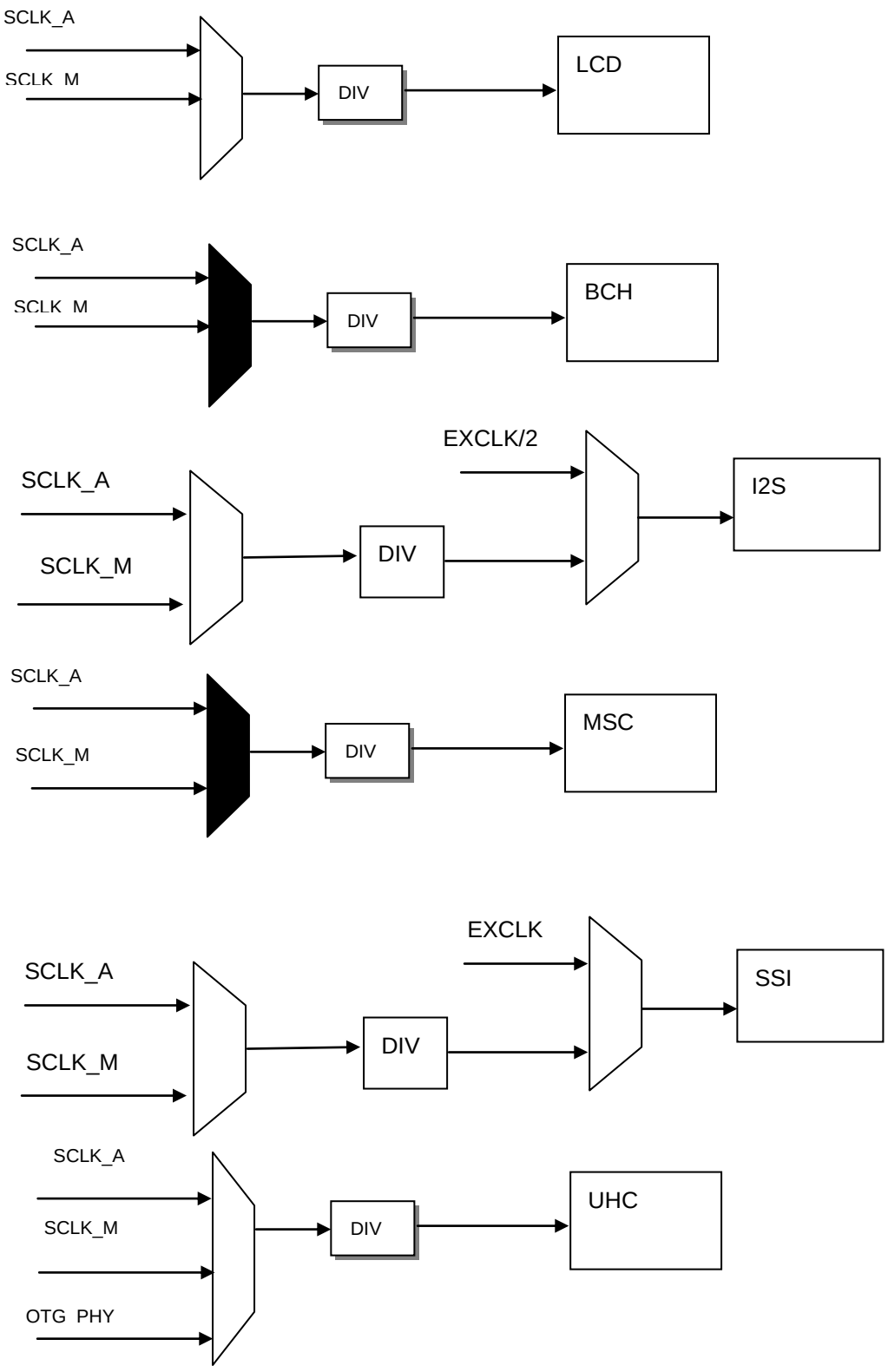
Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



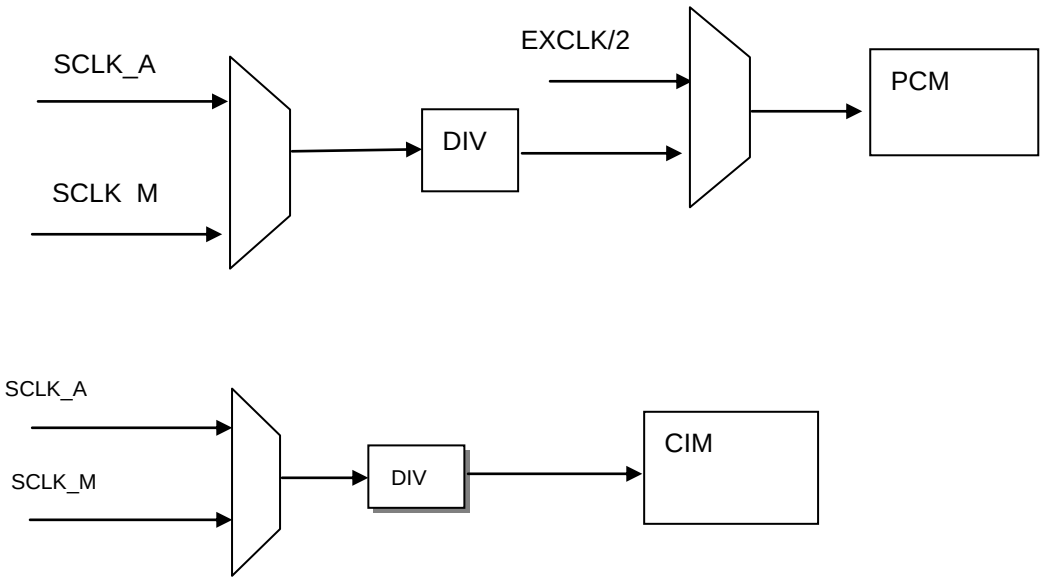




Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Copyright © 2005-2015 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.

16.1.2 CGU Registers

All CGU register 32bit access address is physical address.

Table 16-1 CGU Registers Configuration

Name	description	RW	Reset Value	Address	Access Size
CPCCR	Clock Control Register	RW	0x95000000	0x10000000	32
CPCSR	Clock status register	RW	0x00000000	0x100000D4	32
DDCDR	DDR clock divider Register	RW	0x40000000	0x1000002C	32
VPUCDR	VPU clock divider Register	RW	0x00000000	0x10000030	32
I2SCDR	I2S device clock divider Register	RW	0x00000000	0x10000060	32
I2S1CDR	I2S1 device clock divider Register	RW	0x00000000	0x100000A0	32
USBCDR	OTG PHY clock divider Register	RW	0x00000000	0x10000050	32
LPCDR	LCD pix clock divider Register	RW	0x00000000	0x10000064	32
MSC0CDR	MSC0 clock divider Register	RW	0x40000000	0x10000068	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

MSC1CDR	MSC1 clock divider Register	RW	0x00000000	0x100000A4	32
MSC2CDR	MSC2 clock divider Register	RW	0x00000000	0x100000A8	32
UHCCDR	UHC 48M clock divider Register	RW	0x00000000	0x1000006C	32
SSICDR	SSI clock divider Register	RW	0x00000000	0x10000074	32
CIMCDR	CIM MCLK clock divider Register	RW	0x00000000	0x1000007C	32
CIM1CDR	CIM1 MCLK clock divider Register	RW	0x00000000	0x10000080	32
PCMCDR	PCM device clock divider Register	RW	0x00000000	0x10000084	32
BCHCDR	BCH clock divider Register	RW	0x40000000	0x100000AC	32
CPM_INTR	CPM interrupt Register	RW	0x00000000	0x100000B0	32
CPM_INTRE	CPM interrupt Enable Register	RW	0x00000000	0x100000B4	32
CPSPR	CPM Scratch Pad Register	RW	0x????????	0x10000034	32
CPSPPR	CPM Scratch Protected Register	RW	0x0000a5a5	0x10000038	32
USBPCR	USB Parameter control register	RW	0x429919b8	0x1000003C	32
USBRDT	USB Reset Detect Timer Register	RW	0x02000096	0x10000040	32
USBVBFIL	USB jitter filter Register	RW	0x00ff0080	0x10000044	32
USBPCR1	USB Parameter control register 1	RW	0x8dc23360	0x10000048	32
CPPCR	PLL Control Register	RW	0x80000020	0x1000000C	32
CPAPCR	APLL Control Register	RW	0x00000000	0x10000010	32
CPMPCR	MPLL Control Register	RW	0x00000000	0x10000014	32
CPEPCR	EPLL Control Register	RW	0x00000000	0x10000018	32
CPVPCR	VPLL Control Register	RW	0x00000000	0x1000001C	32

16.1.2.1 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																														0x10000000
Bit	31	30	29	28	27	26	25	24	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		Reserved	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 11: RTCLK The mux clock output is SCLK_A	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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.</td><td></td></tr>
<tr><td>29:28</td><td>SEL_CPLL</td><td>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.</td><td>RW</td></tr>
<tr><td>27:26</td><td>SEL_H0PLL</td><td>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.</td><td>RW</td></tr>
<tr><td>25:24</td><td>SEL_H2PLL</td><td>00 : stop MUX clock output
01: SCLK_A
10: MPLL
11: TCK
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.</td><td>RW</td></tr>
<tr><td>23</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>22</td><td>CE_CPU</td><td>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.</td><td>RW</td></tr>
<tr><td>21</td><td>CE_AHB0</td><td>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.</td><td>RW</td></tr>
<tr><td>20</td><td>CE_AHB2</td><td>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.</td><td>RW</td></tr>
<tr><td>19:16</td><td>PDIV</td><td>Divider for Peripheral Frequency. Specified the PCLK division ratio.
<table>
<tr><th colspan="4">Bit 19~16: PDIV</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>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>
<table>
<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 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.
</td><td></td></tr>
<tr><td>15:12</td><td>H2DIV</td><td>Divider for AHB2 Frequency. Specified the AHB2 CLK division ratio.
<table>
<tr><td colspan="4">Bit 15~12: H2DIV</td><td>Description</td></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 H2DIV == 1111 and CE is 1, H2CLK will stop. H2DIV value will not be changed actually, it is only set register.
When software sets H2DIV == 1110 and CE is 1, H2CLK will continue to run. H2DIV value will not be changed actually, it is</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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>
<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>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			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 Note : for PDIV and AHB2DIV, AHB2's clk frequency must be 1 or 2 times of PDIV's clk frequency. For L2CDIV and CDIV, CPU's clk frequency must be 1,2,3, or 4 times of L2CACHE's clk frequency.

16.1.2.2 Clock Status Register

Clock status register (CPCSR) is a 32-bit read/write register that specifies the status of

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

CPCCR .This register is initialized to 0x00000000 only by any reset. Only word access can be used on CPCSR.

	CPCSR																												0x100000D4		
Bit	31	30	29	28	27	26	25	24	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

16.1.2.3 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 access can be used on DDRCDR.

	DDRCDR																												0x1000002C			
Bit	31	30	29	28	27	26	25	24	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	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
26:4	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

16.1.2.4 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																															0x10000030
Bit	31	30	29	28	27	26	25	24	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 00: VPU clock source is SCLK_A	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		01: VPU clock source is MPLL It is no glitch free mux. Software should be stop VPU clock, when change this bit.	
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

16.1.2.5 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																													0x10000060
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	I2CS	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	I2CS	I2S Clock Source Selection. Selects the I2S clock source between PLL output and pin EXCLK/2. 0: I2S clock source is EXCLK/2 1: I2S clock source is PLL output divided by I2SDIV It is no glitch free mux. Software should be stop I2S clock, when change this bit. If I2CS == 0, the clock divider is gated, don't set CE_I2S	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30	I2PCS	0: SCLK_A output 1: MPLL output	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

16.1.2.6 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 only by any reset. Only word access can be used on LPCDR.

	LPCDR																													0x10000064
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	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

Bits	Name	Description	RW
31	LPCS	00: SCLK_A output 01: 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.	R
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

16.1.2.7 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																														0x10000068
Bit	31	30	29	28	27	26	25	24	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		CE_MSC0	MSC0_BUSY	MSC0_STOP	Reserved											S_CLK0_SEL	Reserved							MSC0CDR							
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	MPCS	00: stop mux clock output 01: select SCLK_A clock output 10: select MPLL clock output The clock mux is 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
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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		frequency change.	
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

16.1.2.8 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																													0x100000A4
Bit	31	30	29	28	27	26	25	24	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

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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		If system don't use this module, may set this bit to reduce power .	
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

16.1.2.9 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
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

14:8	Reserved		
7:0	MSC2CDR	Divider for MSC2 Frequency. Specified the MSC2 device clock division ratio division ratio = (MSC2CDR + 1)*2.	RW

16.1.2.10 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.

	USBCDR																													0x10000050
Bit	31	30	29	28	27	26	25	24	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	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, the clock divider is gated, don't set CE_USB 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_UHC	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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		If system don't use this module, may set this bit to reduce power .	
26:8	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

16.1.2.11 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																													0x1000006C
Bit	31	30	29	28	27	26	25	24	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 OTG_PHY It is no glitch free mux. Software should be stop the module clock, when change this bit **When using OTG_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 UHCCDR have no affect	RW
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.1.2.12 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																													0x10000074
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCS	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	SCS	SSI Clock Source Selection. Selects the SSI clock source between PLL output and pin EXCLK. 0: SSI clock source is EXCLK 1: SSI clock source is PLL output divided by SSICDR If SCS == 0, the clock divider is gated, don't set CE_SSI It is no glitch free mux. Software should be stop the module clock, when change this bit	R
30	SPCS	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_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 frequency change.	RW
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

16.1.2.13 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																													0x1000007C
Bit	31	30	29	28	27	26	25	24	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. The clock mux output is to CIM,CIM1. It is source clock of CIM,CIM1.	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. 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 .	RW
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.1.2.14 CIM1 MCLK clock divider Register

CIM1 mclk clock divider Register (CIM1CDR) is a 32-bit read/write register that specifies the divider of CIM1 mclk clock (CIM1_MCLK). This register is initialized to 0x00000000 only by any reset. Only word access can be used on CIM1CDR.

	CIM1CDR																													0x10000080
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		CE_CIM1	CIM1_BUSY	CIM1_STOP	Reserved																				CIM1CDR							
RST	0	0	0	0	0	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	Reserved		R
30	CE_CIM 1	Change enable for CIM1. If CE_CIM1 is 1 , write on CIM1CDR ill start a frequency changing sequence immediately. If CE_CIM1 is 0, writes on CIM1CDR have no affect	RW
29	CIM1_B USY	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 CIM1_BUSY == 0, may begin another frequency change.	RW
28	CIM1_ST OP	When CIM1_STOP is 1 and CE_CIM1 is 1, the CIM1 clock will stop. When CIM1_STOP is 0 and CE_CIM1 is 1, the CIM 1clock will continue. Software should wait until CIM1_BUSY == 0, may begin another frequency change. CIM1_STOP is prior to CIM1CDR. If system don't use this module, may set this bit to reduce power .	RW
27:8	Reserved	Writing has no effect, read as zero.	R
7:0	CIM1CD R	Divider for CIM1 MCLK Frequency. Specified the CIM1 MCLK clock (CIM1_MCLK) division ratio, which varies from 1 to 256 (division ratio = CIM1CDR + 1). the output clk is io_setup.cim1_mclk_o	RW

16.1.2.15 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PCMCDR																														0x10000084	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCMS	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	PCMS	PCM source clock Selection. 0: PCM source clock is pin EXCLK/2 1: PCM source clock is PLL divider output It is no glitch free mux. Software should be stop the module clock, when change this bit	RW
30	PCMPCS	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	Reserved		
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
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

16.1.2.16 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	BCHCDR																															0x100000AC
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BPCS		CE_BCH	BCH_BUSY	BCH_STOP	Reserved																							BCHCDR			
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	BPCS	00: stop clock mux output 01: select SCLK_A output 10: select MPLL output 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 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

16.1.2.17 MAC PHY Control Register (MPHYC)

The MAC PHY Control Register is a 32-bit read/write register that control GMAC interface.It is initialized to 0x0000000 0by reset.

	MPHYC																															0x100000E0
Bit	31	30	29	28	27	26	25	24	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TXCLK_SEL	MAC_SPEED		Reserved																								SOFT_RST	PHY_INTF		
RST	0	0	0	0	0	0	0	0	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	TXCLK_SEL	0: TXCLK from input pad 1: TXCLK from pll output	RW
30:29	MAC_SPEED	It is only read. 0x: 1000M 10: 10M 11: 100M	R
29:4	Reserved		R
3	SOFT_RST	1:reset gmac 0:don't reset gmac	RW
2:0	PHY_INTF	000: GMII/MII 001: RGMII 100: RMII	RW

16.1.2.18 CPM interrupt Register

	CPM_INTR																0x100000B0														
Bit	31	30	29	28	27	26	25	24	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
1	VBUS_INTR	B device insert interrupt.	R
0	ADEV_INTR	A device insert interrupt.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.1.2.19 CPM interrupt enable Register

	CPM_INTRE																														0x100000B4
Bit	31	30	29	28	27	26	25	24	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

16.1.2.20 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																														0x10000038
Bit	31	30	29	28	27	26	25	24	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																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

16.1.2.21 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																														0x10000034
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.1.2.22 USB Parameter Control Register

The USBPCR is a 32-bit read/write register that allows software to control OTG PHY some functions. It is initialized to 0x409919b8.

	USBPCR																															0x1000003C
Bit	31	30	29	28	27	26	25	24	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. 2’b1x: “idpullup” always active 2’b01: “idpullup” always active when usb suspend 2’b00: use “idpullup” from otg controller	RW
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 OTG 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	This bit controls OTG PHY VBUSVLDEXT signal.	RW
23	VBUSVLDEXTSEL	This bit controls OTG PHY VBUSVLDEXTSEL signal.	RW
22	POR	This bit controls OTG PHY power on reset.	RW
21	SIDDQ	This bit is the OTG PHY analog blocks power down signal.	RW
20	OTG_DISABLE	This bit is the power control for otg block in OTG PHY.	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%	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		3'b010	-3%		
		3'b001	-4.5%		
		3'b000	-6%		
16:14	OTGTUNE	These bits control VBUS valid threshold adjustment.		RW	
		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%		
13:11	SQRXTUNE	These bits control squelch threshold adjustment.		RW	
		3'b111	-20%		
		3'b110	-15%		
		3'b101	-10%		
		3'b100	-5%		
		3'b011	default		
		3'b010	+5%		
		3'b001	+10%		
		3'b000	+15%		
10:7	TXFSLSTUNE	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%		
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%		

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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%		

16.1.2.23 USB Reset Detect Timer Register

It is initialized to 0x2000096

	USBRDT																														0x10000040
Bit	31	30	29	28	27	26	25	24	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

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

16.1.2.24 USB VBUS Jitter Filter Register

	USBVBFIL																														0x10000044
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

16.1.2.25 USB Parameter Control Register1

The USBPCR1 is a 32-bit read/write register that allows software to control OTG and OHCI PHY some functions. It is initialized to 0x8dc23360.

	USBPCR1																													0x10000048
Bit	31	30	29	28	27	26	25	24	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		Reserved		PORT0_RST	Reserved	WORD_IF0	Reserved	PDBAR		IBSOPT[2:0]			XP		SP			SM		START_CLK	OVERCURRENT	INCRX	INCR4	INCR8	INCR16
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	0	0	0	0	0

Bits	Name	Description	RW
31	BVLD_REG	This bit is used to set “bvalid”(VBUS above B-device session threshold) signal.	R
30:28	Reserved	Writing has no effect, read as zero.	
27:26	REFCLKSEL	This bus selects the OTG_PHY reference clock source 11, 10: The PLL uses CLKCORE as reference 01: The XO block uses an external, 2.5v clock supplied on the XO pin 00: The XO block uses the clock from a crystal.	RW
25:24	REFCLKDIV	This bus selects the OTG_PHY reference clock frequency 11: Reserved 10: 48MHz 01: 24MHz 00: 12MHz	RW
23:22	Reserved	Writing has no effect.	R
21	PORT_RST	OTG PHY Port 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.	RW
20	Reserved	Writing has no effect, read as zero.	R
19	WORD_IF	This bit selects utmi data bus width of otg 1: 16bit/30M 0: 8bit/60M	RW
18	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

17	PDBAR	Power down Mode. Enables OHCI PHY power down state. 0: Power down 1: Power On	RW
16:14	IBSOPT[2:0]	OHCI PHY Current option IBSOPT2 IBSOPT1 IBSOPT0 0 0 0 6u 0 0 1 5u 0 1 0 4u	RW
13:12	XP[1:0]	OHCI PHY Cross-point control of DP, DM XP1 XP0 0 X default cross-point: $V_{DD}/2$ 1 1 cross-point up: $V_{DD}/2$ + 400mV 1 0 cross-point down: $V_{DD}/2$ - 400mV	RW
11:9	SP[2:0]	OHCI PHY Pull-up and pull-down control of DP, SP0: 0 for RPD open, 1 for RPD connect SP1 SP2 0 X open 1 0 RPU1 1 1 RPU1 + RPU2	RW
8:6	SM[2:0]	OHCI PHY Pull-up and pull-down control of DM SM0: 0 for RPD open, 1 for RPD connect SM1 SM2 0 X open 1 0 RPU1 1 1 RPU1 + RPU2	RW
5	START_CLK	OHCI Clock control signal**:** This is an asynchronous primary input to the host core. When the OHCI clocks are suspended, the system has to assert this signal to start the clocks (12 and 48 MHz). This should be deasserted after the clocks are started and before the host is suspended again. (Host is suspended means HCFS = SUSPEND or all the OHCI ports are suspended).	RW
4	OVERCURRENT	OHCI Port Overcurrent Indication From Application**:** When asserted by the application, the corresponding port enters Disable state. This signal controlsOHCI controller port state machines. Depending on ownership of the port, the corresponding OHCI controller generates an Overcurrent Detect interrupt. **Note**: that you must implement overcurrent detection logic and provide input to the host. When an overcurrent condition exists, port power remains on. Use the overcurrent condition to control the port power.	RW
3	INCRX	OHCI Burst Alignment Enable**:** Forces AHB master to start INCR4/8/16 busts only on burst boundaries. AHB requires that double word width burst be addressed-aligned only to the double-word boundary. 1'b1: Start INCRX burst only on burst x-aligned addresses 1'b0: Normal AHB operation; start bursts on any double word boundary	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Note: When this function is enabled, the burst are started only when the lowest bits of haddr are: INCR4: haddr[3:0] == 4'b0000 INCR8: haddr[4:0] == 5'b00000 INCR16: haddr[5:0] == 6'b000000	
2	INCR4	OHCI AHB Burst Type INCR4 Enable**:** Enables the AHB master interface to utilize burst INCR8 when appropriate. 1'b1: Use INCR4 when appropriate 1'b0: Do not use INCR4; use other enabled INCRX bursts or unspecified length burst INCR	RW
1	INCR8	OHCI AHB Burst Type INCR8 Enable**:** Enables the AHB master interface to utilize burst INCR8 when appropriate. 1'b1: Use INCR8 when appropriate 1'b0: Do not use INCR8; use other enabled INCRX bursts or unspecified length burst INCR	RW
0	INCR16	OHCI AHB Burst Type INCR16 Enable**:** Enables the AHB master interface to utilize burst INCR16 when appropriate. 1'b1: Use INCR16 when appropriate 1'b0: Do not use INCR16; use other enabled INCRX bursts or unspecified length burst INCR	RW

16.1.2.26 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 0x????0020 only by any reset. Only word access can be used on CPAPCR

	CPAPCR																0x10000010													
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BS	PLLM							Reserved	PLLN					PLLOD		LOCK0	Reserved				ON	PLLBP	PLLEN	PLLST							
RST	?	?	?	?	?	?	?	?	0	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bits	Name	Description	RW
31	BS	0: low band 1: high band	RW
30:24	PLLM	the PLL feedback 7-bit divider.	RW
23	Reserved	Writing has no effect, read as zero.	R
22:18	PLLN	the PLL input 5-bit divider.	RW
17:16	PL OD	00: divide by 1 01: divide by 2 10: divide by 4	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		11: divide by 8	
15	LOCK0	0: the PLL output is not stable 1: the PLL output is stable Software should clear this bit to 0, when this bit equal to 1, it indicates that PLL hadn't stable previously , it is only used to debug.	RW
14:11	Reserved	Writing has no effect, read as zero.	R
10	PLLON	PLL Stabilize Flag. 0: PLL is off or not stable 1: PLL is on and stable	R
9	PLLBP	PLL Bypass. 0: PLL output 1: EXCLK output **Don't change the bit. it is only used to debug.**	RW
8	PLLEN	PLL Enable.	RW
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.	RW

16.1.2.27 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 0x????0002 only by any reset. Only word access can be used on CPMPCR.

	CPMPCR																												0x10000014	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BS	PLLM							Reserved	PLLN					PLLOD		Reserved								PLLEN	PLLBP	Reserved				LOCK	PLL1ON
RST	?	?	?	?	?	?	?	?	0	?	?	?	?	?	?	?	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bits	Name	Description	RW
31	BS	0: low band 1: high band	RW
30:24	PLLM	the PLL1 feedback 7-bit divider.	RW
23	Reserved	Writing has no effect, read as zero.	R
22:18	PLLN	the PLL1 input 5-bit divider.	RW
17:16	PLLOD	00: divide by 1 01: divide by 2 10: divide by 4 11: divide by 8	RW
15:8	Reserved	Writing has no effect, read as zero.	R
7	PLLEN	PLL Enable.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

6	PLLBP		RW
5:1	Reserved	Writing has no effect, read as zero.	R
1	LOCK	0: the PLL output is not stable 1: the PLL output is stable Software should clear this bit to 0, when this bit equal to 1, it indicates that PLL hadn't stable previously , it is only used to debug.	RW
0	PLLON	0: PLL1 doesn't enter on state 1: PLL1 is in on state	R

16.1.3 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 1000MHz. 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:

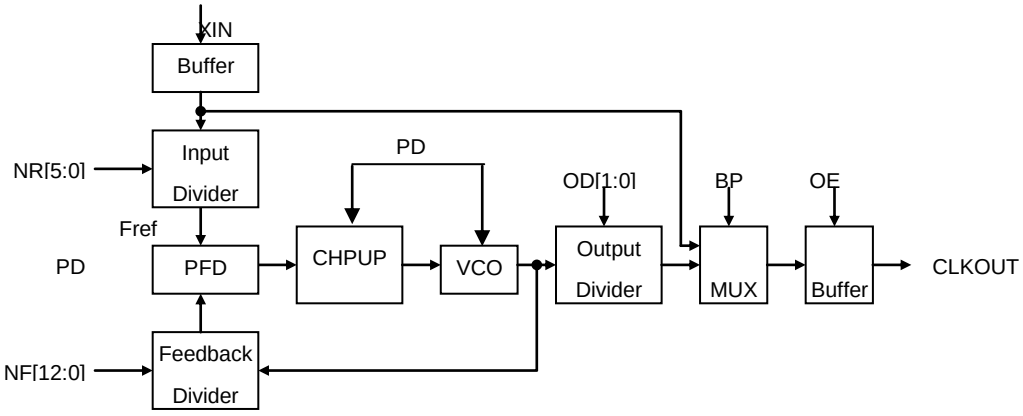


Figure 16-1 Block Diagram of PLL

16.1.3.1 PLL Configuration

PLL Divider Value Setting

There are 3 divider values (N, M and NO) to set the PLL output clock frequency CLKOUT:

1 Input Divider Value N.

N = PLLN of CPPCR

2 Feedback Divider Value M.

M = PLLM of CPPCR

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3 Output Divider Value NO.

Output Divider Setting (OD)	Output Divider Value (NO)
0	1
1	2
2	4
3	8

4 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:

NF = 1+ M0 + M1*2 + M2*4 + M3*8 + M4*16 + M5*32 + M6*64

NR = 1+ N0*1 + N1*2 + N2*4 + N3*8 + N4*16

NO = $2^{od0+2od1}$

FREF = FIN / NR

FVCO = FOUT * NO

FOUT = FIN * NF / (NR*NO), where FREF is the comparison frequency for the PFD.

For proper operation in normal mode, the following constraints must be satisfied:

For high-band,

10 MHz ⩽ FREF ⩽ 50 MHz

500 MHz ⩽ FVCO ⩽ 1000 MHz

62.5 MHz ⩽ FOUT ⩽ 1000 MHz

For low-band:

10 MHz ⩽ FREF ⩽ 50 MHz

300 MHz ⩽ FVCO ⩽ 600 MHz

37.5 MHz ⩽ FOUT ⩽ 600 MHz

16.1.4 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2 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.

16.2.1 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 , 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2.2 Register Description

All PMC register 32bit access address is physical address.

Table 16-2 Power/Reset Management Controller Registers Configuration

Name	description	RW	Initial Value	Address	Access Size
LCR	Low Power Control Register	RW	0x000000F8	0x10000004	32
SPCR0	SRAM Power Control Register0	RW	0x00000000	0x100000B8	32
SPCR1	SRAM Power Control Register1	RW	0x00000000	0x100000BC	32
PSWC0ST	Power Switch Chain0 Start Time	RW	0x00000000	0x10000090	32
PSWC1ST	Power Switch Chain1 Start Time	RW	0x00000000	0x10000094	32
PSWC2ST	Power Switch Chain2 Start Time	RW	0x00000000	0x10000098	32
PSWC3ST	Power Switch Chain3 Start Time	RW	0x00000000	0x1000009c	32
CLKGR0	Clock Gate Register0	RW	0xFFFFFFE0	0x10000020	32
CLKGR1	Clock Gate Register1	RW	0xFFFFFFFF	0x10000028	32
SRBC	Soft Reset and Bus Control Register	RW	0x00000000	0x100000C4	32
SLBC	Sleep Boot Control Register	RW	0x00000000	0x100000C8	32
SLPC	Sleep PC Register	RW	0x????????	0x100000CC	32
ERNG	Enable rng regsiter	RW	0x00000000	0x100000D8	32
RNG	rng regsiter	RW	0x????????	0x100000DC	32
OPCR	Oscillator and Power Control Register	RW	0x000015C0	0x10000024	32

16.2.2.1 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 0x000000F8 by any reset.

	LCR																													0x10000004
Bit	31	30	29	28	27	26	25	24	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_X2D	PD_VPU	Reserved		X2DS	VPUS							PST												DUTY					DOZE	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	1	1	1	1	1	0	0	0

Bits	Name	Description	RW
31	PD_X2D	Power Down Secondary X2D 0: not shut down power supply to Secondary X2D 1: shut down power supply to Secondary X2D	RW
30	PD_VPU	Power Down Module VPU. 0: not shut down power supply to VPU	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: shut down power supply to VPU	
29:28	Reserved		R
27	X2DS	X2D power down status. 0: X2D module not shut down 1: X2D module has entered shut down mode	R
26	VPUS	VPU power down status. 0: VPU module not shut down 1: VPU module has entered shut down mode	R
25:24	Reserved		R
23:20	Reserved	Writing has no effect, read as zero.	R
19:8	PST	Power stability Time. Specifies the Power stabilize time by unit of RTCCLK (approximate 32kHz) cycles.	RW
7:3	DUTY	CPU Clock Duty. Control the CPU clock duty in doze mode. When the DUTY field is 0x1F, the clock is always on and when it is zero, the clock is always off. Set the DUTY field to 0 when the CPU will be disabled for an extended amount of time. 00000: 0/31 duty-cycle 00001: 1/31 duty-cycle 00010: 2/31 duty-cycle … 11111: 31/31 duty-cycle	RW
2	DOZE	Doze Mode. Control the doze mode. When doze mode is canceled, this bit is cleared to 0 automatically. 0: Doze mode is off 1: Doze mode is on	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 **10**: Reserved **11**: Reserved	RW

16.2.2.2 Power Switch Chain0 Start Time Register

	PSWC0ST																														0x10000090	
Bit	31	30	29	28	27	26	25	24	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

16.2.2.3 Power Switch Chain1 Start Time Register

	PSWC1ST																													0x10000094	
Bit	31	30	29	28	27	26	25	24	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

16.2.2.4 Power Switch Chain2 Start Time Register

	PSWC2ST																													0x10000098	
Bit	31	30	29	28	27	26	25	24	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

16.2.2.5 Power Switch Chain3 Start Time Register

	PSWC3ST																													0x1000009C	
Bit	31	30	29	28	27	26	25	24	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.

16.2.2.6 Clock Gate Register

The Clock Gate Register (CLKGR) is a 32-bit read/write register that controls the CLOCK GATE function of peripherals. It is reset to 0x0FFFFFE0.

	CLKGR																													0x10000020	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLKGR																															
RST	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	0	0	0	0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31:0</td><td>CLKGR</td><td>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.

<table>
<tr><th>Bit</th><th>Module</th><th>Description</th></tr>
<tr><td>31</td><td>DDR</td><td></td></tr>
<tr><td>30</td><td>Reserved</td><td></td></tr>
<tr><td>29</td><td>Reserved</td><td></td></tr>
<tr><td>28</td><td>Reserved</td><td></td></tr>
<tr><td>27</td><td>EPDE</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>26</td><td>EPDC</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>25</td><td>LCD</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>24</td><td>CIM1</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>23</td><td>CIM0</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>22</td><td>UHC</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>21</td><td>GMAC</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>20</td><td>PDMA</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>19</td><td>VPU</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>18</td><td>UART3</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>17</td><td>UART2</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>16</td><td>UART1</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>15</td><td>UART0</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>14</td><td>SADC</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>13</td><td>PCM</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>12</td><td>MSC2</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>11</td><td>MSC1</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>10</td><td>AHB_MON</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>9</td><td>X2D</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>8</td><td>AIC</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>7</td><td>I2C2</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>6</td><td>I2C1</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>5</td><td>I2C0</td><td>After reset period, the clock is stopped.</td></tr>
<tr><td>4</td><td>SSI0</td><td></td></tr>
<tr><td>3</td><td>MSC0</td><td></td></tr>
<tr><td>2</td><td>OTG</td><td></td></tr>
<tr><td>1</td><td>BCH</td><td></td></tr>
<tr><td>0</td><td>NEMC</td><td></td></tr>
</table>
</td><td>RW</td></tr>
</table>

16.2.2.7 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

reset and bus transmission. It is initialized to 0x0000000 0by reset.

	SRBC																													
																													0x100000C4	
Bit	31	30	29	28	27	26	25	24	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	Reserved							LCD_STP	LCD_ACK	Reserved				UHC_SR	OTG1_SR	Reserved					AHB0_STP	AHB0_ACK	Reserved	AHB2_STP	AHB2_ACK	Reserved		DDR_STP	DDR0_ACK
RST	0	0	0	0	0	0	0	0	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	Reserved	Writing has no effect, read as zero.	R
27:22	Reserved		
21	LCD_STP	Request for LCD to Stop bus transfer.	RW
20	LCD_ACK	LCD Stop Ack.	R
19:15	Reserved		
14	UHC_SR	UHC soft reset	RW
13	OTG_SR	OTG soft reset	RW
12:9	Reserved		
8	AHB0_STP	Request for AHB0 to Stop bus transfer.	RW
7	AHB0_ACK	AHB0 Stop Ack.	R
6	Reserved		
5	AHB2_STP	Request for AHB2 to Stop bus transfer.	RW
4	AHB2_ACK	AHB2 Stop Ack.	R
3:2	Reserved		R
1	DDR0_STP	Request for DDR to Stop bus transfer.	RW
0	DDR0_AC K	DDR Stop Ack.	R

Important Note:

When USB1.1 PHY enters suspend mode, the software should set CLKGR register, stop UHC clock, and then set UHC_SR to 1.

When USB 1.1 PHY exits from suspend mode, software should clear UHC_SR to 0, then set CLKGR, and start UHC clock.

16.2.2.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SLBC																														0x100000C8
Bit	31	30	29	28	27	26	25	24	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

16.2.2.9 Sleep PC Register (SLPC)

The Sleep PC Register is a 32-bit read/write register that control sleep mode jump address.

	SLPC																														0x100000CC
Bit	31	30	29	28	27	26	25	24	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

16.2.2.10 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 0x000015C0 by reset.

	OPCR																														0x10000024
Bit	31	30	29	28	27	26	25	24	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	Reserved	MASK_VPU	Reserved			L2C_PDEN	Reserved									O1ST								SPENDN0	Reserved	CPU_MODE	O1SE	PD	ERCS	BUS_MODE	OSE
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	1		0	0	0	0	0	0

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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	Reserved		R
29	MASK_VPU	Don't change this bit , it is only used to debug	RW
28:26	Reserved		
25	L2C_PDEN	L2CC power down enable	RW
24:16	Reserved	Writing has no effect, read as zero.	R
15:8	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 otg to enter suspend mode. 0: otg has forced to entered SUSPEND mode 1: otg hasn't forced to entered SUSPEND mode	RW
6	SPENDN1	Don't change the bit. The bit should always be 1	RW
5	CPU_MODE	The default value is 0 When the value is 1, the CPU access is acclerated.	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	PD	The fief controls the state P0 in Sleep mode. 0: The P0 not power down in Sleep mode 1: The P0 power down in Sleep mode	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 bursted	RW
0	OSE	0: EXCLK enable 1: EXCLK disable When SRC_SEL == 11 of CPCCR, the software may set this bit.	RW

16.2.3 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).

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Doze is exited by software, interrupt, reset or SLEEP instruction.

16.2.4 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 IDLE 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.

16.2.5 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. 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.

16.2.6 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 VPU , X2d, software should power on these modules one by one.

16.3 Reset Control Module

16.3.1 Register Description

All RCM register 32bit access address is physical address.

Name	description	RW	Initial Value	Address	Access Size
RSR	Reset Status Register	RW	0x????????	0x10000008	32

16.3.1.1 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																												0x10000008			
Bit	31	30	29	28	27	26	25	24	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.	
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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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	
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

16.3.2 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.

16.3.3 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

17 Timer/Counter Unit

17.1 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), but only timer 0-3 is useful.
  - 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

17.2 Pin Description

Table 17-1 PWM Pins Description

Name	I/O	Description
PWM [7:0]	Output	PWM channel output signals.

17.3 Register Description

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Name	Description	RW	Reset Value	Address	Access Size
TSTR	Timer Status Register	R	0x00000000	0x100020F0	32
TSTSR	Timer Status Set Register	W	0x????????	0x100020F4	32
TSTCR	Timer Status Clear Register	W	0x????????	0x100020F8	32
TSR	Timer STOP Register	R	0x00000000	0x1000201C	32
TSSR	Timer STOP Set Register	W	0x00000000	0x1000202C	32
TSCR	Timer STOP Clear Register	W	0x0000	0x1000203C	32
TER	Timer Counter Enable Register	R	0x0000	0x10002010	16
TESR	Timer Counter Enable Set Register	W	0x????	0x10002014	16
TECR	Timer Counter Enable Clear Register	W	0x????	0x10002018	16
TFR	Timer Flag Register	R	0x003F003F	0x10002020	32
TFSR	Timer Flag Set Register	W	0x????????	0x10002024	32
TFCR	Timer Flag Clear Register	W	0x????????	0x10002028	32
TMR	Timer Mask Register	R	0x00000000	0x10002030	32
TMSR	Timer Mask Set Register	W	0x????????	0x10002034	32
TMCR	Timer Mask Clear Register	W	0x????????	0x10002038	32
TDFR0	Timer Data FULL Register 0	RW	0x????	0x10002040	16
TDHR0	Timer Data HALF Register 0	RW	0x????	0x10002044	16
TCNT0	Timer Counter 0	RW	0x????	0x10002048	16
TCSR0	Timer Control Register 0	RW	0x0000	0x1000204C	16
TDFR1	Timer Data FULL Register 1	RW	0x????	0x10002050	16
TDHR1	Timer Data HALF Register 1	RW	0x????	0x10002054	16
TCNT1	Timer Counter 1	RW	0x????	0x10002058	16
TCSR1	Timer Control Register 1	RW	0x0000	0x1000205C	16
TDFR2	Timer Data FULL Register 2	RW	0x????	0x10002060	16
TDHR2	Timer Data HALF Register 2	RW	0x????	0x10002064	16
TCNT2	Timer Counter 2	RW	0x????	0x10002068	16
TCSR2	Timer Control Register 2	RW	0x0000	0x1000206C	16
TDFR3	Timer Data FULL Register 3	RW	0x????	0x10002070	16
TDHR3	Timer Data HALF Register 3	RW	0x????	0x10002074	16
TCNT3	Timer Counter 3	RW	0x????	0x10002078	16
TCSR3	Timer Control Register 3	RW	0x0000	0x1000207C	16
TDFR4	Timer Data FULL Register 4	RW	0x????	0x10002080	16
TDHR4	Timer Data HALF Register 4	RW	0x????	0x10002084	16
TCNT4	Timer Counter 4	RW	0x????	0x10002088	16
TCSR4	Timer Control Register 4	RW	0x0000	0x1000208C	16
TDFR5	Timer Data FULL Register 5	RW	0x????	0x10002090	16
TDHR5	Timer Data HALF Register 5	RW	0x????	0x10002094	16
TCNT5	Timer Counter 5	RW	0x????	0x10002098	16

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

TCSR5	Timer Control Register 5	RW	0x0000	0x1000209C	16
TDFR6	Timer Data FULL Register 6	RW	0x????	0x100020A0	16
TDHR6	Timer Data HALF Register 6	RW	0x????	0x100020A4	16
TCNT6	Timer Counter 6	RW	0x????	0x100020A8	16
TCSR6	Timer Control Register 6	RW	0x0000	0x100020AC	16
TDFR7	Timer Data FULL Register 7	RW	0x????	0x100020B0	16
TDHR7	Timer Data HALF Register 7	RW	0x????	0x100020B4	16
TCNT7	Timer Counter 7	RW	0x????	0x100020B8	16
TCSR7	Timer Control Register 7	RW	0x0000	0x100020BC	16
TCUMOD0	Timer control mode Register 0	RW	0x????	0x10002100	16
TCUMOD3	Timer control mode Register 3	RW	0x????	0x10002110	16
TCUMOD4	Timer control mode Register 4	RW	0x????	0x10002120	16
TCUMOD5	Timer control mode Register 5	RW	0x????	0x10002130	16
TFWD0	Timer fifo write data Register 0	RW	0x????????	0x10002104	32
TFWD3	Timer fifo write data Register 3	RW	0x????????	0x10002114	32
TFWD4	Timer fifo write data Register 4	RW	0x????????	0x10002124	32
TFWD5	Timer fifo write data Register 5	RW	0x????????	0x10002134	32
TFIFOSR0	Timer fifo state Register	R	0x??	0x10002108	6
TFIFOSR3	Timer fifo state Register	R	0x??	0x10002118	6
TFIFOSR4	Timer fifo state Register	R	0x??	0x10002128	6
TFIFOSR5	Timer fifo state Register	R	0x??	0x10002138	6

17.3.1 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																**0x1000204C, 0x1000205C, 0x1000206C, 0x1000207C, 0x1000208C, 0x1000209C, 0x100020AC, 0x100020BC**															
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;	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td>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.)</td><td></td></tr>
<tr><td>10</td><td>CLRZ</td><td>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.</td><td>RW</td></tr>
<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>
</table></td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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. 0: Disable 1: Enable				RW
0	PCK_EN	Select PCLK as the timer clock input. 0: Disable 1: Enable				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

17.3.2 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																**0x10002040, 0x10002050, 0x10002060, 0x10002070, 0x10002080, 0x10002090, 0x100020A0, 0x100020B0**															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TDFR															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

17.3.3 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TDHR0, TDHR1, TDHR2, TDHR3, TDHR4, TDHR5, TDHR6, TDHR7															0x10002044, 0x10002054, 0x10002064, 0x10002074, 0x10002084, 0x10002094, 0x100020A4, 0x100020B4															
Bit																15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																TDHR															
RST																?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

17.3.4 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 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															0x10002048, 0x10002058, 0x10002068, 0x10002078, 0x10002088, 0x10002098, 0x100020A8, 0x100020B8															
Bit																15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																TCNT															
RST																?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

17.3.5 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TER																0x10002010													
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	
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. 0: Stop counting up 1: Begin counting up	R
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

17.3.6 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TESR																0x10002014													
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. 0: Ignore 1: Set TCEN 5 bit to 1	W
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

17.3.7 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TECR																0x10002018													
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. 0: Ignore 1: Set TCEN 6 bit to 0	W
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

17.3.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TFR																														0x10002020
Bit	31	30	29	28	27	26	25	24	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
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

17.3.9 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																														0x10002024
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
23:16	HFST 7~0	Set HFLAG n 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
7:0	FFST 7~0	Set FFLAG n bit of TFR. 0: Ignore 1: Set FFLAG n bit to 1	W

17.3.10 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																													0x10002028
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

17.3.11 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.

	TMR																													0x10002030
Bit	31	30	29	28	27	26	25	24	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

17.3.12 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TMSR																													0x10002034
Bit	31	30	29	28	27	26	25	24	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
27:24	FIFOEMST 3~0	Set FIFOEMST n bit of TMR. 0: Ignore 1: Set FIFOEMASK n bit to 1	
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

17.3.13 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																													0x10002038
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
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

17.3.14 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																													0x1000201C
Bit	31	30	29	28	27	26	25	24	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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: The clock supplies to timer 5 is stopped	
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
0	STOP 0	0: The clock supplies to timer 0 is supplied 1: The clock supplies to timer 0 is stopped	R

17.3.15 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																													0x1000202C
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 1: Set STOP 2 bit to 1	W
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

17.3.16 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																													0x1000203C
Bit	31	30	29	28	27	26	25	24	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	W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: Set STOP 6 bit to 0	
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
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

17.3.17 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																														0x100020F0	
Bit	31	30	29	28	27	26	25	24	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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: The counter 1 is busy now	
0	Reserved	Writing has no effect, read as zero.	R

17.3.18 Timer Status Set Register (TSTSR)

The TSTSR is a 32-bit write-only register. It contains the timer status set bits for each channel.

	TSTSR																														0x100020F4		
Bit	31	30	29	28	27	26	25	24	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

17.3.19 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																														0x100020F8		
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
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

17.3.20 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																0x10002100, 0x10002110, 0x10002120, 0x10002130												
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

17.3.21 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																**0x10002104, 0x10002114, 0x10002124, 0x10002134**															
Bit	31	30	29	28	27	26	25	24	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

17.3.22 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																**0x10002108, 0x10002118, 0x10002128, 0x10002138**															
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

17.4 Operation

17.4.1 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.

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.

17.4.2 Disable and Shutdown Operation in TCU1 Mode

1 Setting the TECR.TCCL bit to 1 to disable the TCNT.

17.4.3 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

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.

17.4.4 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.

17.4.5 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:

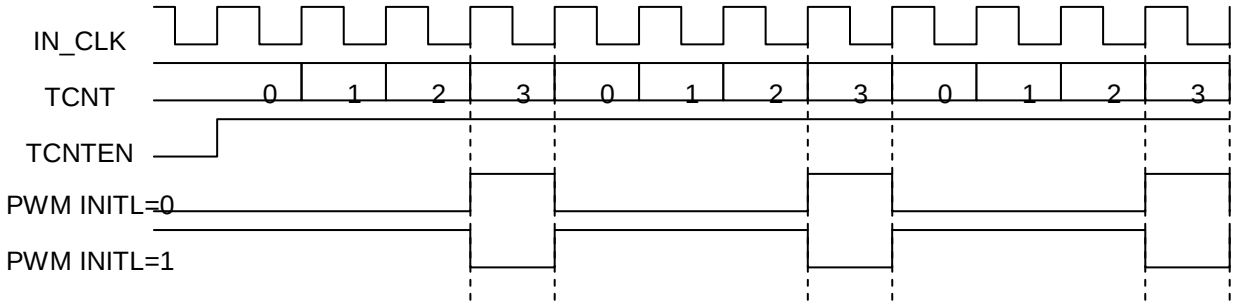
1 It suggested that (1), (2) is often used when the counter clock is very slow.
2 It suggested that (3) is often used when the counter clock is very fast.

17.4.6 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.



17.4.7 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.

17.4.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.
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(0x1000204C,0x00000002);
2. write_reg(0x10002100,0x00000001);
3. write_reg(0x10002104,0x00010001);
   write_reg(0x10002104,0x00020002);
   write_reg(0x10002104,0x00030003);        //write fifo
4. apb_write_reg(0x1000204C,0x00000082);
5. apb_write_reg(0x10002014,0x00000001);
6. if((read_reg(0x10002108) & 0x1) != 1)
   {
    write_reg(0x10002104,0x00040004);     //write fifo
   }
```

17.4.9 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(0x1000204C,0x00000002);
2. write_reg(0x10002100,0x00000001);
```


Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3. write_reg(0x10002104,0x00010001);
   write_reg(0x10002104,0x00020002);
   write_reg(0x10002104,0x00030003);
   write_reg(0x10002104,0x00040004);
   write_reg(0x10002104,0x00050005);
4. write_reg(0x10002100,0x00002A59);
   // conduct ten cycles, each cycle would match five data from fifo
5. write_reg(0x1000204C,0x00000082);
6. write_reg(0x10002014,0x00000001);
7. if(TFIFOSR. **cyc_timer ==** TCUMOD. **cyc_num**)
   {
   finish
   }

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

18 Operating System Timer

18.1 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

18.2 Register Description

In this section, we will describe the registers in OST. Following table lists all the registers definition. All OST 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
OSTDR	Operating System Timer Data Register	RW	0x????????	0x100020e0	32
OSTCNTL	Operating System Timer Counter Lower 32 Bits	RW	0x????????	0x100020e4	32
OSTCNTH	Operating System Timer Counter Higher 32 Bits	RW	0x????????	0x100020e8	32
OSTCSR	Operating System Timer Control Register	RW	0x0000	0x100020ec	16
OSTCNTH BUF	Operating System Timer Counter Higher 32 Bits Buffer	R	0x????????	0x100020fc	32

18.2.1 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

any reset.

	OSTCSR																0x100020ec													
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 PWM output. It is only used in TCU1 mode. 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
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

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

NOTE: The input clock of timer and the PCLK should keep to the rules as follows.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

18.2.2 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																													0x100020e0	
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

18.2.3 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																													0x100020e8	
Bit	31	30	29	28	27	26	25	24	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																													0x100020e4	
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

18.2.4 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																0x100020fc													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

18.3 Operation

18.3.1 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.

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.

18.3.2 Disable and Shutdown Operation

1 Setting the TECR.OSTCCL bit to 1 to disable the OSTCNT.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

19 Interrupt Controller

19.1 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

19.2 Register Description

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 19-1 INTC Register

Name	Description	RW	Reset Value	Address	Access Size
ICSR0	IRQ Source Register	R	0x00000000	0x10001000	32
ICMR0	IRQ Mask Register	RW	0xFFFFFFFF	0x10001004	32
ICMSR0	IRQ Mask Set Register	W	0x????????	0x10001008	32
ICMCR0	IRQ Mask Clear Register	W	0x????????	0x1000100C	32
ICPR0	IRQ Pending Register	R	0x00000000	0x10001010	32
ICSR1	IRQ Source Register	R	0x00000000	0x10001020	32
ICMR1	IRQ Mask Register	RW	0xFFFFFFFF	0x10001024	32
ICMSR1	IRQ Mask Set Register	W	0x????????	0x10001028	32
ICMCR1	IRQ Mask Clear Register	W	0x????????	0x1000102C	32
ICPR1	IRQ Pending Register	R	0x00000000	0x10001030	32
DSR0	IRQ Source Register0 for PDMA	R	0x00000000	0x10001034	32
DMR0	IRQ Mask Register0 for PDMA	RW	0xFFFFFFFF	0x10001038	32
DPR0	IRQ Pending Register0 for PDMA	R	0x00000000	0x1000103C	32
DSR1	IRQ Source Register1 for PDMA	R	0x00000000	0x10001040	32
DMR1	IRQ Mask Register1 for PDMA	RW	0xFFFFFFFF	0x10001044	32
DPR1	IRQ Pending Register1 for PDMA	R	0x00000000	0x10001048	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.1 Interrupt Controller Source Register (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																													0x10001000
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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 Of ICSR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

19.2.2 Interrupt Controller Source Register (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																													0x10001020
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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

Bits Of ICSR1	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

19.2.3 Interrupt Controller Mask Register (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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ICMR0																													0x10001004
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	Reserved
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 Of ICMR0	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

19.2.4 Interrupt Controller Mask Register (ICMR1)

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.

	ICMR1																													0x10001024
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	RTC
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 Of ICMR1	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

19.2.5 Interrupt Controller Mask Set Register (ICMSR0)

This register is used to set bits in the interrupt mask register. This register is write only.

	ICMSR0																													0x10001008
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits Of ICMSR0	Description
0	Ignore.
1	Will set the corresponding interrupt mask bit.

19.2.6 Interrupt Controller Mask Set Register (ICMSR1)

This register is used to set bits in the interrupt mask register. This register is write only.

	ICMSR1																													0x10001028
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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

Bits Of ICMSR1	Description
0	Ignore.
1	Will set the corresponding interrupt mask bit.

19.2.7 Interrupt Controller Mask Clear Register (ICMCR0)

This register is used to clear bits in the interrupt mask register. This register is write only.

	ICMCR0																													0x1000100c
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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 Of ICMCR0	Description
0	Ignore.
1	Will clear the corresponding interrupt mask bit.

19.2.8 Interrupt Controller Mask Clear Register (ICMCR1)

This register is used to clear bits in the interrupt mask register. This register is write only.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	ICMCR1																													0x1000102C
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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

0	Ignore.
1	Will clear the corresponding interrupt mask bit.

19.2.9 Interrupt Controller Pending Register (ICPR0)

This register contains the status of the interrupt sources after masking. This register is read only.

	ICPR0																													0x10001010
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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 Of 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.

19.2.10 Interrupt Controller Pending Register (ICPR1)

This register contains the status of the interrupt sources after masking. This register is read only.

	ICPR1																													0x10001030
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits Of 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.

19.2.11 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																													0x10001034
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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 Of DSR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

19.2.12 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																															0x10001038
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	Reserved
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 Of DMR0	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

19.2.13 Interrupt Pending Register0 for PDMA (DPR0)

This register contains the status of the interrupt sources after masking. This register is read only.

	DPR0																													0x1000103C
Bit	31	30	29	28	27	26	25	24	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	CIM0	Reserved	CIM1	TCU0	TCU1	TCU2	Reserved	EPDC	EPDCE	OTG	Reserved	X2D	SADC	GPIO0	GPIO1	GPIO2	GPIO3	GPIO4	GPIO5	GPIO6	PDMA	Reserved	SSI	Reserved		OHCI	Reserved		BCH	AIC	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 Of 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.

19.2.14 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.

	ICSR1																													0x10001040
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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

Bits Of DSR1	Description
0	The corresponding interrupt source is not pending.
1	The corresponding interrupt source is pending.

19.2.15 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																													0x10001044
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	RTC
RST	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits Of DMR1	Description
0	The corresponding interrupt is not masked.
1	The corresponding interrupt is masked.

19.2.16 Interrupt Pending Register1 for PDMA (DPR1)

This register contains the status of the interrupt sources after masking. This register is read only.

	DPR1																													0x10001048
Bit	31	30	29	28	27	26	25	24	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	VPU	PDMAM	SMB0	SMB1	SMB2	Reserved		GMAC	NEMC	Reserved	DDR	UART0	UART1	UART2	UART3	CPM	HARB0	HARB1	HARB2	Reserved	Reserved	KBC	PCM0	Reserved	Reserved	MSC0	MSC1	MSC2	Reserved	OWI	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

Bits Of 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.

19.3 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

20 Watchdog Timer

20.1 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

20.2 Register Description

In this section, we will describe the registers in WDT. Following table lists all the registers definition. All WDT 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
TDR	Watchdog Timer Data Register	RW	0x????	0x10002000	16
TCER	Watchdog Counter Enable Register	RW	0x00	0x10002004	8
TCNT	Watchdog Timer Counter	RW	0x????	0x10002008	16
TCSR	Watchdog Timer Control Register	RW	0x0000	0x1000200C	16

20.2.1 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																0x1000200C											
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>15:6</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>5:3</td><td>PRESCALE</td><td>These bits select the TCNT count clock frequency.
<table>
<tr><td>Bit 2</td><td>Bit1</td><td>Bit 0</td><td>Description</td></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.
1: Enable
0: Disable</td><td>RW</td></tr>
<tr><td>1</td><td>RTC_EN</td><td>Select RTCCLK as the timer clock input.
1: Enable
0: Disable</td><td>RW</td></tr>
<tr><td>0</td><td>PCK_EN</td><td>Select PCLK as the timer clock input.
1: Enable
0: Disable</td><td>RW</td></tr>
</table>

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

20.2.2 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	TCER																												0x10002004	
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

Note: writing TCEN to be zero can not stop the counter, you must stop it by writing TCU's register TSSR.WDTSS high.That is to say the counter can run when TCEN high and TCU's TSR low, but it can only stop by set TCU's TSR high.

20.2.3 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																												0x10002000	
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TDR															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

20.2.4 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																												0x10002008	
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	TCNT															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

20.3 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21 PDMA Controller

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.

21.1 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

21.2 Block Diagram

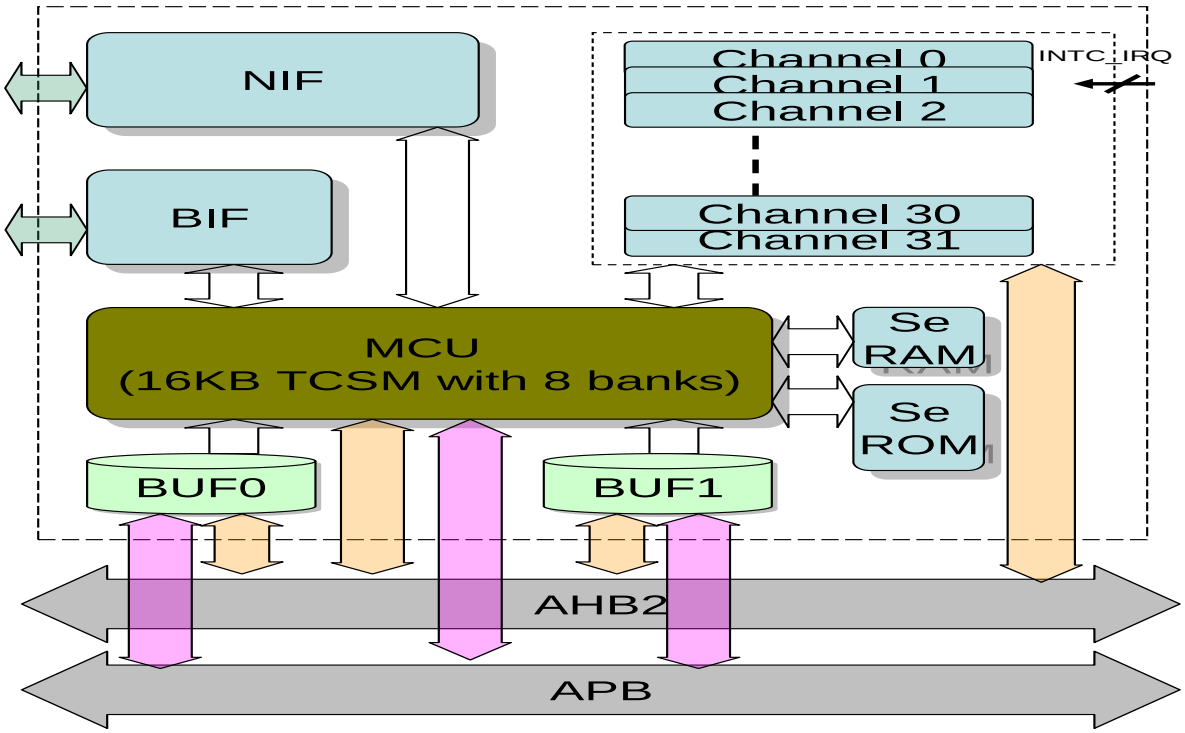


Figure 21-1 Block Diagram of PDMA

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.3 Memory Mapped Register Descriptions

21.3.1 DMA Channel Registers

Table 21-1 DMA Channel Registers (n=0~31)

Name	Description	RW	Reset Value	Address	Access Size (bit)
DSAn	Channel n Source Address 0	RW	0x0	0x13420000 + n*0x20	32
DTAn	Channel n Target Address 0	RW	0x0	0x13420004 + n*0x20	32
DTCn	Channel n Transfer Count 0	RW	0x0	0x13420008 + n*0x20	32
DRTn	Channel n Request Source 0	RW	0x0	0x1342000C + n*0x20	32
DCSn	Channel n Control/Status 0	RW	0x0	0x13420010 + n*0x20	32
DCMn	Channel n Command 0	RW	0x0	0x13420014 + n*0x20	32
DDAn	Channel n Descriptor Address 0	RW	0x0	0x13420018 + n*0x20	32
DSDn	Channel n Stride Difference 0	RW	0x0	0x1342001C + n*0x20	32

21.3.2 Global Control Registers

Table 21-2 Global Control Registers

Name	Description	RW	Reset Value	Address	Access Size (bit)
DMAC	DMA Control	R/W	0x0	0x13421000	32
DIRQP	DMA Interrupt	R	0x0	0x13421004	32
DDR	DMA Doorbell	R/W	0x0	0x13421008	32
DDRS	DMA Doorbell Set	W	0x0	0x1342100C	32
DCKE	DMA Clock Enable	W	0x0	0x13421010	32
DCKES	DMA Clock Enable Set	W	0x0	0x13421014	32
DCKEC	DMA Clock Enable Clear	W	0x0	0x13421018	32
DMACP	DMA Channel Programmable	R/W	0x0	0x1342101C	32
DSIRQP	Channel soft IRQ to MCU	R	0x0	0x13421020	32
DSIRQM	Channel soft IRQ mask	R	0xffffffff	0x13421024	32
DCIRQP	Channel IRQ to MCU	R	0x0	0x13421028	32
DCIRQM	Channel IRQ to MCU mask	R	0xffffffff	0x1342102C	32
DMCS	MCU Control and Status	RW	0xE1	0x13421030	32
DMNMB	MCU Normal MailBox	RW	?	0x13421034	32
DMSMB	MCU Security MailBox	RW	?	0x13421038	32
DMINT	MCU Interrupt	RW	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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.4 DMA Channel Register Definition

21.4.1 DMA Source Address (DSAn, n = 0 ~ 31)

	DSAn																0x13420000 + 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

21.4.2 DMA Target Address (DTAn, n = 0 ~ 31)

	DTAn																0x13420000 + 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

21.4.3 DMA Transfer Count (DTCn, n = 0 ~ 31)

	DTCn																0x13420000 + 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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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.	

21.4.4 DMA Request Types (DRTn, n = 0 ~ 31)

	DRTn																0x13420000 + 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

Table 21-3 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)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.
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.

3 To take special attention: the request type must be rewrite every time before you set the DMAC enable register, even thought the request type is the same as the last DMA transfer. If you do not rewrite the transfer type, PDMA will not make some necessary preparation and can not give out the DMA transaction.

21.4.5 DMA Channel Control/Status (DCSn, n = 0 ~ 31)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DCSn																0x13420000 + 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.	RW0
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 transfer, then software has to clear it to 0 for re-using the channel later.	RW0
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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.4.6 DMA Channel Command (DCMn, n = 0 ~ 31)

	DCMn																0x13420000 + 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
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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 21-4 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)
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
   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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

21.4.7 DMA Descriptor Address (DDAn, n = 0 ~ 31)

This register is ignored in No-Descriptor Transfer mode.

	DDAn																0x13420000 + 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

21.4.8 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																0x13420000 + 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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

21.5 DMA Global Register Definition

21.5.1 DMA Control

	DMAC																														0x13421000
Bit	31	30	29	28	27	26	25	24	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
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	RW0
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	RW0
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

21.5.2 DMA Interrupt Pending (DIRQP)

	DIRQP																														0x13421004
Bit	31	30	29	28	27	26	25	24	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.	RW0

21.5.3 DMA Doorbell (DDB)

	DDB																														0x13421008
Bit	31	30	29	28	27	26	25	24	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.	RW1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.5.4 DMA Doorbell Set (DDS)

	DBS																														0x1342100C
Bit	31	30	29	28	27	26	25	24	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	W1

21.5.5 DMA Channel Programmable (DMACP)

	DMACP																														0x1342101C
Bit	31	30	29	28	27	26	25	24	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

Bits	Name	Description	RW
31:0	DCPn	Channel programmable enable. 0, compatible with previous JZ SOC; 1, firmware controlled channel	RW

21.5.6 DMA Soft IRQ Pending

	DSIRQP																														0x13421020
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.5.7 DMA Soft IRQ Mask

	DSIRQM																0x13421024													
Bit	31	30	29	28	27	26	25	24	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

21.5.8 DMA Channel IRQ Pending to MCU

	DCIRQP																0x13421028													
Bit	31	30	29	28	27	26	25	24	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

21.5.9 DMA Channel IRQ to MCU Mask

	DCIRQM																0x1342102C													
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

21.5.10 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 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.

21.5.11 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.

21.6 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.

21.6.1 MCU Control & Status

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	DMCS																													0x13421030
Bit	31	30	29	28	27	26	25	24	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	Reserved		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
4	BTB_INV	Writing 1 invalidates BTB in MCU. Writing 0 has no effect, read as zero.	W
3	SC_CALL	SecurityCall. Writing 1 triggers a security exception which directs MCU to run security code located in the on-chip security ROM., read as zero.	W1
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.

21.6.2 MCU Normal MailBox

	DMNMB																													0x13421034
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.6.3 MCU Security MailBox

	DMSMB																0x13421038															
Bit	31	30	29	28	27	26	25	24	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 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

21.6.4 MCU Interrupt

	DMINT																0x1342103C															
Bit	31	30	29	28	27	26	25	24	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.	RW0
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

21.6.5 Multiple Bank Tightly Coupled Sharing Memory

In MCU, there is a Tightly Coupled Sharing Memory (TCSM) with 24KB size and total 6 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 21-5 TCSM space

	VA (only available for MCU)	PA (not remapped)	PA (remapped)
BANK0	F4000000h ~ F4000FFFh	13422000h ~ 13422FFFh	F3422000h ~ F3422FFFh
BANK1	F4001000h ~ F401FFFFh	13423000h ~ 1343FFFFh	F3423000h ~ F343FFFFh
BANK2	F4002000h ~ F4002FFFh	13424000h ~ 13424FFFh	F3424000h ~ F3424FFFh
BANK3	F4003000h ~ F4003FFFh	13425000h ~ 13425FFFh	F3425000h ~ F3425FFFh
BANK4	F4004000h ~ F4004FFFh	13426000h ~ 13426FFFh	F3426000h ~ F3426FFFh
BANK5	F4005000h ~ F4005FFFh	13427000h ~ 13427FFFh	F3427000h ~ F3427FFFh

21.6.6 Security ROM & Security RAM

In the PDMA, there is a 32KB on-chip security ROM with start programming address at 0xF5000000 and an 4KB on-chip security RAM with start programming address at 0xF5010000. Both the ROM and the RAM are only accessible in SMD mode by MCU.

21.6.7 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 31~3: reserved, read as zero 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 SecurityCall 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*

21.6.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

21.6.9 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

21.6.10 SecurityCall Accepted by MCU

SecurityCall is an especial exception that can be acknowledged by MCU. There are only two modes for MCU: normal mode and security mode that can be identified by DMCS.SCMD. For convenience of description, simplified term **NMD** (normal mode) and **SMD** (security mode) will be used in later statements. When MCU is running in NMD mode (DMCS.SCMD == 0), a SecurityCall exception can be granted immediately when both CP0.STATUS.ERL and CP0.SATUS.EXL are zero. In detail, do as following steps:

1. load SecurityCall relative data structure to the proper location of TCSM, must be 256-byte aligned
2. set DMCS.SC_OFF to point to the position (0xF4000000 + DMCS.SC_OFF<<8).
3. write 1 to DMCS.SC_CALL to try to trigger a SecurityCall exception.
4. When DMCS.SCMD == 1, which means SecurityCall is successfully acknowledged, the handler entry is 0xF5000000.

Note that active DMCS.SCMD always inhibit new attempt of SecurityCall, thus, the service routine for SecurityCall is not re-enterable, therefore OS running in main CPU must carefully manage the SecurityCall through such as semaphore mechanism.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21.6.11 Interruptable SMD Mode

When SecurityCall is acknowledged, CP0.SATUS.EXL also will be automatically set 1 to inhibit MCU responding any active IRQ to MCU meanwhile CP0.EPC will record the interrupted PC located in TCSM. However, since some functions of SecurityCall may spend long time, so code running in SMD mode should permit being interrupted by IRQ coming occasionally. Therefore, some service routines in Security ROM should clear CP0.SATUS.EXL to 0 if necessary, and after that, once an IRQ is acknowledged in SMD mode, the corresponding handler entry becomes 0xF5000100 other than 0xF4000100, meanwhile an SW invisible flag SC_IF will be automatically set 1 and a special CP0 register CP0.SC_EPC (only accessible by MFC0/MTC0 in SMD mode) will record the interrupted PC located in Security ROM. The code sequence located at 0XF5000100 should be elaborately constructed like following:

```
1.  subu sp, sp, 4*32          //prepare to push GPR
2.  mfc0 k0, $14, 0            //EPC (return PC in NMD mode) to k0
3.  mfc0 k1, $24, 0            //SePC (return PC in SMD mode) to k1
4.  sw ra, 4*31(sp)            //push ra
5.  jal _push_gpr              //subroutine of pushing GPR (except ra)
6.  nop
7.  la k0, _se_ram_top         //top of Security RAM
8.  sw sp, 0(k0)               //sp in SMD mode is always saved at top of Security RAM
9.  la k0, RET_POS             //exit after IRQ in SMD mode to k0
10. la k1, 0xF4000100          //IRQ entry in NMD mode to k1
11. mtc0 k1, $14, 0
12. mtc0 k0, $24, 0
13. MACRO_OF_CHAOS_GPR
14. la sp, _se_ram_top
15. subu sp, sp, 4*32
16. lw sp, 4*26(sp)            //get sp in NMD mode
17. eret                       //*1
RET_POS:                       //*2
18. la k0, _se_ram_top
19. jal _pop_gpr               //subroutine of poping GPR (except ra)
20. lw sp, 0(k0)               //restore sp in SMD mode
21. lw ra, 4*31(sp)            //pop ra
22. addu sp, sp, 4*32
23. mtc0 k0, $14, 0
24. mtc0 k1, $24, 0
25. mfc0 k0, $12, 0
26. mtc0 k1, $30, 0            //SePC to ErrPC
27. ori k0, k0, 4
28. mtc0 k0, $12, 0            //set ERL to Status
29. ERET                       //*3
```


Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTES:

1 After the execution of ERET, MCU enters NMD mode, SC_IF retains 1 and CP0.STATUS.EXL retains 1

2 When SC_IF is active 1 and MCU is in NMD mode, executing ERET always restores SMD mode and returns to the PC pointed by CP0.SC_EPC, furthermore, HW will automatically set 1 to CP0.STATUS.EXL and set 0 to CP0.STATUS.ERL and set 0 to SC_IF. Note that SC_IF being 1 always inhibits any new attempt of SecurityCAll.

3 When CP0.STATUS.ERL is 1, ERET does not clear SMD mode. Therefore, for code in Security ROM, DO NOT set 1 to CP0.STATUS.ERL through MTC0 instruction except here.

21.7 DMA manipulation

21.7.1 Descriptor Transfer Mode

21.7.1.1 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 21-6 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.

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 21-6 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

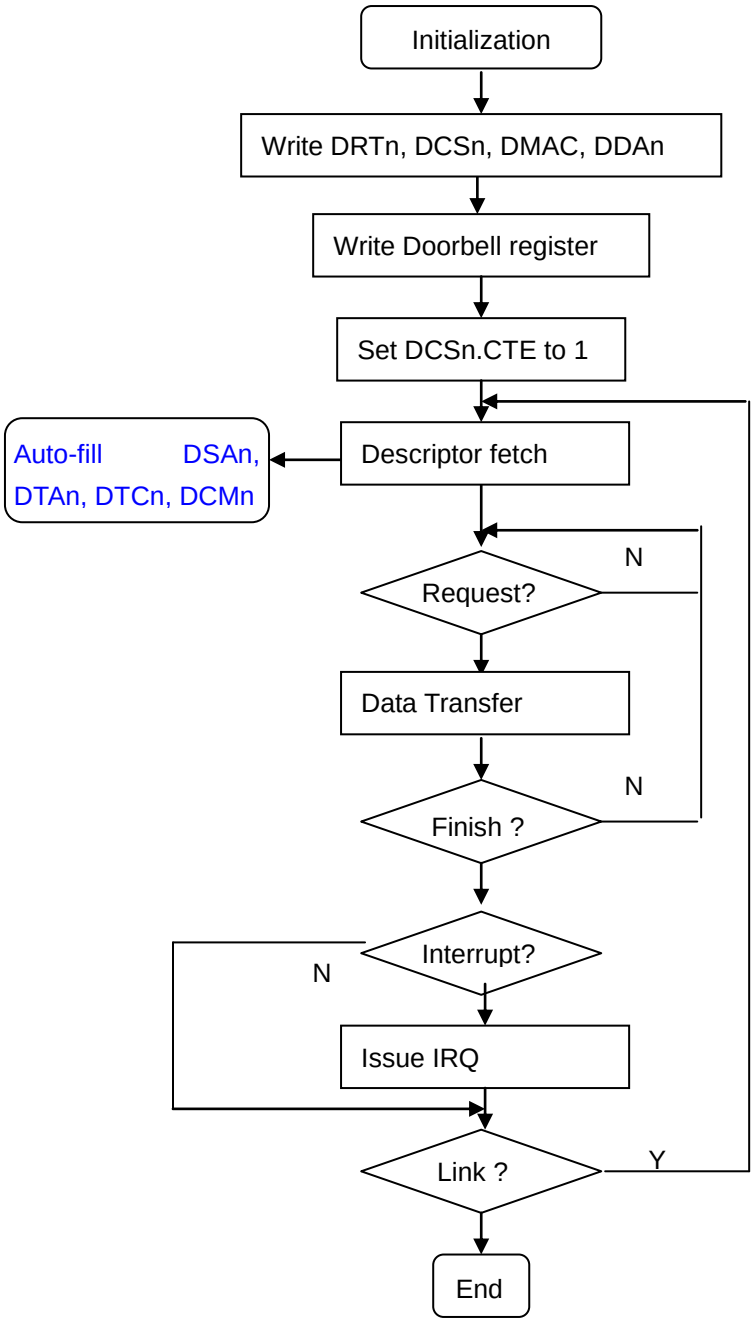


Figure 21-2 Descriptor Transfer Flow

21.7.1.2 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

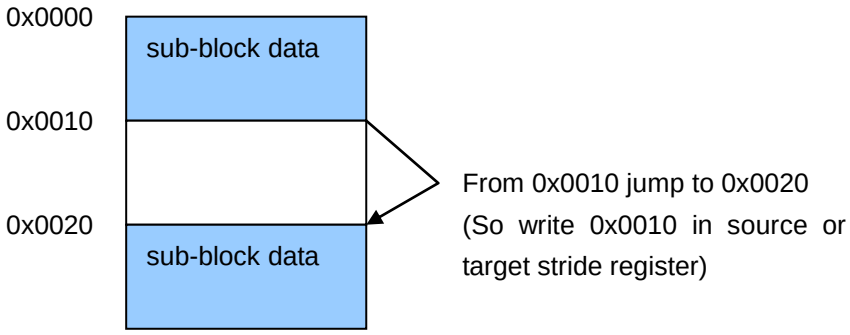


Figure 21-3 Example for Stride Transfer Mode

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.

21.7.2 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.

21.8 DMA Requests

DMA transfer requests are normally generated from either the data transfer source or target, but

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

21.8.1 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.

21.8.2 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.

21.9 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

22 SAR A/D Controller

22.1 Overview

The A/D is CMOS low-power dissipation 12bit touch screen SAR analog to digital converter. It operates with 3.3/1.2V power supply. It is developed as an embedded high resolution ADC targeting to the 65nm 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 three different modes: Touch Screen (measure pen position and pen down pressure), Battery (check the battery power), and two auxiliary input. Touch Screen can transfer the data to memory through the DMA or CPU. Battery and two auxiliary input can transfer the data to memory through CPU.

Features:

- 7 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 4-wire and 5-wire touch panel measurement (Through pin XP, XN, YP, YN and AUX2)
- Support multi-touch detect
- Support write control command by software
- Support voltage measurement (Through pin VBAT)
- Support two auxiliary input (Through pin AUX1, AUX2)
- Single-end and Differential Conversion Mode
- Auto X/Y, X/Y/Z1/Z2 and X/Y/Z1/Z2/X2/Y2 position measurement
- Support external touch screen controller

22.2 Pin Description

Table 22-1 SADC Pin Description

Name	I/O	Description
XN	AI	Touch screen analog differential X- position input
YN	AI	Touch screen analog differential Y- position input
XP	AI	Touch screen analog differential X+ position input
YP	AI	Touch screen analog differential Y+ position input

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

VBAT	AI	VBAT direct input *[1]
AUX1	AI	Auxiliary Input
AUX2	AI	Auxiliary Input

NOTE:

*[1]: Users who already deployed resistor networks on board level can use VBAT to direct measure the battery value.

22.3 Register Description

In this section, we will describe the registers in SAR A/D controller. Following table lists all the register definitions. All registers' 32bit addresses are physical addresses. And detailed function of each register will be described below.

Table 22-2 SADC Register Description

Name	Description	RW	Reset Value	Address	Access Size
ADENA	ADC Enable Register	RW	0x80	0x10070000	8
ADCFG	ADC Configure Register	RW	0x00040000	0x10070004	32
ADCTRL	ADC Control Register	RW	0x3F	0x10070008	8
ADSTATE	ADC Status Register	RW	0x00	0x1007000C	8
ADSAME	ADC Same Point Time Register	RW	0x0000	0x10070010	16
ADWAIT	ADC Wait Time Register	RW	0x0000	0x10070014	16
ADTCH	ADC Touch Screen Data Register	RW	0x00000000	0x10070018	32
ADVDAT	ADC VBAT Data Register	RW	0x0000	0x1007001C	16
ADADAT	ADC AUX Data Register	RW	0x0000	0x10070020	16
ADCLK	ADC Clock Divide Register	RW	0x00041000	0x10070028	32
ADCMD	ADC Command Register	RW	0x00000000	0x10070024	32
ADSTB	ADC stable Register	RW	0x00000000	0x10070034	32

22.3.1 ADC Enable Register (ADENA)

The register ADENA is used to trigger A/D to work.

	ADENA																														0x10070000			
Bit	31	30	29	28	27	26	25	24	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	SLP_MD	Reserved		PENDEN	TCHEN	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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	SLP_MD	SLEEP Mode Control. 1:Enter sleep mode 0:Exit sleep mode	RW
5:4	Reserved	Writing has no effect, read as zero.	R
3	PENDEN	Pen Down Detect Enable control. 0: disable 1: enable	RW
2	TCHEN	Touch Screen Enable Control. 0: disable 1: enable	RW
1	VBATEN	VBAT Enable Control. No matter TCHEN is 1 or 0, VBATEN can be set to 1 to sample the voltage of battery, and when the value of voltage is ready, PBATEN will be cleared by hardware auto.	RW
0	AUXEN	AUX n Enable Control. No matter TCHEN is 1 or 0, AUXEN can be set to 1 to sample the voltage of AUX1, AU2 or AUX3, and when the value of voltage is ready. AUXEN will be cleared by hardware auto.	RW

NOTES:

1 TCHEN, VBATEN and AUXEN can be set to 1 at the same time. The priority of the three mode is AUX > VBAT > TCH.

2 SLP_MD, TCHEN can be set to 1 at the same time. The priority of the two mode is SLP_MD > TCH.

3 When VBATEN and AUXEN are all 0, SLP_MD can be set to 1.

22.3.2 ADC Configure Register (ADCFG)

The register ADCFG is used to configure the A/D.

	ADCFG																														0x10007004	
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPZZ	Reserved							Wire_cel_r	Cmd_sel_r	PRU						DMA_EN	XYZ		SNUM			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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31</td><td>SPZZ[*1]</td><td>The $X_dY_dZ_mZ_n$ of different point measure can be different.
But the $X_dY_dZ_mZ_n$ of the same point measure can be same or different.
0: The $X_dY_dZ_mZ_n$ of the same point measure is all the same
(X_dY_dZ1Z2, X_dY_dZ1Z2, X_dY_dZ1Z2, X_dY_dZ1Z2 … X_dY_dZ1Z2)
1: The $X_dY_dZ_mZ_n$ of the same point measure maybe different
(X_dY_dZ1Z2, X_dY_dZ3Z4, X_dY_dZ3Z4, X_dY_dZ1Z2 … X_dY_dZ1Z2)</td><td>RW</td></tr>
<tr><td>30:24</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>23</td><td>WIRE_SEL</td><td>0: use 4-wire touch panel
1: use 5-wire touch panel</td><td>RW</td></tr>
<tr><td>22</td><td>CMD_SEL</td><td>0: use hardware inter command to control touch panel
1: use software write command to control touch panel</td><td>RW</td></tr>
<tr><td>21:16</td><td>RPU</td><td>Internal Pull-up resistor for Pen Detection.
6’b111111: 64kΩ/63 = 1.02kΩ (least sensitive)
6’b111110: 64KΩ/62 = 1.03KΩ
… (pull-up = 64kΩ / binary value of RPU)
6’b000010: 64KΩ/2 = 32KΩ
6’b000001: 64kΩ/1 = 64kΩ (most sensitive) default
6’b000000: RESERVED (do not use this setting)</td><td>RW</td></tr>
<tr><td>15</td><td>DMA_EN</td><td>When A/D is used as Touch Screen , DMA_EN is used as follows:
0: The sample data is read by CPU
1: The sample data is read by DMA</td><td>RW</td></tr>
<tr><td>14:13</td><td>XYZ</td><td>When A/D is used in Touch Screen mode, XYZ is used as follows:
<table>
<tr><th>XYZ</th><th>Measure</th></tr>
<tr><td>00</td><td>$X_s \rightarrow Y_s$</td></tr>
<tr><td>01</td><td>$X_d \rightarrow Y_d$</td></tr>
<tr><td>10</td><td>$X_d \rightarrow Y_d \rightarrow Z1_d \rightarrow Z2_d$ or $X_d \rightarrow Y_d \rightarrow Z3_d \rightarrow Z4_d$</td></tr>
<tr><td>11</td><td>$X_d \rightarrow Y_d \rightarrow Z1_d \rightarrow Z2_d \rightarrow X2 \rightarrow Y2$ or
$X_d \rightarrow Y_d \rightarrow Z3_d \rightarrow Z4_d \rightarrow X2 \rightarrow Y2$</td></tr>
</table></td><td>RW</td></tr>
<tr><td>12:10</td><td>SNUM</td><td>The number of repeated sampling one point. When A/D is used as Touch Screen, SNUM is used as follows:
<table>
<tr><th>SNUM</th><th>Number</th></tr>
<tr><td>000</td><td>Reserved</td></tr>
<tr><td>001</td><td>1</td></tr>
<tr><td>010</td><td>2</td></tr>
<tr><td>011</td><td>3</td></tr>
<tr><td>100</td><td>4</td></tr>
<tr><td>101</td><td>5</td></tr>
<tr><td>110</td><td>Reserved</td></tr>
<tr><td>111</td><td>Reserved</td></tr>
</table></td><td>RW</td></tr>
<tr><td>9:2</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1:0	CMD	CMD is used to choose the current sample command when ADENA.AUXEN is set to 1. CMD: 00 Reserved; 01 Measure AUX1 voltage; 10 Measure AUX2 voltage; 11 Reserved	RW

NOTE:

*1: X_s, Y_s means the reference mode of X, Y is single-end mode.

X_d, Y_d, $Z1_d$, $Z2_d$, $Z3_d$, Z4d means the reference mode of X, Y, Z1, Z2, Z3, Z4 is differential mode.

When you measure Xs you need to make sure that X-plate is driven by external DC power.

When you measure Ys you need to make sure that Y-plate is driven by external DC power.

22.3.3 ADC Control Register (ADCTRL)

The register ADCTRL is used to control A/D to work.

ADCTRL																												0x10070008					
Bit	31	30	29	28	27	26	25	24	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																										SLPENDM	PENDM	PENUM	DTCHM	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	SLPENDM	In SLEEP mode pen down interrupt mask. 0: enabled 1: masked	RW
4	PENDM	Pen down interrupt mask. 0: enabled 1: masked	RW
3	PENUM	Pen up interrupt mask. 0: enabled 1: masked	RW
2	DTCHM	Touch Screen Data Ready interrupt mask. 0: enabled 1: masked	RW
1	VRDYM	VBAT data ready interrupt mask. 0: enabled	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: masked	
0	ARDYM	AUX data ready interrupt mask. 0: enabled 1: masked	RW

22.3.4 ADC Status Register (ADSTATE)

The register ADSTATE is used to keep the status of A/D.

	ADSTATE																													0x1007000C
Bit	31	30	29	28	27	26	25	24	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																								SLP_RDY	Reserved	SLPEND	PEND	PENU	DTCH	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
31:8	Reserved	Writing has no effect, read as zero.	R
7	SLP_RDY	Sleep state bit. 1:The set of sleep mode is ready 0:The set of sleep mode is not ready	R
6	Reserved	Writing has no effect, read as zero.	R
5	SLPEND	In SLEEP mode pen down interrupt flag. Write 1 to this bit, the bit will clear this bit. 1: active 0: not active	RW
4	PEND	Pen down interrupt flag. Write 1 to this bit, the bit will clear this bit. 1: active 0: not active	RW
3	PENU	Pen up interrupt flag. Write 1 to this bit, the bit will clear this bit. 1: active 0: not active	RW
2	DTCH	Touch screen data ready interrupt flag. Write 1 to this bit, the bit will clear this bit. 1: active 0: not active	RW
1	VRDY	VBAT data ready interrupt flag. Write 1 to this bit, the bit will clear this bit. 1: active 0: not active	RW
0	ARDY	AUX data ready interrupt flag. Write 1 to this bit, the bit will	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		clear this bit. 1: active 0: not active	

22.3.5 ADC Same Point Time Register (ADSAME)

The register ADSAME is used to store the interval time between repeated sampling the same point. The clock of the counter is clk_us.

	ADSAME																														0x10070010
Bit	31	30	29	28	27	26	25	24	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																SCNT															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

22.3.6 ADC Wait Pen Down Time Register (ADWAIT)

The register ADWAIT is used to store the interval time of wait pen down. And the register can be used as the interval time among the different point. The clock of the counter is clk_ms.

	ADWAIT																														0x10070014
Bit	31	30	29	28	27	26	25	24	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																WCNT															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

22.3.7 ADC Touch Screen Data Register (ADTCH)

The read-only ADTCH is corresponded to 16x32 bit FIFO, it keep the sample data for touch screen. 0~11 bits are data, 15 bit is data type. 16~27 bits are data, 31 bit is data type. When write to the register, DATA will be clear to 0.

	ADTCH																														0x10070018
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE1	Reserved			TDATA1												TYPE0	Reserved			TDATA0											
RST	0	0	0	0	0	0	0	0	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31	TYPE1	Type of the Touch Screen Data1. When A/D is used as Touch Screen, ADCFG.XYZ=10 or XYZ=11: TYPE1=1: X_d→Y_d→Z1→Z2 or X_d→Y_d→Z1→Z2→X2→Y2; TYPE1=0: X_d→Y_d→Z3→Z4 or X_d→Y_d→Z3→Z4→X2→Y2. When A/D is used as Touch Screen, ADCFG.XYZ=00 or XYZ=01: TYPE1=0.	RW
30:28	Reserved	Writing has no effect, read as zero.	R
27:16	TDATA1	The concert data of touch screen A/D.	RW
15	TYPE0	Type of the Touch Screen Data2. When A/D is used as Touch Screen, ADCFG.XYZ=10 or XYZ=11: TYPE0=1: X_d→Y_d→Z1→Z2 or X_d→Y_d→Z1→Z2→X2→Y2; TYPE0=0: X_d→Y_d→Z3→Z4 or X_d→Y_d→Z3→Z4→X2→Y2. When A/D is used as Touch Screen, ADCFG.XYZ=00 or XYZ=01: TYPE0=0.	RW
14:12	Reserved	Writing has no effect, read as zero.	R
11:0	TDATA0	The concert data of touch screen A/D.	RW

NOTES:

1 When A/D is used as Touch Screen, ADCFG.XYZ=00.

The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
0	000	Y_s	0	000	X_s

2 When A/D is used as Touch Screen, ADCFG.XYZ=01.

The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
0	000	Y_d	0	000	X_d

3 When A/D is used as Touch Screen, ADCFG.XYZ=10.TYPE=1.

The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
1	000	Y_d	1	000	X_d
1	000	$Z2_d$	1	000	$Z1_d$

Users need to read twice to get the whole data. The first time reading gets the data Yd and Xd. The second time reading gets the data Z2d and Z1d.

The touch pressure measurement formula is as follows: (You can use formula 1 or formula

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2.)

$$R_{TOUCH} = R_{X\text{-}Plate} \bullet \frac{X\text{-}Position}{4096}\left(\frac{Z_2}{Z_1} - 1\right) \quad (1)^{*1}$$

$$R_{TOUCH} = \frac{R_{X\text{-}Plate} \bullet X\text{-}Position}{4096}\left(\frac{4096}{Z_1} - 1\right) - R_{Y\text{-}Plate} \bullet \left(1 - \frac{Y\text{-}Position}{4096}\right) \quad (2)^{*1}$$

4 When A/D is used as Touch Screen, ADCFG.XYZ=10.TYPE=0.
The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
0	000	Y_d	0	000	X_d
0	000	$Z4_d$	0	000	$Z3_d$

Users need to read twice to get the whole data. The first time reading gets the data Y_d and X_d. The second time reading gets the data $Z4_d$ and $Z3_d$.

The touch pressure measurement formula is as follows: (You can use formula 3 or formula 4.)

$$R_{TOUCH} = R_{Y\text{-}Plate} \bullet \frac{Y\text{-}Position}{4096}\left(\frac{Z_4}{Z_3} - 1\right) \quad (3)^{*1}$$

$$R_{TOUCH} = \frac{R_{Y\text{-}Plate} \bullet Y\text{-}Position}{4096}\left(\frac{4096}{Z_3} - 1\right) - R_{X\text{-}Plate} \bullet \left(1 - \frac{X\text{-}Position}{4096}\right) \quad (4)^{*1}$$

5 When A/D is used as Touch Screen, ADCFG.XYZ=11.TYPE=1.
The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
1	000	Y_d	1	000	X_d
1	000	$Z2_d$	1	000	$Z1_d$
1	000	Y2	1	000	X2

Users need to read thrice to get the whole data. The first time reading gets the data Y_d and X_d. The second time reading gets the data $Z2_d$ and $Z1_d$. The third time reading gets the data Y2 and X2.

6 When A/D is used as Touch Screen, ADCFG.XYZ=11.TYPE=0.
The format of touch screen data is as follows:

Type1	Reserved	Data1	Type0	Reserved	Data0
0	000	Y_d	0	000	X_d
0	000	$Z4_d$	0	000	$Z3_d$
0	000	Y2	0	000	X2

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Users need to read thrice to get the whole data. The first time reading gets the data Y_d and X_d. The second time reading gets the data $Z4_d$ and $Z3_d$. The third time reading gets the data Y2 and X2.

NOTE:

*1: To determine pen or finger touch, the pressure of the touch needs to be determined. Generally, it is not necessary to have very high performance for this test; therefore, the 8-bit resolution mode is recommended (however, calculations will be shown here are in 12bit resolution mode).

$R_{X\text{-plate}}$: Total X-axis resistor value (about 200Ω~ 600Ω)

$R_{Y\text{-plate}}$: Total Y-axis resistor value (about 200Ω~ 600Ω)

X-Position: X-axis voltage sample value

Y-Position: Y-axis voltage sample value

Z1, Z2: Z1, Z2 voltage sample value

Z3, Z4: Z3, Z4 voltage sample value

X2, Y2: X2, Y2 voltage sample value

22.3.8 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																													0x1007001C
Bit	31	30	29	28	27	26	25	24	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 calculate as follow formula:

Vbat = (slope1/5000 + 0.2932)*VDATA + intercept1/10+9.51

Vbat = (slope2/5000 + 0.2939)*VDATA + intercept2/8+5.2

slope1 equals to EFUSE2 second low 8bit, and this is a signed value

intercept1 equals to EFUSE2 first low 8bit , this also a signed value

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

slope2 and intercept2 like slope1/intercept1, there are also singed value, but there value are EFUSE4 second low 8bit and EFUSE4 first low 8bit individually.

22.3.9 ADC AUX Data Register (ADADAT)

The read-only ADADAT is a 16-bit register, it keep the sample data. 0~11 bits are data.

	ADSDAT																													0x10070020
Bit	31	30	29	28	27	26	25	24	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:

$$V_{SAD} = \frac{ADATA}{4096} \bullet AVDD33$$

22.3.10 ADC Clock Divide Register (ADCLK)

The register ADCLK is used to set the A/D's clock dividing number.

	ADCLK																													0x10070028
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLKDIV_MS																CLKDIV_US								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	CLKDIV_MS	Dividing number to get ms clock from ADC clock. ms_clk = us_clk / (CLK_MS +1)	RW
15:8	CLKDIV_US	Dividing number to get us clock from ADC clock. us_clk = adc_clk / (CLKDIV_US+1) 0 ≤CLKDIV_10 ≤127	RW
7:0	CLKDIV	Dividing number to get ADC clock from device clock. The A/D works at the frequency between 20KHz and 200KHz.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		If CLKDIV = N, Then the freq of adc_clk = dev_clk / (N+1). 0 ≤ N ≤ 255	

22.3.11 ADC Command Register (ADCMD)

ADC Command Register ADCMD is used for write touch screen control command by software. Then, if the cmd_sel_r = 1, the controller will read ADCMD's command, then use command to control touch screen. The controller has 32x32 bit FIFO to store commands, the command format like this.

	ADCMD																0x10070024														
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIL	RPU					XPSUP	XNSUP	YPSUP	XPGRU	XNGRU	YNGRU	VREFNAUX	VREFNXN	VREFNXP	VREFNYN	VREFPVDD33	VREFPAUX	VREFPXN	VREFPXP	VREFPYP	XPADC	XNADC	YPADC	YNADC	WIPEADC	AUX2ADC	AUX1ADC	RPUWP	RPUXP	RPUYP	APIL
RST	0	0	0	0	0	0	0	0	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	PIL	Current used for pressure measurement. 0: Ip = 200μA(default) 1: Ip = 400μA	RW
30:26	RPU	Internal Pull-up resistor for Pen Detection.	RW
25	XPSUP	XP to TPVDD control Switch. 0: open; 1:close.	RW
24	XNSUP	XN to TPGND control Switch. 0: open; 1:close.	RW
23	YPSUP	YP to TPVDD control Switch. 0: open; 1:close.	RW
22	XPGRU	XP to TPGND control Switch. 0: open; 1:close.	RW
21	XNGRU	XN to TPGND control Switch. 0: open; 1:close.	RW
20	YNGRU	YN to TPGND control Switch. 0: open; 1:close.	RW
19	VREFNAUX	ADC low voltage reference to AUX control switch. 0: open; 1: close.	RW
18	VREFNXN	ADC low voltage reference to XN control switch. 0: open; 1: close.	RW
17	VREFNXP	ADC low voltage reference to XP control switch. 0: open; 1: close.	RW
16	VREFNYN	ADC low voltage reference to YN control switch. 0: open; 1: close.	RW
15	VREFPVDD33	ADC high voltage reference to VDD control switch. 0: open; 1: close.	RW
14	VREFPAUX	ADC high voltage reference to AUX control switch. 0: open; 1: close.	RW
13	VREFPXN	ADC high voltage reference to XN control switch. 0: open; 1: close.	RW
12	VREFPXP	ADC high voltage reference to XP control switch. 0: open; 1: close.	RW
11	VREFPYP	ADC high voltage reference to YP control switch. 0: open; 1: close.	RW
10	XPADC	Use XP as ADC input channel control switch. 0: open; 1: close.	RW
9	XNADC	Use XN as ADC input channel control switch. 0: open; 1: close.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8	YPADC	Use YP as ADC input channel control switch. 0: open; 1: close.	RW
7	YNADC	Use YN as ADC input channel control switch. 0: open; 1: close.	RW
6	WIPEADC	Use WIPE as ADC input channel control switch. 0: open; 1: close.	RW
5	AUX2ADC	Use AUX2 as ADC input channel control switch. 0: open; 1: close.	RW
4	AUX1ADC	Use AUX1 as ADC input channel control switch. 0: open; 1: close.	RW
3	RPUWP	Connect WP to RPU control switch. 0: open; 1: close.	RW
2	RPUXP	Connect XP to RPU control switch. 0: open; 1: close.	RW
1	RPUYP	Connect YP to RPU control switch. 0: open; 1: close.	RW
0	APIL	Use inter current source control switch. 0: open; 1: close.	RW

22.3.12 ADC Stable Register (ADCSTB)

ADC Stable register is used for delay some ADC clock cycle to wait ADC stable when sampling VBAT and AUX voltage.

	ADCSTB																														0x10070034
Bit	31	30	29	28	27	26	25	24	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

22.4 SAR A/D Controller Guide

The following describes steps of using SAR-ADC.

22.4.1 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 Touch Screen, VBAT and AUX.

3 When you want to power down SADC to get lower power, you should disable Touch Screen, VBAT and AUX, and then set ADENA.POWER to 1.

22.4.2 A Sample Touch Screen Operation

(Pen Down → Sample some data of several points → Pen Up)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

1 Set ADCTRL to 0x1f to mask all the interrupt of SADC.
2 Set DMA_EN to choose whether to use DMA to read the sample data out or to use CPU to read the sample data out.
3 Set ADCFG.RPU to choose the Internal Pull-up resistor for Pen Detection.
4 Set ADCFG.WIRE_SEL to choose 4-wire or 5-wire mode in pendown detect.
5 Set ADCFG.CMD_SEL to choose use hardware inter command or software command control touch screen. If you want to use software command, you must write your command by ADCMD register. And if you want to use hardware inter command, please straight to set ADCFG.SPZZ.
6 Set ADCFG.SPZZ and ADCFG.XYZ to choose sample mode.
   a $X_s \rightarrow Y_s$ (Single-end X→Single-end Y).
   b $X_d \rightarrow Y_d$ (Differential X→Differential Y).
   c $X_d \rightarrow Y_d \rightarrow Z1_d \rightarrow Z2_d$ or $X_d \rightarrow Y_d \rightarrow Z3_d \rightarrow Z4_d$ (Reference register ADCFG.SPZZ)
   (Differential X→Differential Y→Differential Z1→Differential Z2 or
   Differential X→Differential Y→Differential Z3→Differential Z4).
   d $X_d \rightarrow Y_d \rightarrow Z1_d \rightarrow Z2_d \rightarrow X2 \rightarrow Y2$ or $X_d \rightarrow Y_d \rightarrow Z3_d \rightarrow Z4_d \rightarrow X2 \rightarrow Y2$ (Reference register bit SPZZ).
7 Set ADCFG.SNUM to choose one point sampling times.
8 Set ADCLK.CLKDIV, ADCLK.CLKDIV_US and ADCLK.CLKDIV_MS to set A/D clock frequency.
9 Set ADWAIT to decide the wait time of pen down and the interval time between sampling different points. This time delay is necessary because when pen is put down or pen position change, there should be some time to wait the pen down signal to become stable.
10 Set ADSAME to decide the interval time between repeated sampling the same point. User can repeat sampling one point to get the most accurate data.
11 Set ADCTRL.PENDM to 0 to enable the pen down interrupt of touch panel.
12 Set ADENA.TCHEN to 1 to start touch panel.
13 When pen down interrupt happened, you should set ADCTRL.PENDM to 1 and clear ADSTATE.PEND to close pen down interrupt. Then you should clear ADSTATE.PENDU and set ADCTRL.PENUM to 0 to enable pen up interrupt.
14 When pen down interrupt happened, the SARADC is sampling data. When ADSTATE.DTCH to 1, user must read the sample data from ADTCH. The SARADC will not sample the next point until the whole data of the one point are read (no matter by CPU or DMA). If ADCFG.XYZ is mode zero and mode one, user needs to read 1*ADCFG.SNUM times to get the whole data. In mode two, user needs to read 2*ADCFG.SNUM times to get the whole data. And in mode three, user needs to read 3*ADCFG.SNUM times to get all data.
15 Repeat 14 till pen up interrupt happened.
16 When pen up interrupt happened, you should set ADCTRL.PENUM to 1 and clear ADSTATE.PENU. Then you should clear ADSTATE.PEND and set ADCTRL.PENDM to 0 to enable pen down interrupt.
17 Wait pen down interrupt and repeat from 13.
18 When you want to shut down the touch screen, user can set the ADENA.TCHEN to 0. If the

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

last point is not sampled completely, user can abandon it.

22.4.3 SLEEP mode Sample Operation

1 If the register ADCLK have not been set before, you should set ADCLK.CLKDIV, ADCLK.CLKDIV_US and ADCLK.CLKDIV_MS to set A/D clock frequency.
2 Clear ADSTATE.SLP_RDY, then you can set ADENA.SLP_MD to 1. When ADSTATE.SLP_RDY = 1, the Touch Screen is have entered the SLEEP mode.
3 After that you should clear ADSTATE.SLPEND and set ADCTRL.SLPENDM to 0 to enable “in SLEEP mode pen down interrupt” and mask all other interrupts. Then you can execute the SLEEP instruction to enter the SLEEP mode.
4 When “in SLEEP mode pen down interrupt” happened, it will switch from the SLEEP mode to NORMAL. Then you should set ADCTRL.SLPENDM to 1 and clear ADSTATE.SLPEND to close “in SLEEP mode pen down interrupt”. Clear ADSTATE.SLP_RDY, and you should set ADENA.SLP_MD to 0. When ADSTATE.SLP_RDY = 1, the Touch Screen is have exited the SLEEP mode.
5 Then you can do any other operations.

22.4.4 VBAT Sample Operation

1 Set ADCLK.CLKDIV, ADCLK.CLKDIV_US and ADCLK.CLKDIV_MS 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.

22.4.5 AUX Sample Operation

1 Set ADCFG.CMD to choose one CMD. (AUX1 or AUX2)
2 Set ADCLK.CLKDIV, ADCLK.CLKDIV_US and ADCLK.CLKDIV_MS 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.

22.4.6 Disable Touch Screen

1 When ADENA.TCHEN=1, ADENA.VBATEN=0, ADENA.AUXEN=0.
2 Set ADENA.TCHEN to 0.
3 Read ADENA.TCHEN till it is set to 0 by hardware, then Touch Screen is fully disabled.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

22.4.7 Multi-touch Operation

If you want to detect multi-touch, you should follow to steps as below.

1 Set ADENA.YYZ=11.
2 Set ADCTRL.PENDM to 0 to enable the pen down interrupt of touch panel.
3 Set ADCFG.TCHEN=1 to start touch panel.
4 When pen down interrupt happened, you should set ADCTRL.PENDM to 1 and clear ADSTATE.PEND to close pen down interrupt. Then you should clear ADSTATE.PENDU and set ADCTRL.PENUM to 0 to enable pen up interrupt.
5 When ADSTATE.DTCH to 1, you can read the sample data from ADTCH. The measured data recorded as $X2_1$, $Y2_1$, you need to compare the measurement values of $X2_1$, $Y2_1$ and calibration values of X2, Y2. If $X2_1$<X2 and $Y2_1$<Y2, now is two points touch.
6 If the next measure is two points touch, the measured data recorded as $X2_2$, $Y2_2$, you can compare the $X2_1$, $y2_1$ and $X2_2$, $Y2_2$. If $X2_2$ >$X2_1$ and $Y2_2$ > $y2_1$, the touch movement state is shrinkage; if $X2_2$ <$X2_1$ and $Y2_2$ < $Y2_1$, the touch movement state is expand.

NOTE: Before in normally measurement ,you must calibration the values of X2 and Y2. When you calibration the X2, Y2 value, you need a single point of touch the touch panel, record the measurements data. Then repeat these measurements at least three times, recording measure data is $X2_1$, $X2_2$, $X2_3$ and $Y2_1$, $Y2_2$, $Y2_3$. You can use the formula (5) ,(6) to calculate the X2, Y2 value.

$$X2 = \frac{X2_1 + X2_2 + X2_3}{3} \quad (5)$$

$$Y2 = \frac{Y2_1 + Y2_2 + Y2_3}{3} \quad (6)$$

22.4.8 Use Software Command Operation

If you want to use software write command, you should follow to steps as below.

1 Set ADCFG.CMD_SEL =1.
2 Read ADCMD register once, discard the read-in data. This reading purpose is to activate the command logic.
3 Write you command to ADCMD register.
4 Write 0x00000000 to ADCMD register, indicate the written command is end.
5 Set ADENA.TCHEN=1 to start touch panel.

NOTE: The RPU values are 6bits in SARADC, but in ADCMD register , it has 5 bits only. In fact, the lowest RPU bit circuit is given as a fixed value of 1.

22.4.9 Use 5-wire touch panel Operation

If you want to use 5-wrie touch panel, you should complete the following set.

1 Set ADCFG.WIRE_SEL=1 to use 5-wire pendown detect.
2 Write all your commands to the command FIFO through ADCMD register.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

3 Set ADENA.TCHEN=1 to start touch panel.

NOTE: In 5-wire mode, the ADCFG.SNUM will disable. The control logic will execution the control commands until the command value equals to 0x00000000.

22.4.10 Use External Touch Screen Controller Operation

If you want to use external touch screen controller, you should set ADCENA.TCHEN=0 and ADCENA.PEND=0, than you can use external touch screen controller freely.

NOTE: In this mode, all switches will open(default), but you can use VBAT or AUX sample operation by configure the appropriate register.

22.4.11 Use TSC to support keypad

SADC TSC function can apply to a keypad, if touch screen is not used. Suppose the keypad is a NxM matrix, where X direction has N key columns and Y direction has M key rows. Kij is used to indicate the key in ith column from left to right and jth row from bottom to top, where i=0~(N-1) and j=0~(M-1). Figure 22-1 is a 6x5 keypad circuit. The blue color is for X direction network and pink color is for Y. The networks are composed by resistors and metal line. These two networks should be connected to SADC 4 pins: XP/XN/YP/YN as illustrated in the figure. The gray circle is the key. When no key pressing, X network and Y network is open circuit. When a key is pressed, the X network and Y network is shorted under the key position.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

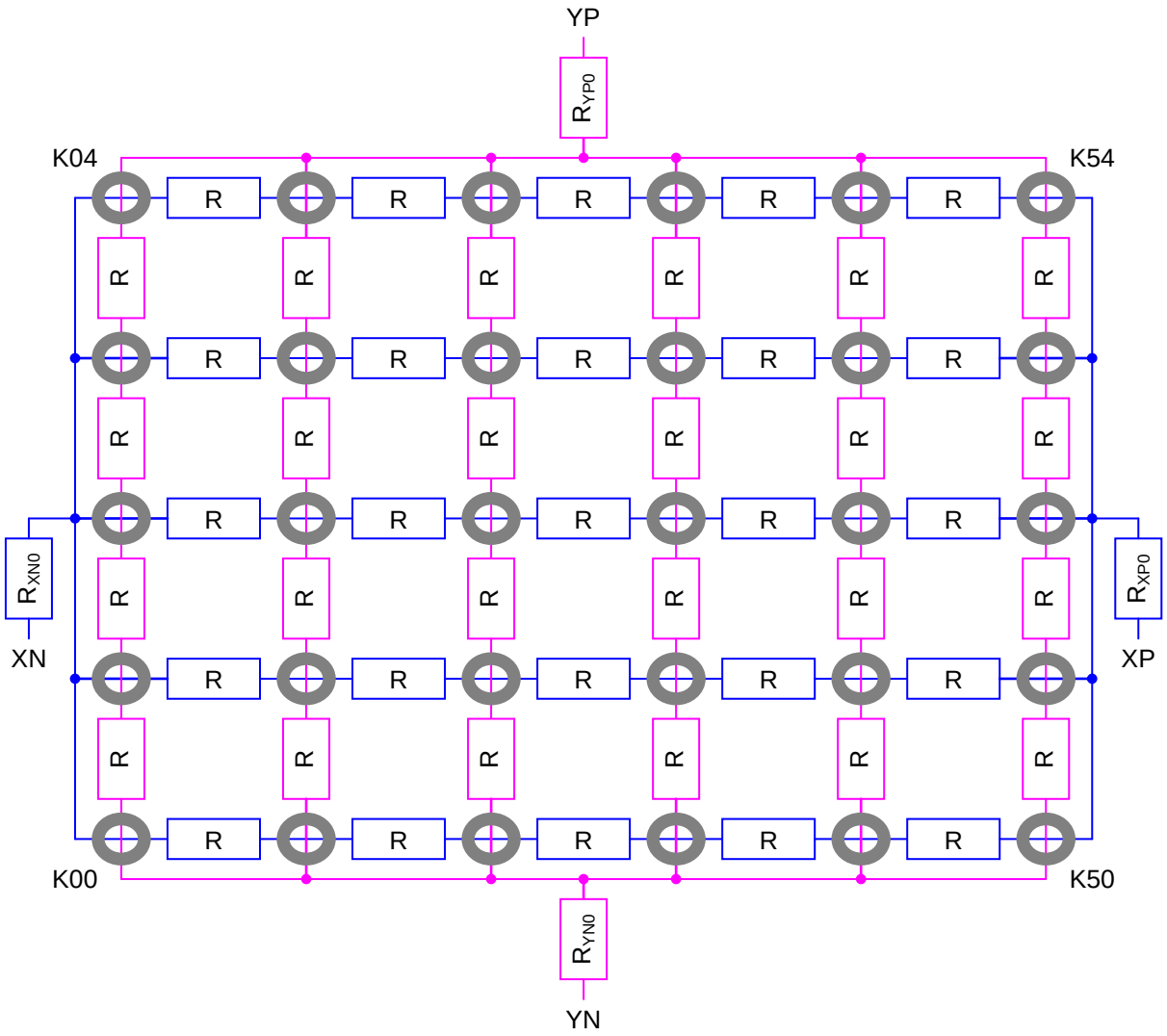


Figure 22-1 6x5 keypad circuit

When SADC is in waiting for pen-down status (C=1100), the equivalent circuit is show in Figure 22-2. When the key is not pressed, XP is open and the PEN is pulled to VDDADC, which is logic 1. When the key Kij is pressed, the circuit is: VDDADC→(10kΩ resistor)→R_{XP}→R_{YN}→VSSADC.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

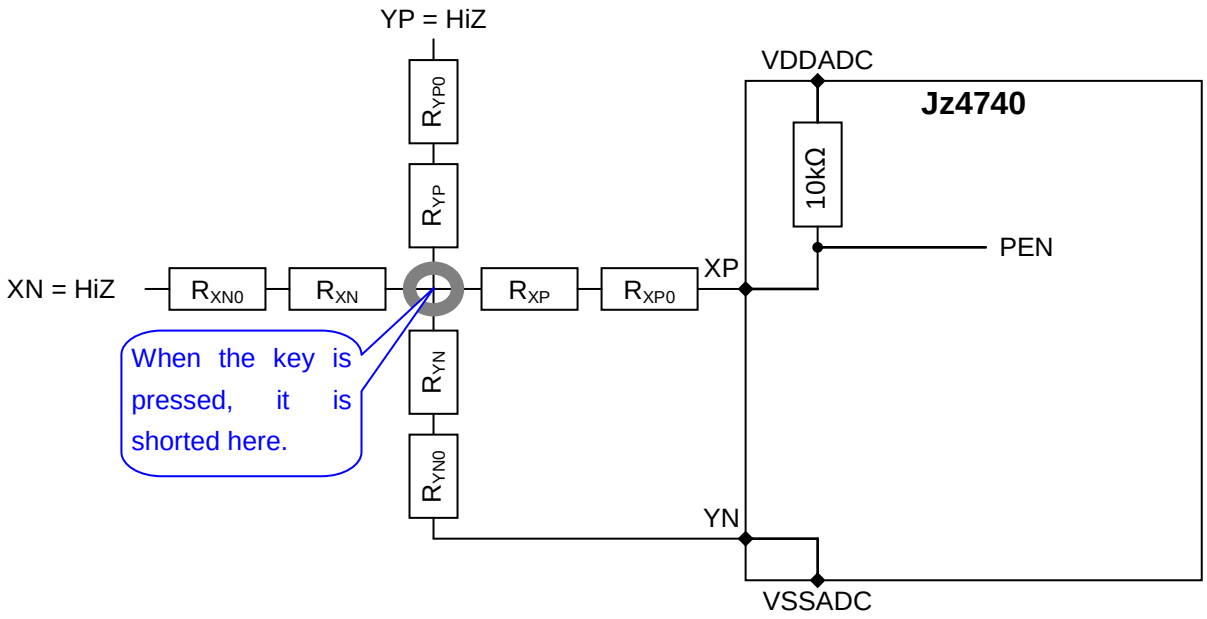


Figure 22-2 Wait for pen-down (C=1100) circuit

Where

$$R_{XP} = \frac{(N-1)^2 - i^2}{M \times (N-1-i) + 2i} \times R$$

$$R_{YN} = \frac{j \times (2M-2-j)}{N \times j + 2M - 2 - 2j} \times R$$

To ensure logic 0 at PEN in this case, following formula should be obeyed.

$$R_{XP} + R_{YN} + R_{XP0} + R_{YN0} \le 3k\Omega \quad (7)$$

It is suggested the value of N and M is as close to each other as possible. For N=2~20, M=2~20 and M=(N-1, N or N+1), we found

$$R_{XP} + R_{YN} < 2.7 \times R \quad (8)$$

After key pressing is found, the key Kij location, columns and row, should be measured by using C=0010 and C=0011 respectively. The equivalent circuits are show in Figure 22-3 and Figure 22-4, where

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

$$R_{X0} = \frac{N-1}{M-1} \times R$$
$$R_{Y0} = \frac{M-1}{N-1} \times R$$
$$R_{XNi} = i \times R$$
$$R_{XPi} = (N-1-i) \times R$$
$$R_{YNj} = j \times R$$
$$R_{YPj} = (M-1-j) \times R$$

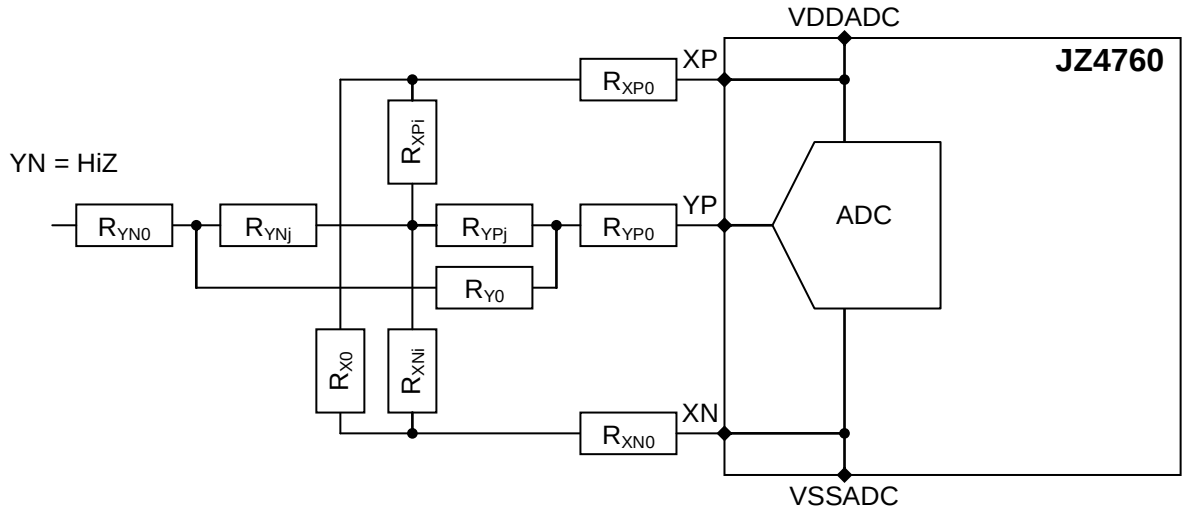


Figure 22-3 Measure X-position (C=0010) circuit

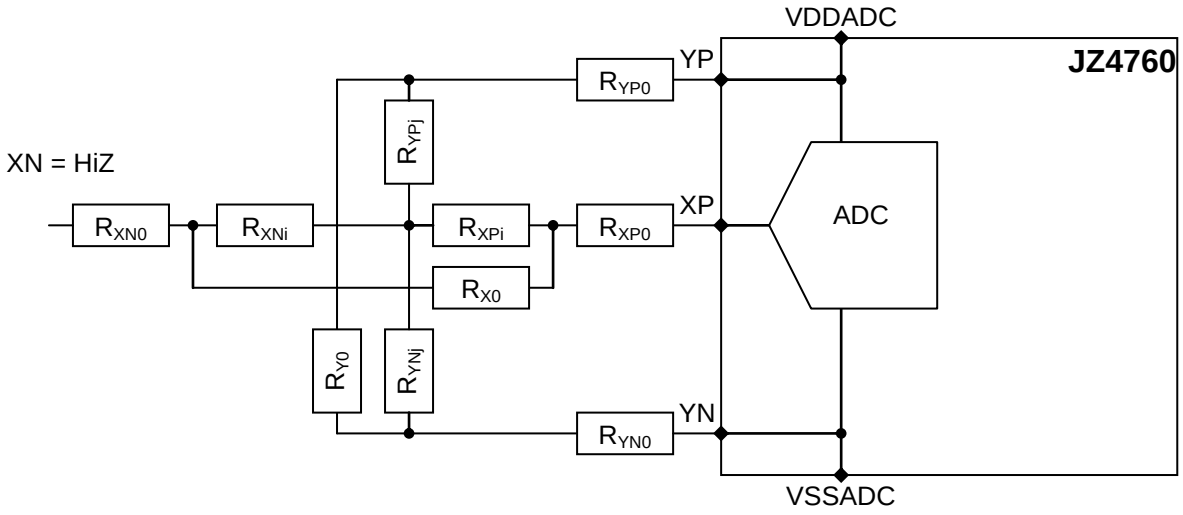


Figure 22-4 Measure Y-position (C=0011) circuit

So for Kij pressing, we should get ADC converted number Ni and Nj for i and j respectively.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

$$Ni = \frac{R_{XN0} + \frac{i}{M}R}{R_{XN0} + \frac{N-1}{M}R + R_{XP0}} \times 4096$$

$$Nj = \frac{R_{YN0} + \frac{j}{N}R}{R_{YN0} + \frac{M-1}{N}R + R_{YP0}} \times 4096$$

It is required the resistor between XP and XN in case of C=0010, between YP and YN in case of C=0011, must be ≥200Ω and it better be ≥500Ω. Also consider the requirement in formula (7) and (8) above, we suggest to put $R_{XP0} = R_{XN0} = R_{YP0} = R_{YN0}$ = 50Ω or 100Ω, put R = 500Ω ~ 1kΩ.

To use the keypad, the software should set:

ADENA.TCHEN = 1

ADCFG.XYZ = 10

The operation is similar to touch screen.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

23 Real Time Clock

23.1 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 32768Hz oscillator, the real time and alarm logic, and the power down and wakeup control logic.

23.1.1 Features

RTC module has following features:

- Embedded 32768Hz oscillator for 32k clock generation with an external 32k crystal
- 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

23.1.2 Signal Descriptions

RTC has 7 signal IO pins and 1 power pin. They are listed and described in.

Pin Names	Pin Loc	IO	IO Cell Char.	Pin Description	Power
RTCLK	AA12	AI	32768Hz	RTCLK: 32768 clock input or OSC input	VDDRTC
XRTCLK	Y12	AO		XRTCLK: OSC output	VDDRTC
PWRON	U13	O	8mA	PWRON: Power on/off control of main power	VDDRTC
WKUP_	U9	I	Schmitt	WKUP_: Wake signal after main power down	VDDRTC
PPRST_	T12	I	Schmitt	PPRST_: RTC power on reset and RESET-KEY reset input	VDDRTC
VDDRTC	W10	P		VDDRTC: power for RTC and hibernating mode controlling that never power down	-

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Pin Names	Pin Loc	IO	IO Cell Char.	Pin Description	Power
CLK32K PD14	AA11	O IO	8mA pullup-pe	32768Hz clock output PD14: GPIO group D bit14. When main power down, this pin is controlled by RTC register	VDDRTC
LDOOUT	Y11	AIO		LDOOUT: capacitor pin for RTC LDO, need a 1nF decoupling capacitor to ground	

RTCLK/XRTCLK pins. We have an embedded oscillator for 32768Hz crystal. These two pins are the crystal XTALI and XTALO connection pins. If an input clock is used instead, please input it to RTCLK pin.

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. (Default 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.

23.2 Register Description

Table 23-1 Registers for real time clock

Name	Description	RW	Reset Value	Address	Access Size
RTCCR	RTC Control Register	RW	0x00000081*1*2	0x10003000	32
RTCSR	RTC Second Register	RW	0x????????	0x10003004	32
RTCSAR	RTC Second Alarm Register	RW	0x????????	0x10003008	32
RTCGR	RTC Regulator Register	RW	0x0???????	0x1000300C	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 23-2 Registers for hibernating mode

Name	Description	RW	Reset Value	Address	Access Size
HCR	Hibernate Control Register	RW	0x00000000*1	0x10003020	32
HWFCR	Wakeup filter counter	RW	0x0000???0	0x10003024	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Register in Hibernate mode				
HRCR	Hibernate reset counter Register in Hibernate mode	RW	0x00000??0	0x10003028	32
HWCR	Wakeup control Register in Hibernate mode	RW	0x00000008*[1]	0x1000302C	32
HWRSR	Wakeup Status Register in Hibernate mode	RW	0x00000000*[1]	0x10003030	32
HSPR	Scratch pattern register	RW	0x????????	0x10003034	32
WENR	Write enable pattern register	RW	0x00000000	0x1000303C	32
CKPCR	Configure the CLK32K pin value	RW	0x00000010	0x10003040	32
PWRONCR	Configure the power on mode	RW	0x???5????	0x10003048	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 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.

23.2.1 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																													0x10003000
Bit	31	30	29	28	27	26	25	24	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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31:7</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>7</td><td>WRDY</td><td>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. Please reference to descriptions in 23.2 for some more details. This bit is read only and write to it is ignored.</td><td>R</td></tr>
<tr><td>6</td><td>1HZ</td><td>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.</td><td>RW</td></tr>
<tr><td>5</td><td>1HZIE</td><td>1Hz interrupt enable. Writing to this bit takes effect immediately without delay.
<table>
<tr><th>1HZIE</th><th>Description</th></tr>
<tr><td>0</td><td>1Hz interrupt is disabled.</td></tr>
<tr><td>1</td><td>1Hz interrupt is enabled. RTC issues interrupt when 1HZ bit is set.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>4</td><td>AF</td><td>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.</td><td>RW</td></tr>
<tr><td>3</td><td>AIE</td><td>Alarm interrupt enable.
<table>
<tr><th>AIE</th><th>Description</th></tr>
<tr><td>0</td><td>Alarm interrupt is disabled.</td></tr>
<tr><td>1</td><td>Alarm interrupt is enabled. RTC issues interrupt when AF is set.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>2</td><td>AE</td><td>Alarm enable.
<table>
<tr><th>AE</th><th>Description</th></tr>
<tr><td>0</td><td>Alarm function is disabled.</td></tr>
<tr><td>1</td><td>Alarm function is enabled.</td></tr>
</table></td><td>RW</td></tr>
<tr><td>1</td><td>SELEXC</td><td>The divided EXCLK is selected as RTCLK in rtc-hiber module.
<table>
<tr><th>SELEXC</th><th>Description</th></tr>
<tr><td>0</td><td>OSC32K or RTCLK input clock is selected as RTCLK in rtc-hiber module.</td></tr>
<tr><td>1</td><td>The divided EXCLK is selected as RTCLK in rtc-hiber module.</td></tr>
</table>
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.</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		When the main chip power down, SELEXC will be 0 in internal circuit, in this time, RTCLK will use OSC32K clock.	
0	RTCE	Real time clock enable. RTCE | Description 0 | Real time clock function is disabled. 1 | Real time clock function is enabled.	RW

23.2.2 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																0x10003004													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

23.2.3 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 enabled (RTCCR.RTCE = 1). RTCSAR can be read and write by software and is not initialized by any reset.

	RTCSAR																0x10003008													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

23.2.4 RTC Regulator Register (RTCGR)

RTCGR is serves as the real time clock regulator, which is used to adjust the interval of the 1Hz pulse.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	RTCGR																														0x1000300C
Bit	31	30	29	28	27	26	25	24	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: 0 – Write to RTCGR is allowed. LOCK: 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 word, 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 word, 1Hz pulse is trigged every (NC1HZ + 1) 32kHz clock cycles, if RTCGR.ADJC = 0.	RW

23.2.5 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																														0x10003020
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 / VDDRTC / No power down, keep power on. 1 / 0 V / Power down enable, turn power off.	RW

23.2.6 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																														0x10003024					
Bit	31	30	29	28	27	26	25	24	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 If this value is configured to 0, and the pin keeps low longer than 15 RTCLK periods, it wakes up RTC from Hibernate.	RW
4:0	Reserved	Writing has no effect, read as zero.	R

23.2.7 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_ .

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	HRCR																															0x10003028
Bit	31	30	29	28	27	26	25	24	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 waiting 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, 62.5ms.	RW
10:0	Reserved	Writing has no effect, read as zero.	R

23.2.8 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																															0x1000302C
Bit	31	30	29	28	27	26	25	24	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																													WKUPVL	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 .

Bits	Name	Description	RW
31:3	EPDET	Power detect enable. 0x5aa5a5a: enable(default) 0x1a55a5a5: disable	RW
2	WKUPVL	RTC wakeup pin valid level. 0: Low level sensitive (default) 1: High sensitive	RW
1	Reserved	Writing has no effect, read as zero.	R
0	EALM	RTC Alarm wakeup enable. 0: disable 1: enable	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.2.9 HIBERNATE Wakeup Status Register (HWRSR)

The HIBERNATE Wakeup Status Register is a 32-bit read/write register that reflects wakeup status bits.

	HWRSR																												0x10003030	
Bit	31	30	29	28	27	26	25	24	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 or another reset occurs. 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 software clears this bit. 1 — Hibernate reset has occurred since the last time the software clears this bit.	RW
4	PPR	PAD PIN Reset. When a PPRST_ is detected, PPR 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. **PPR** / **Description** 0 — PPRST_ reset has not occurred since last time the software clears this bit.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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

23.2.10 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																													0x10003034	
Bit	31	30	29	28	27	26	25	24	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

23.2.11 Write Enable Pattern Register (WENR)

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.

	WENR																													0x1000303C	
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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. **WEN** / **Description** 0: Other RTC registers is locked, write these registers will be ignored. 1: Other RTC registers can be changed.	
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

23.2.12 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																													0x10003040
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	1	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. **CK32PULL** / **Description** 0: Pull Up is enabled. 1: Pull Up is disabled.	RW
3	Reserved	Writing has no effect, read as zero.	R
2:1	CK32CTL	Output RTCLK to CLK32K pin. **CK32CTL** / **Description** 00: CLK32K pin is set to general input.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			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 or gpio (HIBERNATE), Write to this pin will output to CLK32K pin, if configured.		RW

23.2.13 Power on control register (PWRONCR)

This is a register used to control poweron signal. The PWROFF_LEN and PWRON are not initialized by any reset, but OSC_EN and BIAS_CTRL initialized by pprst.

	PWRONCR																0x10003048													
Bit	31	30	29	28	27	26	25	24	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													OSC_EN	BIAS_CTL										PWROFF_LEN				Reserved			PWRON
RST	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	?

Bits	Name	Description	RW
31:8	Reserved	Writing has no effect, read as zero.	R
18	OSC_EN	RTC oscillator enable	RW
17:16	BIAS_CTRL	Bias current control	RW
7:4	PWROFF_LEN	Power off assert length. Number of 4096 RTCLK cycles. Minimum 125ms and Maximum 2 second if the RTCLK is 32768Hz.	RW
3:1	Reserved	Writing has no effect, read as zero.	R
0	PWRON_EN	0: disable 1: enable If this bit set to 1, when PPRST_N assert, the PWRON will output low voltage, after PPRST_N. PWRON change to high voltage.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.3 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

23.4 Operate Mode

23.4.1 Normal Mode

23.4.1.1 Power Detect

When pprst reset, the HWCR.EPDET set to 0x1a55a5a5, and the power detect is disable. If you want To Enable the power detect, you should set HWCR.EPDET to other value, but not the 0x1a55a5a5. If power detect is enable, when CORE 1.2V lose, the RTC will into hibernate mode. And if power detect is not enable, the RTC can't enter hibernate mode when CORE 1.2V is lose.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

23.4.1.2 Power On Timing Diagram

The RTC support two power on procedure. If PWRONCR.PWRON_EN is 0, the procedure like figure 1 -1, and when PWRONCR.PWRON_EN value is 1, the power on procedure will like figure 1 -2.

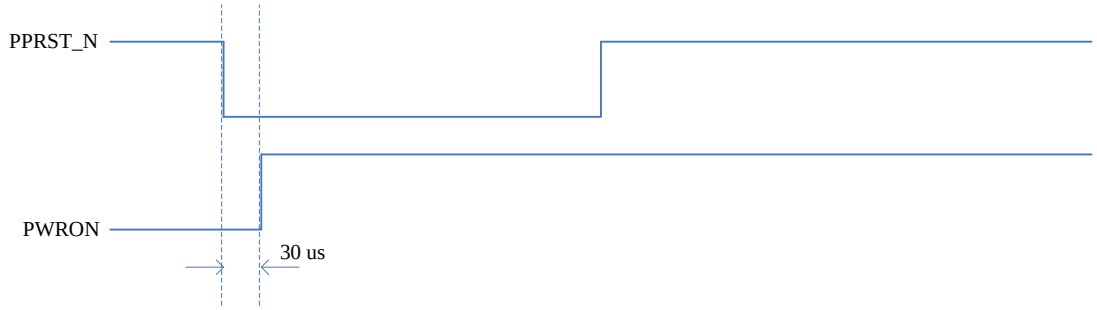


Figure 23-1 Core Power On directly

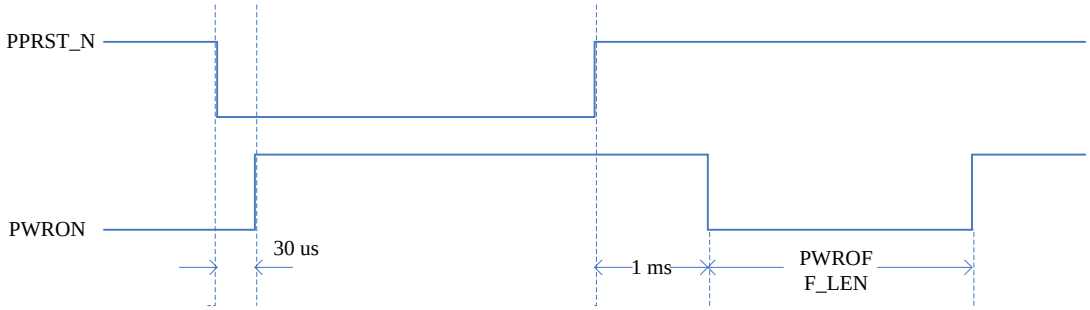


Figure 23-2 Core Power On-Off-On

***Note**: The register PWRONCR.PWROFF_LEN indicate the power off length in figure 1-2.

23.4.2 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 to core and IO. When a wakeup event occurs, the core enters through a hibernate reset. Only CPM wake up logic and RTC is operating in HIBERNATE mode.

23.4.2.1 Procedure to Enter HIBERNATE mode

Before enter Normal HIBERNATE mode, software must complete following steps:

1 Finish the current operation and preserve all data to flash.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

2 Configure the wake-up sources properly by configure HWCR.
3 Set HIBERNATE MODE. (Set PD bit in HCR to 1)

23.4.2.2 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.

23.5 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 23-3 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

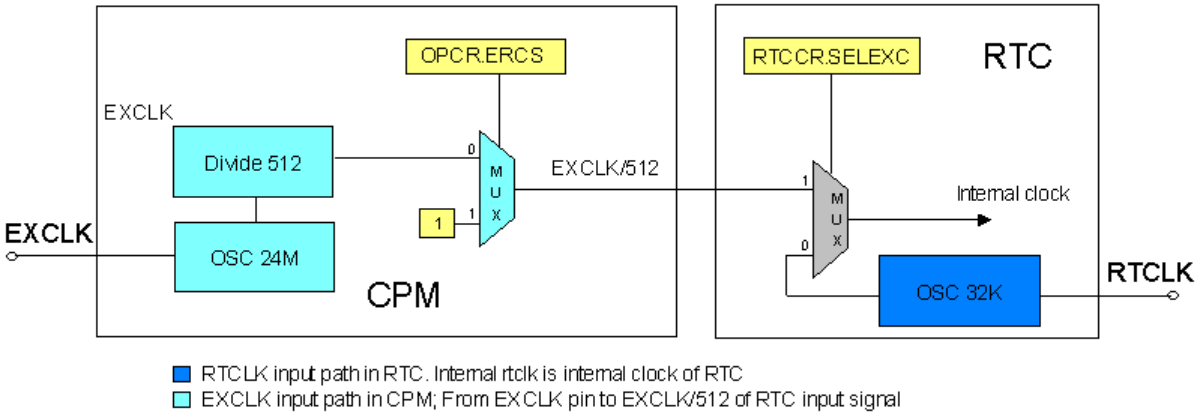


Figure 23-3 RTC clock selection path

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 1; close 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 Clear CLKGR.RTC of CPM to 0; open PCLK to RTC.

g Configure all RTC registers but RTCCR.SELEXC.

h Check RTCCR.SELEXC == 1.

i IF YES, finish this sequence; IF NO, do step (1) 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

24 EFUSE Slave Interface (EFUSE)

24.1 Overview

Total 256 bits of EFUSE are provided, separated into lower 128bits segment and higher 128bits segment.

Each segment can be programmed separately or together.
Each segment has a protect bit, which has higher priority than program segment selection.

Programming frequency should be from 133Mhz to 166Mhz.
Programming time is around 3ms for either program in 128bits or 256bits.

Initial value of EFUSE is 0, when programmed to 1, it won'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.
  Important: VDDQ pin should be kept 0V except during programming. Maximum accumulative time for VDDQ pin exposed under 2.5V+/-10% should be less than 1 sec.
- In read mode, leave VDDQ to 0V.

24.2 Register Description

Following table lists all the register definitions. All registers' 32bit addresses are physical addresses.

Table 24-1 EFUSE Register Description

Name	Description	RW	Reset Value	Address	Access Size
EFSCTL	EFUSE Control Register	RW	0x00	0x134100DC	32
EFUSE0	EFUSE Data 0 Register	RW	0x????????	0x134100E0	32
EFUSE1	EFUSE Data 1 Register	RW	0x????????	0x134100E4	32
EFUSE2	EFUSE Data 2 Register	RW	0x????????	0x134100E8	32
EFUSE3	EFUSE Data 3 Register	RW	0x????????	0x134100EC	32
EFUSE4	EFUSE Data 4 Register	RW	0x????????	0x134100F0	32
EFUSE5	EFUSE Data 5 Register	RW	0x????????	0x134100F4	32
EFUSE6	EFUSE Data 6 Register	RW	0x????????	0x134100F8	32
EFUSE7	EFUSE Data 7 Register	RW	0x????????	0x134100FC	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

24.2.1 EFUSE Control Register (EFSCTL)

	EFSCTL																0x134100DC							
Bit																	7	6	5	4	3	2	1	0
																	Reserved						HI_WEN	LO_WEN
RST																	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:2	Reserved	Writing has no effect, read as zero.	R
1	HI_WEN	High 128 bit write enable bit. *[1] 0: Disable 1: Enabled	RW
0	LO_WEN	Low 128 bit write enable bit. *[1] 0: Disable 1: Enabled	RW

NOTE:

*[1]: DATA[31]/PRTCT bits take higher priorities than HI_WEN & LOW_WEN.

24.2.2 EFUSE Data Register (EFUSEn)

	EFUSEn (n=0,1,2,4,5,6)																0x134100E0~0x134100FC															
Bit	31	30	29	28	27	26	25	24	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	EUFSE DATA register.	RW

EFUSE7 [31]	EFUSE7 [30]	~ EFUSE7 [1]	EFUSE7 [0]
EFUSE6 [31]	EFUSE6 [30]	~ EFUSE6 [1]	EFUSE6 [0]
EFUSE5 [31]	EFUSE5 [30]	~ EFUSE5 [1]	EFUSE5 [0]
EFUSE4 [31]	EFUSE4 [30]	~ EFUSE4 [1]	EFUSE4 [0]
EFUSE3 [31]	EFUSE3 [30]	~ EFUSE3 [1]	EFUSE3 [0]
EFUSE2 [31]	EFUSE2 [30]	~ EFUSE2 [1]	EFUSE2 [0]
EFUSE1 [31]	EFUSE1 [30]	~ EFUSE1 [1]	EFUSE1 [0]
EFUSE0 [31]	EFUSE0 [30]	~ EFUSE0 [1]	EFUSE0 [0]

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	EFUSEn (n=3,7)																0x134100E0~0x134100FC													
Bit	31	30	29	28	27	26	25	24	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[31]/PRTCT	DATA [0-30]																														
RST	0	0	0	0	0	0	0	0	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	DATA[31]/PRTCT	EUFSE DATA bit 31 / Protect bit. *[1] EFSUE3: When this bit programmed to 1, EFUSE0~3 lower 128 bits become un-programmable forever. EFUSE7: When this bit programmed to 1,EFUSE4~7 higher 128 bits become un-programmable forever.	RW
30:0	DATA [30:0]	EUFSE DATA register.	

EFUSE7 [30]	EFUSE7 [29] ~	EFUSE7 [1]	EFUSE7 [0]
EFUSE6 [30]	EFUSE6 [29] ~	EFUSE6 [1]	EFUSE6 [0]
EFUSE5 [30]	EFUSE5 [29] ~	EFUSE5 [1]	EFUSE5 [0]
EFUSE4 [30]	EFUSE4 [29] ~	EFUSE4 [1]	EFUSE4 [0]
EFUSE3 [30]	EFUSE3 [29] ~	EFUSE3 [1]	EFUSE3 [0]
EFUSE2 [30]	EFUSE2 [29] ~	EFUSE2 [1]	EFUSE2 [0]
EFUSE1 [30]	EFUSE1 [29] ~	EFUSE1 [1]	EFUSE1 [0]
EFUSE0 [30]	EFUSE0 [29] ~	EFUSE0 [1]	EFUSE0 [0]

NOTE:

*[1]: DATA[31]/PRTCT bits take higher priorities than HI_WEN & LOW_WEN.

24.3 Flow

24.3.1 Write EFUSE Flow

1 Write full-256 bit.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

a Config AHB2 freq to 166Mhz.
b Connect VDDQ pin to 2.5V. *1
c Write register EFUSE0~EFUSE7.
d Write register EFSCTL.HI_WEN and EFSCTL.LO_WEN to 1 at the same time.
e Wait till register EFSCTL.HI_WEN and EFSCTL.LO_WEN are set to 0.
f Disconnect VDDQ pin from 2.5V and finish writing EFUSE.

2 Write low-128 bit.
a Config AHB2 freq to 166Mhz.
b Connect VDDQ pin to 2.5V. *1
c Write register EFUSE0~EFUSE3.
d Write register EFSCTL.LO_WEN to 1.
e Wait till register EFSCTL.LO_WEN is set to 0.
f Disconnect VDDQ pin from 2.5V and finish writing EFUSE.

3 Write high-128 bit.
a Config AHB2 freq to 166Mhz.
b Connect VDDQ pin to 2.5V. *1
c Write register EFUSE3~EFUSE7.
d Write register EFSCTL.HI_WEN to 1.
e Wait till register EFSCTL.HI_WEN is set to 0.
f Disconnect VDDQ pin from 2.5V and finish writing EFUSE.

NOTE:

*1: Do NOT pre-charge VDDQ too early before fuse program started.
The maximum 2.5V supply time to VDDQ must be strictly controlled less than 1sec.
Programming window is below 3ms.
Use on-chip counter and GPIO to coordinate external supply source.

24.3.2 Read EFUSE Flow

1. Read register EFUSE0~EFUSE7 after chip is reset.
2. EFUSE2 high 16bits stored SARADC calibration data.
3. The data from EFUSE0 to EFUSE3 is chipid, except EFUSE2 high 16bits.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 6

PERIPHERALS

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

25 General-Purpose I/O Ports

25.1 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

25.1.1 GPIO Port A Summary

Table 25-1 GPIO Port A summary

Bit N	PA N	Pull (U/D)	Shared Function Port Selected by				
			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)	-	-	
16	16	U rst-pe	rd_o(o)	-	-	-	
17	17	U rst-pe	we_o(o)	-	-	-	
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 rst-pe	(i)	msc0_d0(io)	ssi0_dr_i(i)	-	1

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

21	21	U rst-pe	cs1_o(o)	msc0_d1(io)	-	-	
22	22	U rst-pe	cs2_o(o)	msc0_d2(io)	-	-	
23	23	U rst-pe	cs3_o(o)	msc0_d3(io)	ssi0_ce0(io)	-	
27	27	U	wait_i(i)	-	-	-	
29	29	U	dqsn(io)	-	-	-	
30	30	-	-	-	-	-	6
31	31	-	-	-	-	-	7,11

25.1.2 GPIO Port B Summary

Table 25-2 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)	-	-	-	
3	03	U	sa3_o(o)	-	-	-	
4	04	U	sa4_o(o)	gmac_crs_i(i)		-	
5	05	U	sa5_o(o)	-	-	-	
6	06	U	cim0_pclk_i(i)	-	-	-	
7	07	U	cim0_hsyn_i(i)	-	-	-	
8	08	U rst-pe	cim0_vsyn_i(i)	gmac_txer_o(o)	-	-	
9	09	U	cim0_mclk_o(o)	-	-	epd_pwc_o(o)	
10	10	D	cim0_d0_i(i)	gmac_txd4_o(o)	-	epd_pwr0_o(o)	
11	11	D	cim0_d1_i(i)	gmac_txd5_o(o)	-	epd_pwr1_o(o)	
12	12	U	cim0_d2_i(i)	gmac_txd6_o(o)	-	epd_sce2_o(o)	
13	13	U	cim0_d3_i(i)	gmac_txd7_o(o)	-	epd_sce3_o(o)	
14	14	U	cim0_d4_i(i)	gmac_rxd4_i(i)	-	epd_sce4_o(o)	
15	15	U	cim0_d5_i(i)	gmac_rxd5_i(i)	-	epd_sce5_o(o)	
16	16	D	cim0_d6_i(i)	gmac_rxd6_i(i)	-	epd_pwr2_o(o)	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

17	17	D	cim0_d7_i(i)	gmac_rxd7_i(i)	-	epd_pwr3_o(o)	
20	20	U	msc2_d0(io)	-	-	-	
21	21	U	msc2_d1(io)	-	-	-	
28	28	U	msc2_clk_o(o)	-	-	-	
29	29	U rst-pe	msc2_cmd(io)	-	-	-	
30	30	U	msc2_d2(io)	-	-	-	
31	31	U rst-pe	msc2_d3(io)	-	-	-	

25.1.3 GPIO Port C Summary

Table 25-3 GPIO Port C summary

Bit	PC	Pull	Shared Function Port Selected by				
N	N	(U/D)	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)	-	-	-	
3	03	U	lcd_b3_o(o)	-	-	-	
4	04	U	lcd_b4_o(o)	-	-	-	
5	05	U	lcd_b5_o(o)	-	-	-	
6	06	U	lcd_b6_o(o)	-	-	-	
7	07	U	lcd_b7_o(o)	-	-	-	
8	08	U	lcd_pclk_o(o)	-	-	-	
9	09	U	lcd_de_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)	-	-	-	
13	13	U	lcd_g3_o(o)	-	-	-	
14	14	U	lcd_g4_o(o)	-	-	-	
15	15	U	lcd_g5_o(o)	-	-	-	
16	16	U	lcd_g6_o(o)	-	-	-	
17	17	U	lcd_g7_o(o)	-	-	-	
18	18	U	lcd_hsyn(io)	-	-	-	
19	19	U	lcd_vsyn(io)	-	-	-	
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)	-	-	-	
23	23	U	lcd_r3_o(o)	-	-	-	
24	24	U	lcd_r4_o(o)	-	-	-	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

25	25	U	lcd_r5_o(o)	-	-	-	
26	26	U	lcd_r6_o(o)	-	-	-	
27	27	U	lcd_r7_o(o)	-	-	-	

25.1.4 GPIO Port D Summary

Table 25-4 GPIO Port D summary

Bit N	PD N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
14	14	U	-	-	-	-	10
15	15	D rst-pe	exclk_i(o)	-	-	-	
17	17	U	-	-	-	-	2,5
18	18	U	-	-	-	-	3,5
19	19	U	-	-	-	-	4,5
20	20	U	msc1_d0(io)	ssi0_dr_i(i)	-	-	
21	21	U	msc1_d1(io)	ssi0_dt(io)	-	-	
22	22	U	msc1_d2(io)	ssi0_gpc(io)	-	-	
23	23	U rst-pe	msc1_d3(io)	ssi0_ce1(io)	-	-	
24	24	U	msc1_clk_o(o)	ssi0_clk(io)	-	-	
25	25	U rst-pe	msc1_cmd(io)	ssi0_ce0(io)	-	-	
26	26	U	uart1_rxd_i(i)	gmac_rxd2_i(i)	-	-	
27	27	U	uart1_cts_i(i)	gmac_rxd3_i(i)	-	-	
28	28	U rst-pe	uart1_txd_o(o)	gmac_txd2_o(o)	-	-	
29	29	U rst-pe	uart1_rts_o(o)	gmac_txd3_o(o)	-	-	
30	30	U	i2c0_sda(io)	-	-	-	
31	31	U	i2c0_sck(io)	-	-	-	

25.1.5 GPIO Port E Summary

Table 25-5 GPIO Port E summary

Bit N	PE N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
0	00	D	pwm0(io)	i2c2_sda(io)	-	-	
1	01	D	pwm1_o(o)	-	-	-	
2	02	U	pwm2_o(o)	-	-	-	
3	03	U	pwm3(io)	i2c2_sck(io)	sysclk_o(o)	-	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		rst_pe					
10	10	D rst-pe	drv_vbus_o(o)	-	-	-	
20	20	U	msc0_d0(io)	msc1_d0(io)	msc2_d0(io)	-	
21	21	U	msc0_d1(io)	msc1_d1(io)	msc2_d1(io)	-	
22	22	U	msc0_d2(io)	msc1_d2(io)	msc2_d2(io)	-	
23	23	U	msc0_d3(io)	msc1_d3(io)	msc2_d3(io)	-	
28	28	U	msc0_clk_o(o-1)	msc1_clk_o(o)	msc2_clk_o(o)	-	
29	29	U	msc0_cmd(io)	msc1_cmd(io)	msc2_cmd(io)	-	
30	30	U	i2c1_sda(io)	-	-	-	
31	31	U	i2c1_sck(io)	-	-	-	

25.1.6 GPIO Port F Summary

Table 25-6 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)	bclk(io)	-	-	
1	01	U rst-pe	uart0_cts_i(i)	lrclk(io)	-	-	
2	02	U rst-pe	uart0_rts_o(o)	sdati_i(i)	-	-	
3	03	U rst-pe	uart0_txd_o(o)	sdato_o(o)	-	-	
4	04	D	gmac_txd0_o(o)	uart2_txd_o(o)	-	-	
5	05	D	gmac_txd1_o(o)	uart2_rxd_i(i)	-	-	
6	06	D	gmac_txclk_i(i)	-	-	-	
7	07	D	gmac_rxclk_i(i)	-	-	-	
8	08	D	gmac_rxer_i(i)	-	-	epd_pwr4_o(o)	
9	09	D	gmac_rxdv_i(i)	-	-	epd_pwr5_o(o)	
10	10	D	gmac_rxd0_i(i)	dmic_clk_o(o)	-	epd_pwr6_o(o)	
11	11	D	gmac_rxd1_i(i)	dmic_in_i(i)	-	epd_pwr7_o(o)	
12	12	U	gmac_txen_o(o)	pcm0_do_o(o)	-	-	
13	13	U	gmac_mdc_o(o)	pcm0_clk(io)	-	-	
14	14	U	gmac_mdio(io)	pcm0_syn(io)	-	-	
15	15	U	gmac_col_i(i)	pcm0_di_i(i)	-	-	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

25.1.7 GPIO Port G Summary

Table 25-7 GPIO Port G summary

Bit N	PF N	Pull (U/D)	Shared Function Port Selected by				
			0	1	2	3	Note
6	06	U	cim1_pclk_i(i)	-	-	-	
7	07	U	cim1_hsyn_i(i)	-	-	-	
8	08	U rst-pe	cim1_vsyn_i(i)	-	-	-	
9	09	U	cim1_mclk_o(o)	-	-	-	
10	10	D	cim1_d0_i(i)	sd8(io)	-	-	
11	11	D	cim1_d1_i(i)	sd9(io)	-	-	
12	12	U	cim1_d2_i(i)	sd10(io)	-	-	
13	13	U	cim1_d3_i(i)	sd11(io)	-	-	
14	14	U	cim1_d4_i(i)	sd12(io)	-	-	
15	15	U	cim1_d5_i(i)	sd13(io)	-	-	
16	16	D	cim1_d6_i(i)	sd14(io)	-	-	
17	17	D	cim1_d7_i(i)	sd15(io)	-	-	

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 input during boot.

3 PD18: GPIO group D bit 18 is used as BOOT_SEL1 input during boot.

4 PD19: GPIO group D bit 19 is used as BOOT_SEL2 input during boot.

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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/PS2 Pins.

25.2 Registers Description

Table 25-8 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 25-8 GPIO Registers

Name	Description	RW	Reset Value	Address	Size
GPIO PORT A					
PAPIN	PORT A PIN Level Register	R	0x????????	0x10010000	32
PAINT	PORT A Interrupt Register	RW	0x00000000	0x10010010	32
PAINTS	PORT A Interrupt Set Register	W	0x????????	0x10010014	32
PAINTC	PORT A Interrupt Clear Register	W	0x????????	0x10010018	32
PAMSK	PORT A Interrupt Mask Register	RW	0xFFFFFFFF	0x10010020	32
PAMSKS	PORT A Interrupt Mask Set Register	W	0x????????	0x10010024	32
PAMSKC	PORT A Interrupt Mask Clear Register	W	0x????????	0x10010028	32
PAPAT1	PORT A Pattern 1 Register	RW	0xFFFFFFFF	0x10010030	32
PAPAT1S	PORT A Pattern 1 Set Register	W	0x????????	0x10010034	32
PAPAT1C	PORT A Pattern 1 Clear Register	W	0x????????	0x10010038	32
PAPAT0	PORT A Pattern 0 Register	RW	0x00000000	0x10010040	32
PAPAT0S	PORT A Pattern 0 Set Register	W	0x????????	0x10010044	32
PAPAT0C	PORT A Pattern 0 Clear Register	W	0x????????	0x10010048	32
PAFLG	PORT A FLAG Register	R	0x00000000	0x10010050	32
PAFLGC	PORT A FLAG Clear Register	W	0x????????	0x10010058	32
PAPEN	PORT A PULL Disable Register	RW	0x00000000	0x10010070	32
PAPENS	PORT A PULL Disable Set Register	W	0x????????	0x10010074	32
PAPENC	PORT A PULL Disable Clear Register	W	0x????????	0x10010078	32
GPIO PORT B					
PBPIN	PORT B PIN Level Register	R	0x????????	0x10010100	32
PBINT	PORT B Interrupt Register	RW	0x00000000	0x10010110	32
PBINTS	PORT B Interrupt Set Register	W	0x????????	0x10010114	32
PBINTC	PORT B Interrupt Clear Register	W	0x????????	0x10010118	32
PBMSK	PORT B Interrupt Mask Register	R	0xFFFFFFFF	0x10010120	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

PBMSKS	PORT B Interrupt Mask Set Register	W	0x????????	0x10010124	32
PBMSKC	PORT B Interrupt Mask Clear Register	W	0x????????	0x10010128	32
PBPAT1	PORT B Pattern 1 Register	R	0xFFFFFFFF	0x10010130	32
PBPAT1S	PORT B Pattern 1 Set Register	W	0x????????	0x10010134	32
PBPAT1C	PORT B Pattern 1 Clear Register	W	0x????????	0x10010138	32
PBPAT0	PORT B Pattern 0 Register	RW	0x00000000	0x10010140	32
PBPAT0S	PORT B Pattern 0 Set Register	W	0x????????	0x10010144	32
PBPAT0C	PORT B Pattern 0 Clear Register	W	0x????????	0x10010148	32
PBFLG	PORT B FLAG Register	R	0x00000000	0x10010150	32
PBFLGC	PORT B FLAG Clear Register	W	0x????????	0x10010158	32
PBPEN	PORT B PULL Disable Register	RW	0x00000000	0x10010170	32
PBPENS	PORT B PULL Disable Set Register	W	0x????????	0x10010174	32
PBPENC	PORT B PULL Disable Clear Register	W	0x????????	0x10010178	32
GPIO PORT C					
PCPIN	PORT C PIN Level Register	R	0x????????	0x10010200	32
PCINT	PORT C Interrupt Register	RW	0x00000000	0x10010210	32
PCINTS	PORT C Interrupt Set Register	W	0x????????	0x10010214	32
PCINTC	PORT C Interrupt Clear Register	W	0x????????	0x10010218	32
PCMSK	PORT C Interrupt Mask Register	R	0xFFFFFFFF	0x10010220	32
PCMSKS	PORT C Interrupt Mask Set Register	W	0x????????	0x10010224	32
PCMSKC	PORT C Interrupt Mask Clear Register	W	0x????????	0x10010228	32
PCPAT1	PORT C Pattern 1 Register	R	0xFFFFFFFF	0x10010230	32
PCPAT1S	PORT C Pattern 1 Set Register	W	0x????????	0x10010234	32
PCPAT1C	PORT C Pattern 1 Clear Register	W	0x????????	0x10010238	32
PCPAT0	PORT C Pattern 0 Register	RW	0x00000000	0x10010240	32
PCPAT0S	PORT C Pattern 0 Set Register	W	0x????????	0x10010244	32
PCPAT0C	PORT C Pattern 0 Clear Register	W	0x????????	0x10010248	32
PCFLG	PORT C FLAG Register	R	0x00000000	0x10010250	32
PCFLGC	PORT C FLAG Clear Register	W	0x????????	0x10010258	32
PCPEN	PORT C PULL Disable Register	RW	0x00000000	0x10010270	32
PCPENS	PORT C PULL Disable Set Register	W	0x????????	0x10010274	32
PCPENC	PORT C PULL Disable Clear Register	W	0x????????	0x10010278	32
GPIO PORT D					
PDPIN	PORT D PIN Level Register	R	0x????????	0x10010300	32
PDINT	PORT D Interrupt Register	RW	0x00000000	0x10010310	32
PDINTS	PORT D Interrupt Set Register	W	0x????????	0x10010314	32
PDINTC	PORT D Interrupt Clear Register	W	0x????????	0x10010318	32
PDMSK	PORT D Interrupt Mask Register	R	0xFFFFFFFF	0x10010320	32
PDMSKS	PORT D Interrupt Mask Set Register	W	0x????????	0x10010324	32
PDMSKC	PORT D Interrupt Mask Clear Register	W	0x????????	0x10010328	32
PDPAT1	PORT D Pattern 1 Register	R	0xFFFFFFFF	0x10010330	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

PDPAT1S	PORT D Pattern 1 Set Register	W	0x????????	0x10010334	32
PDPAT1C	PORT D Pattern 1 Clear Register	W	0x????????	0x10010338	32
PDPAT0	PORT D Pattern 0 Register	RW	0x00000000	0x10010340	32
PDPAT0S	PORT D Pattern 0 Set Register	W	0x????????	0x10010344	32
PDPAT0C	PORT D Pattern 0 Clear Register	W	0x????????	0x10010348	32
PDFLG	PORT D FLAG Register	R	0x00000000	0x10010350	32
PDFLGC	PORT D FLAG Clear Register	W	0x????????	0x10010358	32
PDPEN	PORT D PULL Disable Register	RW	0x00000000	0x10010370	32
PDPENS	PORT D PULL Disable Set Register	W	0x????????	0x10010374	32
PDPENC	PORT D PULL Disable Clear Register	W	0x????????	0x10010378	32
GPIO PORT E					
PEPIN	PORT E PIN Level Register	R	0x????????	0x10010400	32
PEINT	PORT E Interrupt Register	RW	0x00000000	0x10010410	32
PEINTS	PORT E Interrupt Set Register	W	0x????????	0x10010414	32
PEINTC	PORT E Interrupt Clear Register	W	0x????????	0x10010418	32
PEMSK	PORT E Interrupt Mask Register	R	0xFFFFFFFF	0x10010420	32
PEMSKS	PORT E Interrupt Mask Set Register	W	0x????????	0x10010424	32
PEMSKC	PORT E Interrupt Mask Clear Register	W	0x????????	0x10010428	32
PEPAT1	PORT E Pattern 1 Register	R	0xFFFFFFFF	0x10010430	32
PEPAT1S	PORT E Pattern 1 Set Register	W	0x????????	0x10010434	32
PEPAT1C	PORT E Pattern 1 Clear Register	W	0x????????	0x10010438	32
PEPAT0	PORT E Pattern 0 Register	RW	0x00000000	0x10010440	32
PEPAT0S	PORT E Pattern 0 Set Register	W	0x????????	0x10010444	32
PEPAT0C	PORT E Pattern 0 Clear Register	W	0x????????	0x10010448	32
PEFLG	PORT E FLAG Register	R	0x00000000	0x10010450	32
PEFLGC	PORT E FLAG Clear Register	W	0x????????	0x10010458	32
PEPEN	PORT E PULL Disable Register	RW	0x00000000	0x10010470	32
PEPENS	PORT E PULL Disable Set Register	W	0x????????	0x10010474	32
PEPENC	PORT E PULL Disable Clear Register	W	0x????????	0x10010478	32
GPIO PORT F					
PFPIN	PORT F PIN Level Register	R	0x????????	0x10010500	32
PFINT	PORT F Interrupt Register	RW	0x00000000	0x10010510	32
PFINTS	PORT F Interrupt Set Register	W	0x????????	0x10010514	32
PFINTC	PORT F Interrupt Clear Register	W	0x????????	0x10010518	32
PFMSK	PORT F Interrupt Mask Register	R	0xFFFFFFFF	0x10010520	32
PFMSKS	PORT F Interrupt Mask Set Register	W	0x????????	0x10010524	32
PFMSKC	PORT F Interrupt Mask Clear Register	W	0x????????	0x10010528	32
PFPAT1	PORT F Pattern 1 Register	R	0xFFFFFFFF	0x10010530	32
PFPAT1S	PORT F Pattern 1 Set Register	W	0x????????	0x10010534	32
PFPAT1C	PORT F Pattern 1 Clear Register	W	0x????????	0x10010538	32
PFPAT0	PORT F Pattern 0 Register	RW	0x00000000	0x10010540	32

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

PFPAT0S	PORT F Pattern 0 Set Register	W	0x????????	0x10010544	32
PFPAT0C	PORT F Pattern 0 Clear Register	W	0x????????	0x10010548	32
PFFLG	PORT F FLAG Register	R	0x00000000	0x10010550	32
PFFLGC	PORT F FLAG Clear Register	W	0x????????	0x10010558	32
PFPEN	PORT F PULL Disable Register	RW	0x00000000	0x10010570	32
PFPENS	PORT F PULL Disable Set Register	W	0x????????	0x10010574	32
PFPENC	PORT F PULL Disable Clear Register	W	0x????????	0x10010578	32
GPIO PORT G					
PFPIN	PORT G PIN Level Register	R	0x????????	0x10010600	32
PFINT	PORT G Interrupt Register	RW	0x00000000	0x10010610	32
PFINTS	PORT G Interrupt Set Register	W	0x????????	0x10010614	32
PFINTC	PORT G Interrupt Clear Register	W	0x????????	0x10010618	32
PFMSK	PORT G Interrupt Mask Register	R	0xFFFFFFFF	0x10010620	32
PFMSKS	PORT G Interrupt Mask Set Register	W	0x????????	0x10010624	32
PFMSKC	PORT G Interrupt Mask Clear Register	W	0x????????	0x10010628	32
PFPAT1	PORT G Pattern 1 Register	R	0xFFFFFFFF	0x10010630	32
PFPAT1S	PORT G Pattern 1 Set Register	W	0x????????	0x10010634	32
PFPAT1C	PORT G Pattern 1 Clear Register	W	0x????????	0x10010638	32
PFPAT0	PORT G Pattern 0 Register	RW	0x00000000	0x10010640	32
PFPAT0S	PORT G Pattern 0 Set Register	W	0x????????	0x10010644	32
PFPAT0C	PORT G Pattern 0 Clear Register	W	0x????????	0x10010648	32
PFFLG	PORT G FLAG Register	R	0x00000000	0x10010650	32
PFFLGC	PORT G FLAG Clear Register	W	0x????????	0x10010658	32
PFPEN	PORT G PULL Disable Register	RW	0x00000000	0x10010670	32
PFPENS	PORT G PULL Disable Set Register	W	0x????????	0x10010674	32
PFPENC	PORT G PULL Disable Clear Register	W	0x????????	0x10010678	32

NOTE: PX**** in the description of register as follows means PA****, PB****, PC****, PD****, PE**** and PF****.

25.2.1 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.

	PAPIN, PBPIN, PCPIN, PDPIN, PEPIN, PFPIN																**0x10010000, 0x10010100, 0x10010200, 0x10010300, 0x10010400, 0x10010500, 0x10010600,**															
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

25.2.2 PORT Interrupt Registers (PxINT)

PAINT, PBINT, PCINT, PDINT, PEINT and PFINT are six 32-bit interrupt registers.

	PAINT, PBINT, PCINT, PDINT, PEINT, PFINT																0x10010010, 0x10010110, 0x10010210, 0x10010310, 0x10010410, 0x10010510, 0x10010610,														
Bit	31	30	29	28	27	26	25	24	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

25.2.3 PORT Interrupt Set Registers (PxINTS)

PAINTS, PBINTS, PCINTS, PDINTS, PEINTS and PFINTS are six 32-bit interrupt set registers.

	PAINTS, PBINTS, PCINTS, PDINTS, PEINTS, PFINTS																0x10010014, 0x10010114, 0x10010214, 0x10010314, 0x10010414, 0x10010514, 0x10010614,														
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

25.2.4 PORT Interrupt Clear Registers (PxINTC)

PAINTC, PBINTC, PCINTC, PDINTC, PEINTC and PFINTC are six 32-bit interrupt clear registers.

	PAINTC, PBINTC, PCINTC, PDINTC, PEINTC, PFINTC																0x10010018, 0x10010118, 0x10010218, 0x10010318, 0x10010418, 0x10010518, 0x10010618,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.5 PORT Mask Registers (PxMSK)

PAMSK, PBMSK, PCMSK, PDMSK, PEMSK and PFMSK are six 32-bit PORT MASK registers.

	PAMSK, PBMSK, PCMSK, PDMSK. PEMSK, PFMSK																0x10010020, 0x10010120, 0x10010220, 0x10010320, 0x10010420, 0x10010520, 0x10010620,													
Bit	31	30	29	28	27	26	25	24	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	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	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

25.2.6 PORT Mask Set Registers (PxMSKS)

PAMSKS, PBMSKS, PCMSKS, PDMSKS, PEMSKS and PFMSKS are six 32-bit PORT MASK set registers.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PAMSKS, PBMSKS, PCMSKS, PDMSKS, PEMSKS, PFMSKS																0x10010024, 0x10010124, 0x10010224, 0x10010324, 0x10010424, 0x10010524, 0x10010624,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.7 PORT Mask Clear Registers (PxMSKC)

PAMSKC, PBMSKC, PCMSKC, PDMSKC, PEMSKC and PFMSKC are six 32-bit PORT MASK clear registers.

	PAMSKS, PBMSKC, PCMSKC, PDMSKC, PEMSKC, PFMSKC																0x10010028, 0x10010128, 0x10010228, 0x10010328, 0x10010428, 0x10010528, 0x10010628,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.8 PORT PAT1/Direction Registers (PxPAT1)

PAPAT1, PBPAT1, PCPAT1, PDPAT1, PEPAT1 and PFPAT1 are six 32-bit PORT pattern1/direction registers.

	PAPAT1, PBPAT1, PCPAT1, PDPAT1, PEPAT1, PFPAT1																0x10010030, 0x10010130, 0x10010230, 0x10010330, 0x10010430, 0x10010530, 0x10010630,													
Bit	31	30	29	28	27	26	25	24	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	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

25.2.9 PORT PAT1/Direction Set Registers (PxPAT1S)

PAPAT1S, PBPAT1S, PCPAT1S, PDPAT1S, PEPAT1S and PFPAT1S are six 32-bit PORT pattern1/direction set registers.

	PAPAT1S, PBPAT1S, PCPAT1S, PDPAT1S, PEPAT1S, PFPAT1S																**0x10010034, 0x10010134, 0x10010234, 0x10010334, 0x10010434, 0x10010534, 0x10010634**															
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.10 PORT PAT1/Direction Clear Registers (PxPAT1C)

PAPAT1C, PBPAT1C, PCPAT1C, PPAT1C, PEPAT1C and PFPAT1C are six 32-bit PORT pattern1 or direction clear registers.

	PAPAT1S, PBPAT1C, PCPAT1C, PDPAT1C, PEPAT1C, PFPAT1C																**0x10010038, 0x10010138, 0x10010238, 0x10010338, 0x10010438, 0x10010538, 0x10010638,**															
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.	W

25.2.11 PORT PAT0/Data Registers (PxPAT0)

PAPAT0, PBPAT0, PCPAT0, PDPAT0, PEPAT0 and PFPAT0 are six 32-bit PORT Pattern 0 or DATA registers.

	PAPAT0, PBPAT0, PCPAT0, PDPAT0, PEPAT0, PFPAT0																0x10010040, 0x10010140, 0x10010240, 0x10010340, 0x10010440, 0x10010540, 0x10010640,													
Bit	31	30	29	28	27	26	25	24	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. 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	RW

25.2.12 PORT PAT0/Data Set Registers (PxPAT0S)

PAPAT0S, PBPAT0S, PCPAT0S, PDPAT0S, PEPAT0S and PFPAT0S are six 32-bit PORT Pattern0 or DATA set registers.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	PAPAT0S, PBPAT0S, PCPAT0S, PDPAT0S, PEPAT0S, PFPAT0S																0x10010044, 0x10010144, 0x10010244, 0x10010344, 0x10010444, 0x10010544, 0x10010644,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.13 PORT PAT0/Data Clear Registers (PxPAT0C)

PAPAT0C, PBPAT0C, PCPAT0C, PDPAT0C, PEPAT0C and PFPAT0C are six 32-bit PORT Pattern 0 or DATA clear registers.

	PAPAT0C, PBPAT0C, PCPAT0C, PDPAT0C, PEPAT0C, PFPAT0C																0x10010048, 0x10010148, 0x10010248, 0x10010348, 0x10010448, 0x10010548, 0x10010648,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.14 PORT FLAG Registers (PxFLG)

PAFLG, PBFLG, PCFLG, PDFLG, PEFLG and PFFLG are six 32-bit GPIO FLAG registers. They are read-only registers.

	PAFLG, PBFLG, PCFLG, PDFLG, PEFLG, PFFLG																0x10010050, 0x10010150, 0x10010250, 0x10010350, 0x10010450, 0x10010550, 0x10010650,													
Bit	31	30	29	28	27	26	25	24	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

25.2.15 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.

	PAFLGC, PBFLGC, PCFLGC, PDFLGC, PEFLGC, PFFLGC																0x10010058, 0x10010158, 0x10010258, 0x10010358, 0x10010458,0x10010558, 0x10010658,													
Bit	31	30	29	28	27	26	25	24	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.	R

25.2.16 PORT PULL Disable Registers (PxPE)

PAPE, PBPE, PCPE, PDPE, PEPE and PFPE are six 32-bit PORT PULL disable registers.

	PAPE, PBPE, PCPE, PDPE, PEPE, PFPE																0x10010070, 0x10010170, 0x10010270, 0x10010370, 0x10010470, 0x10010570, 0x10010670,													
Bit	31	30	29	28	27	26	25	24	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	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		down is pin dependence 1: No pull up or pull down resistor connects to the port	

25.2.17 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.

	PAPES, PBPES, PCPES, PDPES, PEPES, PFPES																0x10010074, 0x10010174, 0x10010274, 0x10010374, 0x10010474, 0x10010574, 0x10010674,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.2.18 PORT PULL Clear Registers (PxPEC)

PAPEC, PBPEC, PCPEC, PDPEC, PEPEC and PFPEC are six 32-bit PORT PULL clear registers.

	PAPES, PBPEC, PCPEC, PDPEC, PEPEC, PFPEC																0x10010078, 0x10010178, 0x10010278, 0x10010378, 0x10010478, 0x10010578, 0x10010678,													
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

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.	W

25.3 Program Guide

25.3.1 Port Function Guide

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.
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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26 SMB Controller

26.1 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.

26.1.1 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
- 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
- 3 independent SMB channels (SMB0, SMB1, SMB2)

26.1.2 Pin Description

Table 26-1 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.

Totally speaking, for standard-mode SMB-bus system, the pull up resistor depends on following

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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. (Refer to ’ Philips Semiconductor, *THE SMB-BUS SPECIFICATION*, Version 2.1. Jan, 2000’)

26.2 Registers

26.2.1 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 26-2 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	0x10052000	Address base of SMB2
SMB4	0x10052000	Address base of SMB2

Table 26-3 Registers Memory Map-Address Offset

Name	Addr Offset	Description	Width	RW	Reset
SMBCON	0x00	SMB control	8bits	RW	0x7D
SMBTAR	0x04	SMB target address	13bits	RW	0x1055
SMBSAR	0x08	SMB slave address	10bits	RW	0x055
SMBDC	0x10	SMB data buffer and command	9bits	RW	0x000
SMBSHCNT	0x14	Standard speed SMB SCL high count	16bits	RW	0x0190
SMBSLCNT	0x18	Standard speed SMB SCL low count	16bits	RW	0x01D6
SMBFHCNT	0x1C	Fast speed SMB SCL high count	16bits	RW	0x003C
SMBFLCNT	0x20	Fast speed SMB SCL low count	16bits	RW	0x0082
SMBINTST	0x2C	SMB Interrupt Status	12bits	R	0x000
SMBINTM	0x30	SMB Interrupt Mask	12bits	R/W	12’h8FF
SMBRXTL	0x38	SMB RxFIFO Threshold	6 bits	RW	0x1F
SMBTXTL	0x3C	SMB TxFIFO Threshold	6 bits	RW	0x20

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

SMBCINT	0x40	Clear Interrupts	1 bit	R	0x0
SMBCRXUF	0x44	Clear RXUF Interrupt	1 bit	R	0x0
SMBCRXOF	0x48	Clear RX_OVER Interrupt	1 bit	R	0x0
SMBCTXOF	0x4C	Clear TX_OVER Interrupt	1 bit	R	0x0
SMBCRXREQ	0x50	Clear RDREQ Interrupt	1 bit	R	0x0
SMBCTXABT	0x54	Clear TX_ABRT Interrupt	1 bit	R	0x0
SMBCRXDN	0x58	Clear RX_DONE Interrupt	1 bit	R	0x0
SMBCACT	0x5c	Clear ACTIVITY Interrupt	1 bit	R	0x0
SMBCSTP	0x60	Clear STOP Interrupt	1 bit	R	0x0
SMBCSTT	0x64	Clear START Interrupt	1 bit	R	0x0
SMBCGC	0x68	Clear GEN_CALL Interrupt	1 bit	R	0x0
SMBENB	0x6C	SMB Enable	1 bit	RW	0x0
SMBST	0x70	SMB Status register	7 bits	R	0x06
SMBSDAHD	0x7C	SMB SDA HolD time Register	16 bits	RW	0x55
SMBABTSRC	0x80	SMB Transmit Abort Status Register	32 bits	R	0x0000
SMBDMACR	0x88	DMA Control Register	2 bits	R/W	0x0
SMBDMATDLR	0x8c	DMA Transmit Data Level	4 bits	R/W	0x0
SMBDMARDLR	0x90	DMA Receive Data Level	4 bits	R/W	0x0
SMBSDASU	0x94	SMB SDA Setup Register	8 bits	RW	0x64
SMBACKGC	0x98	SMB ACK General Call Register	1 bit	RW	0x1
SMBENBST	0x9C	SMB Enable Status Register	3 bits	R	0x0
SMBFLT	0xA0	SMB Filter Register	8bits	RW	0x1

26.2.2 Registers and Fields Description

26.2.2.1 SMBCON (SMB Control Register)

This resiger can be written only when SMB is disabled. Writes at other time have on effect.

	SMBCON																BASE + 0x00															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	Reserved									SLVDIS	REST	MATP	SATP	SPD		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 © 2005-2015 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	REST	Determines whether RESTART conditions may be sent when acting as a master. 0: disable 1: enable	RW
4	MATP	This bit is a read-only copy of SMBTAR.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	SPD	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

26.2.2.2 SMBTAR (SMB Target Address Register)

	SMBTAR																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 © 2005-2015 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 1.	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 SMBINTST 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 SMBTAR and SMBSAR 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: It is not necessary to perform any write to this register if SMB is enabled as an SMB slave only.

26.2.2.3 SMBSAR (SMB Slave Address Register)

	SMBSAR																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

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

NOTE: It is not necessary to perform any write to this register if SMB is enabled as an SMB master only.

26.2.2.4 SMBDC (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.

SMBDC																BASE + 0x10															
Bit																15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																Reserved					REST	STP	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	REST	This bit controls whether a RESTART is issued before the byte is sent or received. 1: If SMBCON.REST is 1, a RESTERT is issued before the byte is sent or received; if SMBCON.REST is 0, a STOP followed by a START is issued instead. 0: If SMBCON.REST is 1, a RESTERT is issued if the transfer direction is changing from the previous command.	
9	STP	This bit controls whether a STOP is issued after the byte is sent or received. 1: STOP is issued after the byte 0: STOP is not issued after the byte	
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	RW
7:0	DAT	This register contains the data to be transmitted or received on the SMB bus.	RW

NOTE: 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.

26.2.2.5 SMBSHCNT (SMB Standard Speed SCL High Count Register)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBSHCNT															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. Writes at other times have no effect. SCL high time of SMB is (SMBSHCNT + 8) SMB_clk periods.	RW

NOTE: Minimum value allowed for the SMBSHCNT registers is 6. If the set value was less than 6, it will be automatically set to 6.

26.2.2.6 SMBSLCNT (SMB Standard Speed SCL Low Count Register)

	SMBSLCNT															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. Writes at other times have no effect. SCL low time of SMB is (SMBSLCNT + 1) SMB_clk periods.	RW

NOTE: 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.

26.2.2.7 SMBFHCNT (SMB Fast Speed SCL High Count Register)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBFHCNT																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. Writes at other times have no effect.	RW

NOTE: Minimum value allowed for the SMBSHCNT registers is 6. If the set value was less than 6, it will be automatically set to 6.

26.2.2.8 SMBFLCNT (SMB Fast Speed SCL Low Count Register)

	SMBFLCNT																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. Writes at other times have no effect.	RW

NOTE: 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.

26.2.2.9 SMBINTST (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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBINTST																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 SMBCGC 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 regardless of whether SMB is operating in slave or master mode.	R
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 SMBCACT register 3 Reading the SMBCINT 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 SMBABTSRC 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 SMBCTXABT 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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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 SMBDC register. This bit is set to 0 just after the processor reads the SMBCRREQregister.	
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 SMBENB 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 SMBENB = 0, this bit is set to 0.	R
3	TXOF	Set during transmit if the transmit buffer is filled to SMBTX_BUFFER_DEPTH and the processor attempts to issue another SMB command by writing to the SMBDC register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when SMB_en goes to 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 hardware when buffer level goes below the threshold. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (SMBENB[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 SMBENB bit 0 is programmed with a 0, regardless of the activity that continues.	R
1	RXOF	Set if the receive buffer is completely filled to 2 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 (SMBENB[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when SMB_en goes to 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 SMBDC register. If the module is disabled (SMBENB[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when SMB_en goes to 0, this interrupt is cleared.	R

26.2.2.10 SMBINTM (SMB Interrupt Mask Register)

These bits mask their corresponding interrupt status bits. They are active low; a value of 0 prevents a bit from generating an interrupt.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBINTM																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

26.2.2.11 SMBRXTL (SMB Receive FIFO Threshold Register)

	SMBRXTL																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.	RW

26.2.2.12 SMBTXTL (SMB Transmit FIFO Threshold Register)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBTXTL																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.	RW

26.2.2.13 SMBCINT (SMB Clear Combined and Individual Interrupt Register)

	SMBCINT																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 SMBABTSRC register. This bit does not clear hardware clearable interrupts but software clearable interrupts. Refer to Bit 9 of the SMBABTSRC register for an exception to clearing SMBABTSRC.	R

26.2.2.14 SMBCRXUF (SMB Clear RXUF Interrupt Register)

	SMBCRXUF																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 SMBINTST register.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.2.2.15 SMBCRXOF (SMB Clear RXOF Interrupt Register)

	SMBCRXOF																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
0	CRXOF	Read this register to clear the RXOF interrupt (bit 1) of the SMBINTST register.	R

26.2.2.16 SMBCTXOF (SMB Clear TX_OVER Interrupt Register)

	SMBCTXOF																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

26.2.2.17 SMBCRXREQ (SMB Clear RDREQ Interrupt Register)

	SMBCRXREQ																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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

26.2.2.18 SMBCTXABT (SMB Clear TX_ABRT Interrupt Register)

	SMBCTXABT																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

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 SMBINTST register, and the SMBABTSRC 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 SMBABTSRC register for an exception to clearing SMBABTSRC.	R

26.2.2.19 SMBCRXDN (SMB Clear RX_DONE Interrupt Register)

	SMBCRXDN																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 SMBINTST register.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.2.2.20 SMBCACT (SMB Clear ACTIVITY Interrupt Register)

	SMBCACT																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

26.2.2.21 SMBCSTP (SMB Clear STOP Interrupt Register)

	SMBCSTP																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

26.2.2.22 SMBCSTT (SMB Clear START Interrupt Register)

	SMBCSTT																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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

26.2.2.23 SMBCGC (SMB Clear GEN_CALL Interrupt Register)

	SMBCGC																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

26.2.2.24 SMBENB (SMB Enable Register)

	SMBENB																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	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		operation.	
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 SMBINTST 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

26.2.2.25 SMBST 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 SMBENB register:

- Bits 1 and 2 are set to 1
- Bits 3 and 4 are set to 0

When the master or slave state machine goes to idle and ic_en=0:

- Bits 5 and 6 are set to 0

	SMBST																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 1: Slave FSM is not in IDLE state	R
5	MSTACT	Master FSM Activity Status. 0: Master FSM is in IDLE state 1: Master FSM is not in IDLE state	R
4	RFF	Receive FIFO Completely Full. When the receive FIFO is completely full,	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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	
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

26.2.2.26 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.2.2.27 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	31 30 29 28 27 26 25 24 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	RXFLR					
RST	0 0 0 0 0 0 0 0 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
31:6	Reserved	Reserved.	N/A
5:0	RXFLR	Contains the number of valid data entries in the receive FIFO	R

26.2.2.28 SMBSDAHD (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 master and slave mode, in units of SMB_clk 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 is disabled.

	SMBSDAHD	BASE + 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
		SDAHD
RST	0 0 0 0 0 0 0 0 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	Reserved.	N/A
15:0	SDAHD	SMB Hold Time. Sets the required SDA hold time in units of SMB_clk period.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.2.2.29 SMBABTSRC (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 (SMBCON[5]=1), the SPECIAL bit must be cleared (SMBTAR[11]), or the GC_OR_START bit must be cleared (SMBTAR[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.

	SMBABTSRC																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_N	ABRT_TXDATA_N OACK	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

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.	R
16	USER_ABRT	This is a master-mode-only bit. Master has detected the transfer abort	
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 SMBDC register.	R
14	SLV_ARBLOST	1: Slave lost the bus while transmitting data to a remote master. SMBABTSRC[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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

12	ARB_LOST	1: Master has lost arbitration, or if SMBABTSRC[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 (SMBRESTART_EN bit (SMBCON[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 (SMBCON[5]=1), the SPECIAL bit must be cleared (SMBTAR[11]), or the GC_OR_START bit must be cleared (SMBTAR[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 (SMBRESTART_EN bit (SMBCON[5]) = 0) and the user is trying to send a START Byte. Reset value: 0x0.	R
8	ABRT_HS_NORSTRT	1: The restart is disabled (SMBRESTART_EN bit (SMBCON[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).	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Reset value: 0x0.	
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

26.2.2.30 SMBDMACR (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.

	SMBDMACR																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

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	R/W
0	RDEN	**Receive DMA Enable**. This bit enables/disables the receive DMA channel. 0: Receive DMA disabled 1: Receive DMA enabled	R/W

26.2.2.31 SMBDMATDLR (SMB DMA Transmit Data Level Register)

The register is used to config dma transmit data level.

	SMBDMACR																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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

26.2.2.32 SMBDMARDLR (SMB DMA Transmit Data Level Register)

The register is used to config dma receive data level.

	SMBDMACR															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**. 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.	R/W

26.2.2.33 SMBSDASU (SMB SDA Setup Register)

This register controls the amount of time delay (in terms of number of SMB_clk 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 tSU:DAT (NOTE 2) as detailed in the SMB Bus Specification.

NOTE: The length of setup time is calculated using [(SMBSDASU - 1) * (ic_clk_period)], so if the user requires 10 ic_clk periods of setup time, they should program a value of 11.

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. (Refer to ' Philips Semiconductor, *THE SMB-BUS SPECIFICATION*, Version 2.1. Jan, 2000')

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBSDASU																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 SMB_clk frequency of 10 MHz, SMBSDASU should be programmed to a value of 11.	R/W

26.2.2.34 SMBACKGC (SMB ACK General Call Register)

The register controls whether SMB responds with an ACK or NACK when it receives an SMB General Call address.

	SMBACKGC																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 (by asserting ic_data_oe) when it receives a General Call. When set to 0, the SMB does not generate General Call interrupts.	R/W

26.2.2.35 SMBENBST (SMB Enable Status)

The register is used to report the SMB hardware status when the SMBENB register is set from 1 to 0; that is, when SMB is disabled.

If SMBENB has been set to 1, bits 2:1 are forced to 0, and bit 0 is forced to 1.
If SMBENB has been set to 0, bits 2:1 is only be valid as soon as bit 0 is read as '0'.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	SMBENBST															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 SMBENB 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 SMBENB 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 transfer. **NOTE:** The CPU can safely read this bit when IC_EN (bit 0) is read as 0. Reset value: 0x0.	R
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 SMBENB register from 1 to 0. This bit is set when the CPU writes a 0 to the SMBENB 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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		there is master activity, or when the SMB bus is idle. 2 The CPU can safely read this bit when IC_EN (bit 0) is read as 0. Reset value: 0x0.	
0	SMBEN	ic_en Status. This bit always reflects the value driven on the output port ic_en. When read as 1, SMB is deemed to be in an enabled state. When read as 0, SMB is deemed completely inactive. **NOTE:** The CPU can safely read this bit anytime. When this bit is read as 0, the CPU can safely read SLVRDLST (bit 2) and SLVDISB (bit 1). Reset value: 0x0.	R

26.2.2.36 SMBFLT (SMB Filter Register)

	SMBFLT																BASE + 0xA0							
Bit																	7	6	5	4	3	2	1	0
	FILCNT																							
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, of the longest vaule 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.	RW

26.3 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

to 0 and 1, respectively.

26.3.1 SMB Behavior

The SMB can be controlled via software to be either:

- An SMB master only, communicating with other SMB slaves.

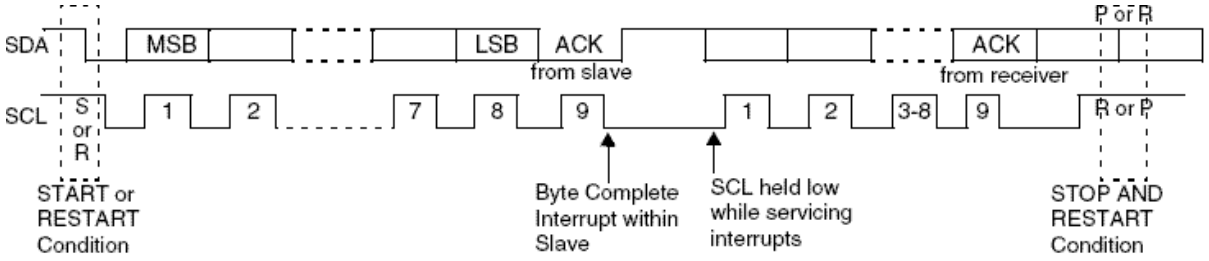
OR

- 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 transaction after the last byte is received, and then the master issues a STOP condition or addresses another slave after issuing a RESTART condition.



26.3.2 Master Mode Operation

This section includes the following topics:

- "Initial Configuration"
- "Dynamic SMBTAR or SMB10BITADDR_MASTER Update"
- "Master Transmit and Master Receive"

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

26.3.2.1 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 SMBCON 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 SMBTAR.

3 Set the expected SCL frequency. SMBCON.SPD = 2'b01, only SMBSHCNT and SMBSLCNT are needed to be configured; SMBCON.SPD = 2'b10, only SMBFHCNT and SMBFLCNT 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 SMBTAR 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).

5 Enable the SMB by writing a 1 into SMBENB register. And wait for the SMBENBST.SMBEN = 1.

6 Now write the transfer direction and data to be sent to the SMBDC register (This can be done by DMA). If the SMBDC 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 SMBCON.STPHLD is 0, then further writes to the TX FIFO results in an independent SMB transaction.

26.3.2.2 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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).

26.3.2.3 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 (SMBDC). The *CMD* bit (SMBDC[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.

26.3.3 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”

26.3.3.1 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 SMBSAR register (bits 9:0) to set the slave address. This is the address to which the SMB responds.
3 Write to the SMBCON 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 SMBENB 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

26.3.3.2 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 SMBSAR 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 SMBINTM[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 SMBINTST register.
   a Reads that indicate SMBINTST[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.

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 SMBCTXABT register before attempting to write into the TX FIFO. See register SMBINTST for more details. If the TX_ABRT interrupt has been masked, due to of SMBINTM[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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

SMBINTST[5] register.

5 Software writes to the SMBDC 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 SMBINTST 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.

26.3.3.3 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 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 SMBINTM[2] register to 0 or setting SMBTX_TL to a value larger than 0, then it is recommended that a timing routine (described in "Slave-Transmitter Operation for a Single Byte" on page 57) be implemented for periodic reads of the "SMBST" on page 136 register. Reads of the SMBST 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 SMBDC 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.

26.3.3.4 Slave-Transfer Operation For Bulk Transfers

In the standard SMB protocol, all transactions are single byte transactions and the programmer

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 SMBINTST register being set to 0, then it is recommended that a timing routine be used to activate periodic reads of the SMBINTST register. Reads of SMBINTST 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.

26.3.4 Disabling SMB

The register SMBENBST is added to allow software to unambiguously determine when the hardware has completely shutdown in response to the SMBENB register being set from 1 to 0.

26.3.4.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

4 The variable POLL_COUNT is initialized to zero.
5 Set SMBENB to 0.
6 Read the SMBENBST register and test the SMB_EN bit (bit 0). Increment POLL_COUNT by one. If POLL_COUNT >= MAX_T_POLL_COUNT, exit with the relevant error code.
7 If SMBENBST[0] is 1, then sleep for tSMB_poll and proceed to the previous step. Otherwise, exit with a relevant success code.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27 Synchronous Serial Interface

27.1 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.

The SSI operates in master mode (the attached peripheral functions as a slave) and supports serial bit rates from 7.2 KHz to 54 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
- 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

27.2 Pin Description

Table 27-1 SSI Controller Pins Description

Name	I/O	Description
SSI_CLK	Output	Serial bit-rate clock
SSI_CE0_	Output	First slave select enable
SSI_CE1_	Output	Second slave select enable
SSI_GPC	Output	General purpose control signal to external chip
SSI_DT	Output	Transmit data (serial data out)
SSI_DR	Input	Receive data (serial data in)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

SSI_GPC is general-purpose control signal, synchronized with SSI_CLK, can be used for LCD control.

27.3 Register Description

27.3.1 Register Mapping

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 27-2 SSI Serial Port Registers

Name	RW	Reset Value	Address Offset	Access Size
SSIDR	RW	0x??	0x10043000	32
SSICR0	RW	0x0000	0x10043004	32
SSICR1	RW	0x00087860	0x10043008	32
SSISR	RW	0x00000098	0x1004300C	32
SSIITR	RW	0x0000	0x10043010	16
SSIICR	RW	0x00	0x10043014	8
SSIGR	RW	0x0000	0x10043018	16
SSIRCNT	RW	0x00000000	0x1004301C	32

27.3.2 SSI Data Register (SSIDR)

	SSIDR																														0x10043000
Bit	31	30	29	28	27	26	25	24	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	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:17	D31-D17	D31-D17 to be written to/read from Transmit/Receive FIFO.	R
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 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.	RW

27.3.3 SSI Control Register0 (SSICR0)

	SSICR0																													0x10043004
Bit	31	30	29	28	27	26	25	24	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												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
31:20	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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td colspan="2">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</td><td></td></tr>
<tr><td>15</td><td>SSIE</td><td colspan="2">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 FIFOs/registers are properly configured and be flush/reset manually when necessary before enabling SSI.</td><td>RW</td></tr>
<tr><td>14</td><td>TIE</td><td colspan="2">This bit enables/disables the transmit-FIFO half-empty interrupt TXI.
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.
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’10 (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.</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: SSIRCNT.RCNT will be used.	
3	TFMODE	0: new fifo empty mode 1: old fifo empty mode	RW
2	TFLUSH	Flush the transmit FIFO when set to 1. Always return 0 when read.	RW
1	RFLUSH	Flush the receive FIFO when set to 1. Always return 0 when read.	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).

27.3.4 SSI Control Register1 (SSICR1)

	SSICR1																													0x10043008
Bit	31	30	29	28	27	26	25	24	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		Reserved	ITFRM	UNFIN	Reserved	FMAT		TTRG				MCOM				RTRG				FLEN					Reserved	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.	RW

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	

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td></td><td><table><tr><td></td><td>SSI_CE1_ is high level valid</td><td></td></tr></table></td><td></td></tr>
<tr><td>29:28</td><td>TFVCK</td><td>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’01, SSICR1.TFVCK is ignored.<table><tr><th>TFVCK[1:0]</th><th>Description</th><th></th></tr><tr><td>00</td><td>half or one SSI_CLK cycle delay time</td><td>Initial value</td></tr><tr><td>01</td><td>1 more SSI_CLK cycle delay is added</td><td></td></tr><tr><td>10</td><td>2 more SSI_CLK cycle delay is added</td><td></td></tr><tr><td>11</td><td>3 more SSI_CLK cycle delay is added</td><td></td></tr></table></td><td>RW</td></tr>
<tr><td>27:26</td><td>TCKFI</td><td>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’01, SSICR1.TFVCK is ignored.<table><tr><th>TCKFI[1:0]</th><th>Description</th><th></th></tr><tr><td>00</td><td>half or one SSI_CLK cycle delay</td><td>Initial value</td></tr><tr><td>01</td><td>1 more SSI_CLK cycle delay is added</td><td></td></tr><tr><td>10</td><td>2 more SSI_CLK cycle delay is added</td><td></td></tr><tr><td>11</td><td>3 more SSI_CLK cycle delay is added</td><td></td></tr></table></td><td>RW</td></tr>
<tr><td>25</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>24</td><td>ITFRM</td><td>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’00 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</td><td>RW</td></tr>
<tr><td>23</td><td>UNFIN</td><td>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’01. (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.</td><td>RW</td></tr>
<tr><td>22</td><td>Reserved</td><td>Writing has no effect, read as zero.</td><td>R</td></tr>
<tr><td>21:20</td><td>FMAT</td><td>These bits set the operating transfer format.</td><td>RW</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			FMAT	**Description**		
			00	Motorola's SPI format	Initial value	
			01	TI's SSP format		
			10	National Microwire 1 format		
			11	National Microwire 2 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				RW
15:12	MCOM	Defines the length of command. Only used when SSICR1.FMAT = b'10 or b'11 (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'b01), 2-bit data length (FLEN = 0) isn't supported. The bit length of one data = FLEN + 2				RW
2	Reserved	Writing has no effect, read as zero.				R
1	PHA	This bit sets the phase of the SSI_CLK from the beginning of a data frame for Motorola's SPI format (SSICR1.FMAT = B'00). 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 Motorola's SPI format during idle state. (SSICR1.FMAT = B'00). 0: SSI_CLK keeps low level when idle 1: SSI_CLK keeps high level when idle				RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.3.5 SSI Status Register (SSISR)

	SSISR																												0x1004300C
Bit	31	30	29	28	27	26	25	24	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								TFIFO-NUM								RFIFO-NUM								END	BUSY	TFF	RFE	TFHE	RFHF	UNDR	OVER
RST	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
31:24	Reserved	Writing has no effect, read as zero.	R
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 Transmit-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.	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. 0: Receive-FIFO is not empty 1: Receive-FIFO is empty	R
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, If SSICR0.TIE = 1, it will generate SSI 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, If SSICR0.RIE = 1, it will generate SSI RXI interrupt	R
1	UNDR	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.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Underrun has not occurred 1: Underrun has occurred, when SSICR0.TEIE is set, it will generate SSI TEI interrupt. Write '0' to clear this bit, writing '1' has no effect	
0	OVER	Receive-FIFO overrun status, new received data will lose. 0: Overrun has not occurred 1: Overrun has occurred; When SSICR0.REIE is set, it will generate SSI REI interrupt. Write '0' to clear this bit, writing '1' has no effect	RW

27.3.6 SSI Interval Time Control Register (SSIITR)

SSIITR																0x10043010															
Bit																15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																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. 0: Use SSI bit clock (SSI_CLK) as the interval counter clock source 1: Use 32K clock as the interval counter clock source	RW
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 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 * device_clock period* When SSIITR.CNTCLK = 1: *Interval time ≥ [CNTCLK clock period] * [Value of IVLTM + 1] + 1 * device_clock period;* *Interval time ≤ [CNTCLK clock period] * [Value of IVLTM + 2] + 2 * device_clock period*	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.3.7 SSI Interval Character-per-frame Control Register (SSIICR)

	SSIICR																												0x10043014							
Bit																									7	6	5	4	3	2	1	0
	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'00 (Motorola's SPI format is selected). SSIICR is ignored for SSICR1.FMAT ≠ B'00. The desired transfer number of characters-per-frame is (SSIICR set value + 1).	RW

27.3.8 SSI Clock Generator Register (SSIGR)

	SSIGR																												0x10043018							
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.3.9 SSI Receive Counter Register (SSIRCNT)

	SSIRCNT																															0x1004301C
Bit	31	30	29	28	27	26	25	24	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

27.4 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.5 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 SPI are full-duplex protocols, Microwire uses a 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.

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.

27.5.1 Motorola's SPI Format Details

27.5.1.1 General Single Transfer Formats

The figures below show the timing of general single transfer format.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

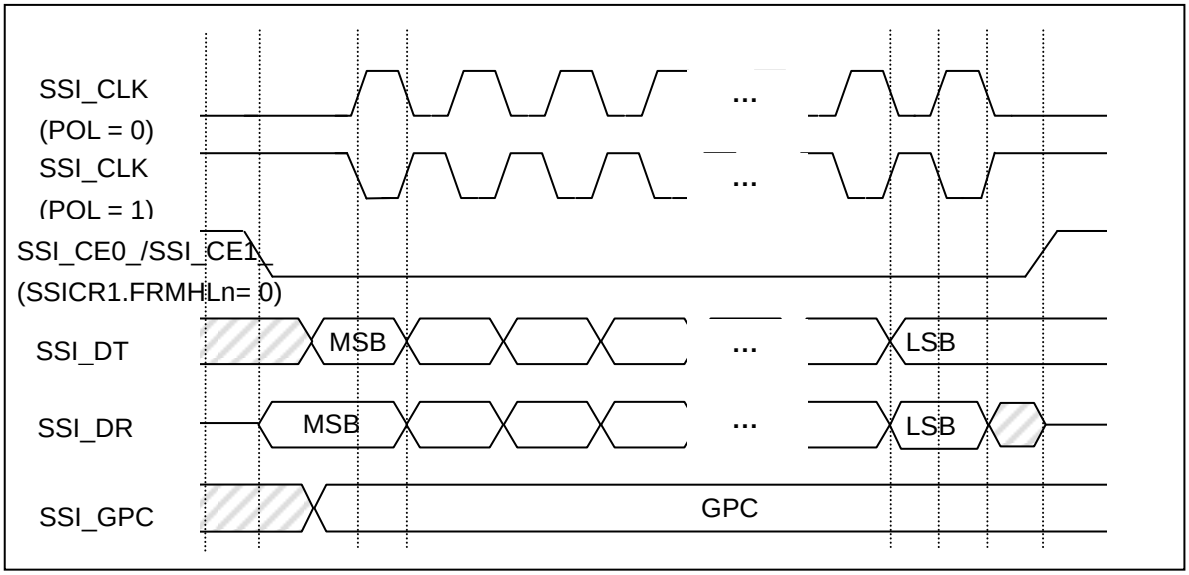


Figure 27-1 SPI Single Character Transfer Format (PHA = 0)

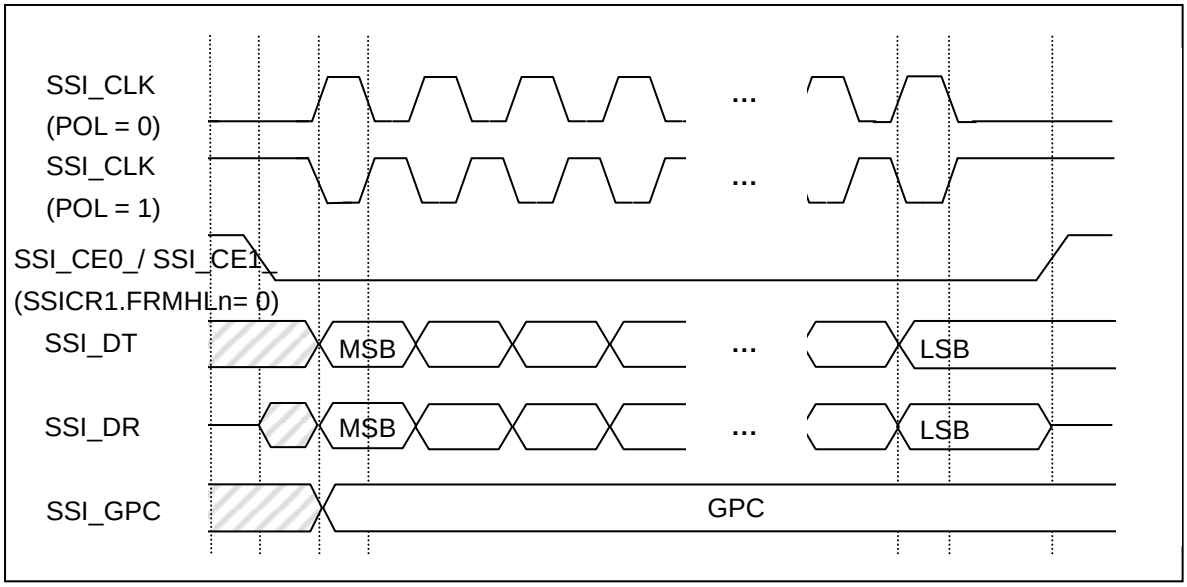


Figure 27-2 SPI Single Character Transfer Format (PHA = 1)

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 SPI protocol, input data on SSI_DR should be stable at every sample clock edge.

Drive data onto SSI_DT 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).

27.5.1.2 Back-to-Back Transfer Formats

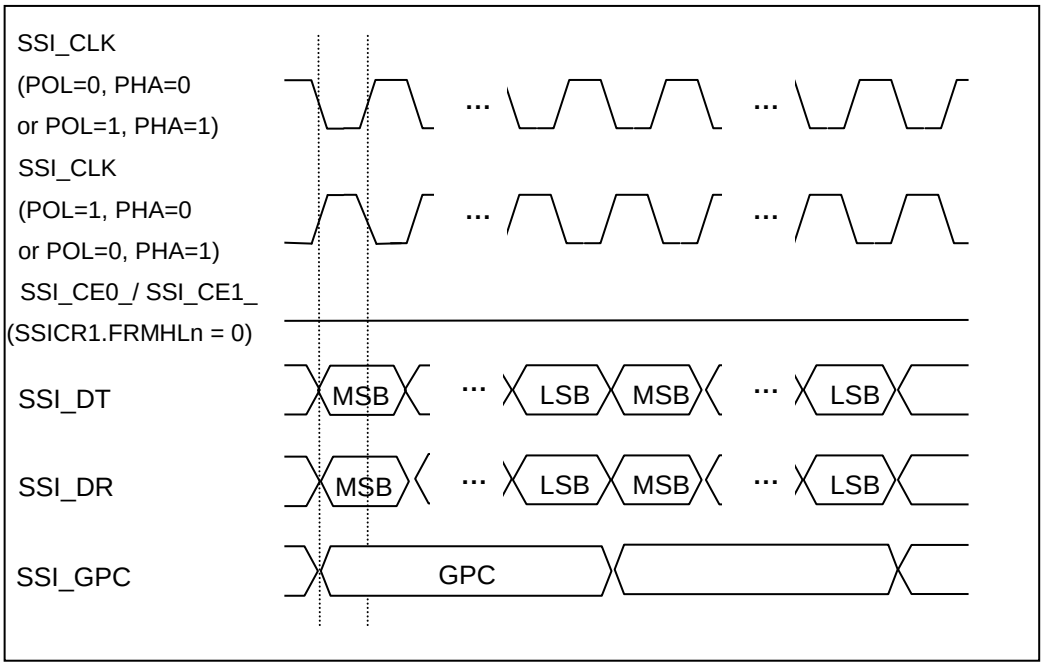


Figure 27-3 SPI Back-to-Back Transfer Format

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.LFST = 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

27.5.1.3 Frame Interval Mode Transfer Format

When in interval mode (SSIITR. IVLTM ≠ '0'), 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.

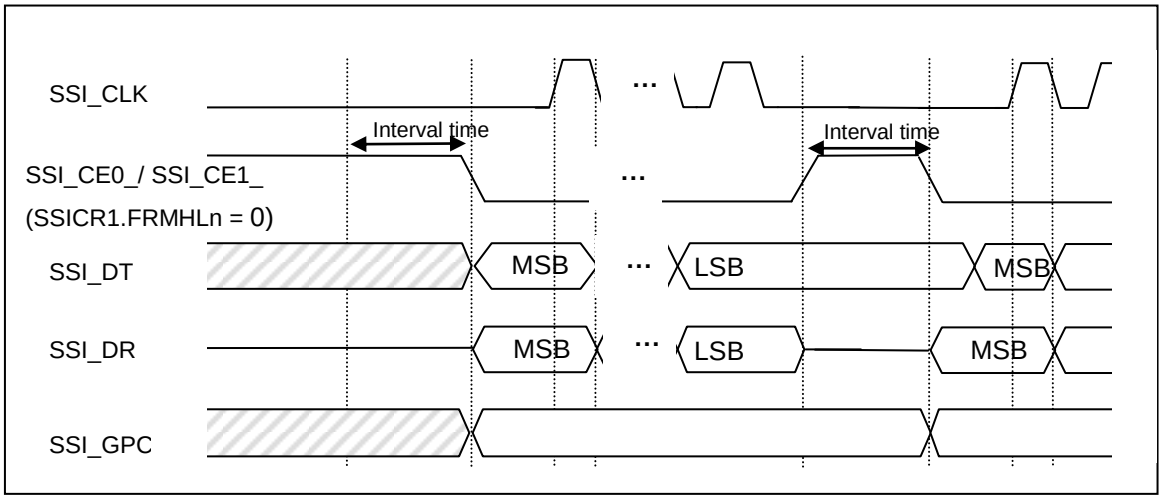


Figure 27-4 SPI Frame Interval Mode Transfer Format (ITFRM = 0, LFST = 0)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

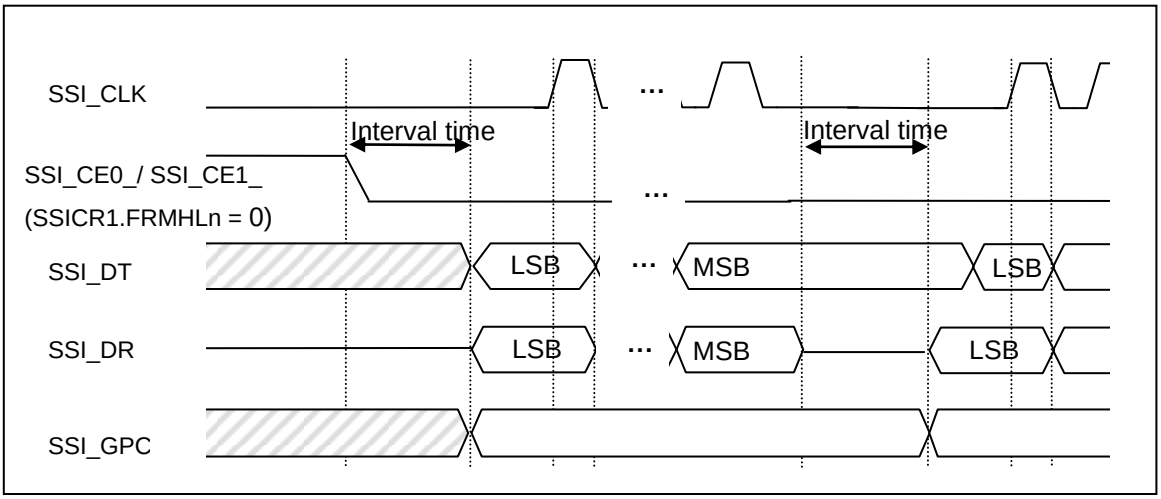


Figure 27-5 SPI Frame Interval Mode Transfer Format (ITFRM = 1, LFST = 1)

27.5.2 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.

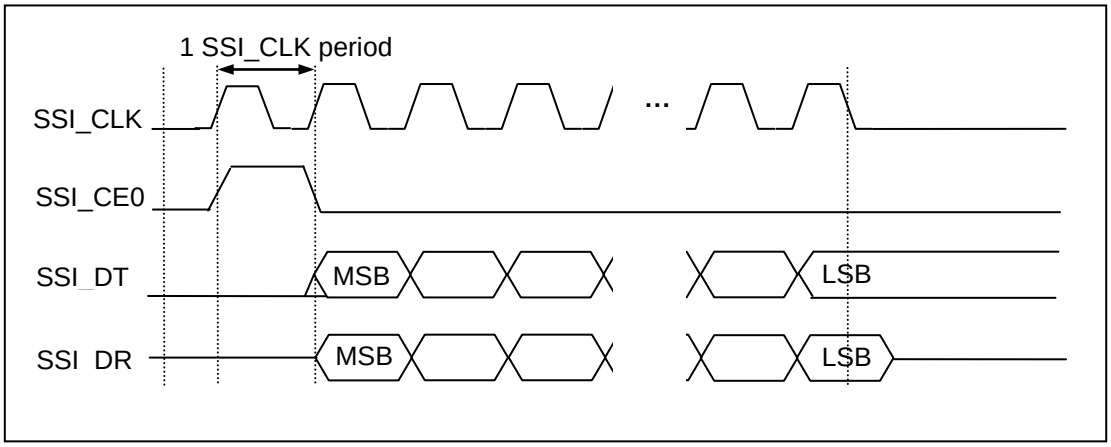


Figure 27-6 TI's SSP Single Transfer Format

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

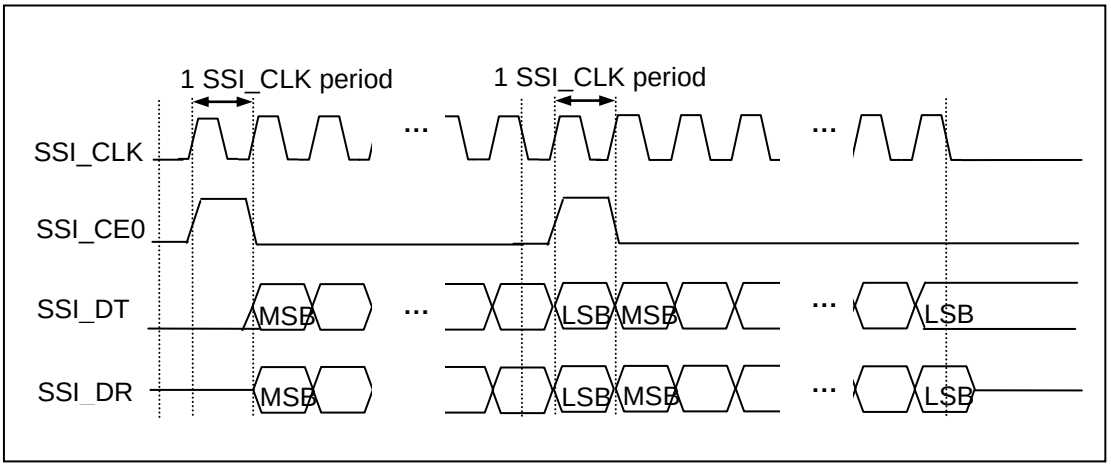


Figure 27-7 TI's SSP Back-to-back Transfer Format

27.5.3 National Microwire Format Details

It supports format 1 and format 2.

Format		Drive Edge	Sample Edge
NM Format 1	Master	Falling Edge	Rising Edge
	Slave	Falling Edge	Rising Edge
NM Format 2	Master	Falling Edge	Falling Edge
	Slave	Rising Edge	Rising Edge

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

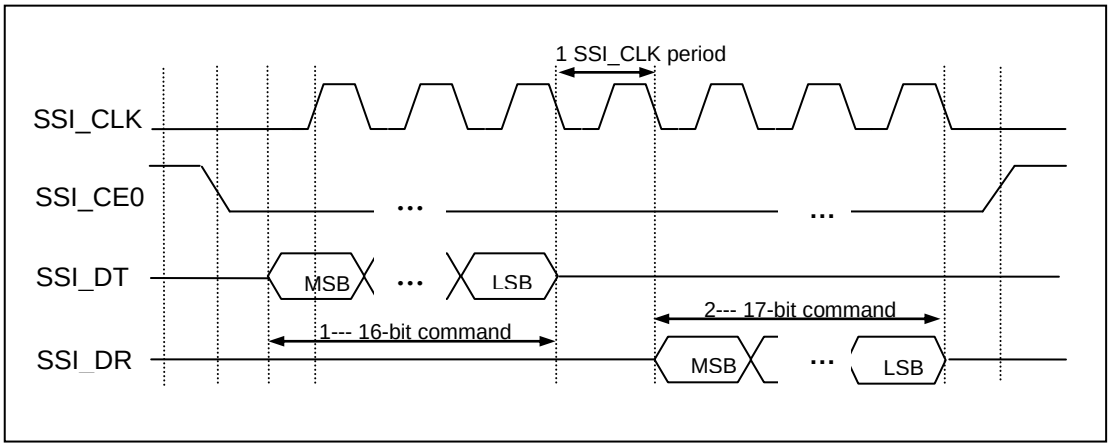


Figure 27-8 National Microwire Format 1 Single Transfer

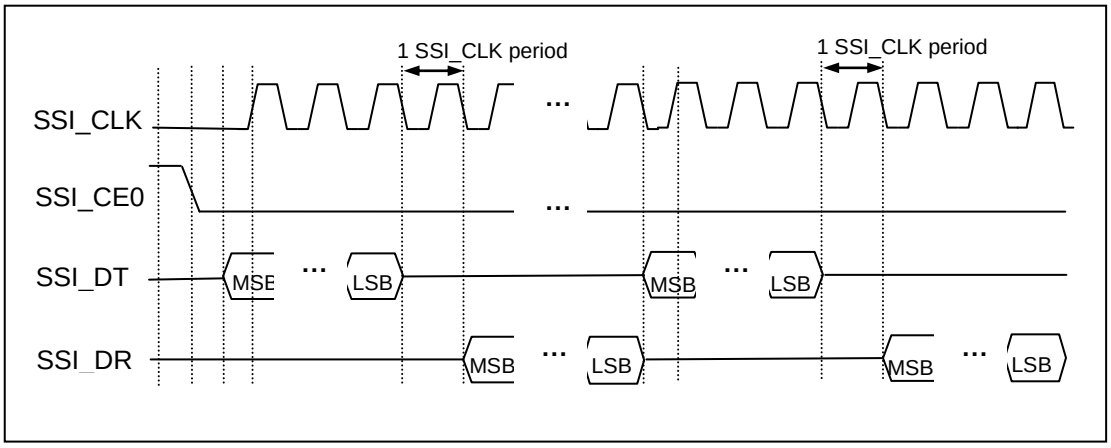


Figure 27-9 National Microwire Format 1 Back-to-back Transfer

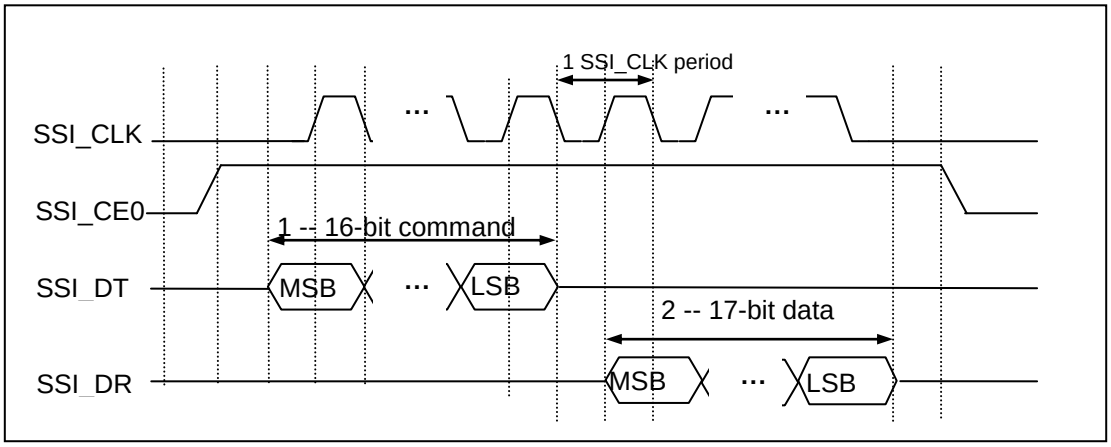


Figure 27-10 National Microwire Format 2 Read Timing

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

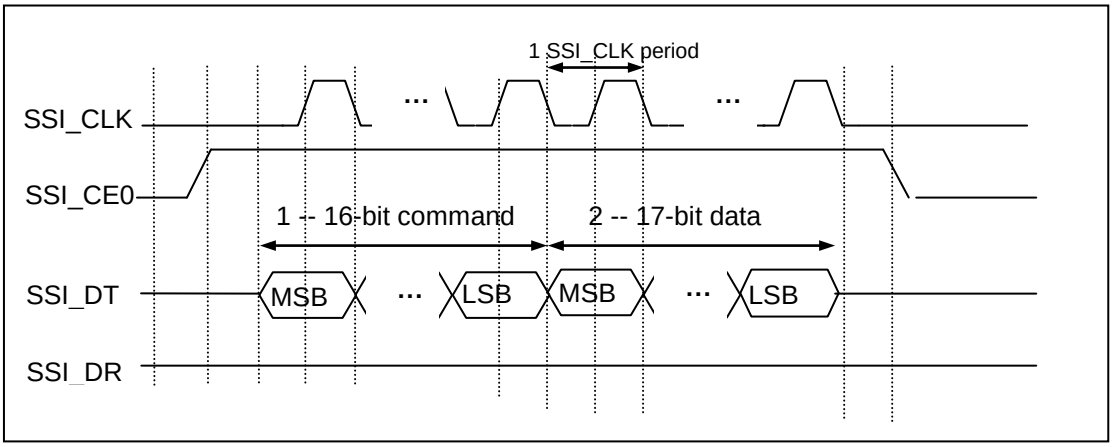


Figure 27-11 National Microwire Format 2 Write Timing

27.6 Interrupt Operation

In SSI, there are TXI, RXI, TEI and REI total 4 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 27-3 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.UNDR	SSICR0.TEIE	**TEI**	Impossible
Receive	R-FIFO is half-full or more	SSISR.RFHF	SSICR0.RIE	**RXI**	Possible
	Receive overrun error	SSISR.OVER	SSICR0.REIE	**REI**	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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

28 UART Interface

28.1 Overview

This chapter describes the universal asynchronous receiver/transmitter (UART) serial ports. There are four 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.

28.1.1 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

28.1.2 Pin Description

Table 28-1 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: UART3, UART1, UART0 support RxD, TxD, RTS_, CTS_.
UART2 supports RxD, TxD.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.2 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.

28.2.1 Register Mapping

There are 4 UART controllers in chip, name UART0, UART1, UART2, UART3 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.

Table 28-2 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

28.2.2 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 © 2005-2015 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

28.2.3 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

28.2.4 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 © 2005-2015 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

28.2.5 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.** 0: Disable the transmit data request interrupt	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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

28.2.6 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 28-3 for details.	R
0	INPEND	**Interrupt Pending.** 0: interrupt is pending 1: No interrupt pending	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 28-3 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

28.2.7 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: 8 0b10: 16 0b11: 32	W

Copyright © 2005-2015 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

28.2.8 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 © 2005-2015 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

28.2.9 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 © 2005-2015 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 © 2005-2015 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

28.2.10 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 © 2005-2015 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

28.2.11 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																								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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.2.12 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																									?	?	?	?	?	?	?	?

28.2.13 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 © 2005-2015 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

28.2.14 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.

28.2.15 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.2.16 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

28.2.17 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

28.3 Operation

The following sections describe the UART operations that include flow of configuration, data transmission, data reception, and Infrared mode.

28.3.1 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 © 2005-2015 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.

28.3.2 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.

28.3.3 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

28.3.4 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.

28.3.5 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.

28.3.6 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.

28.3.7 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 © 2005-2015 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.

28.3.8 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

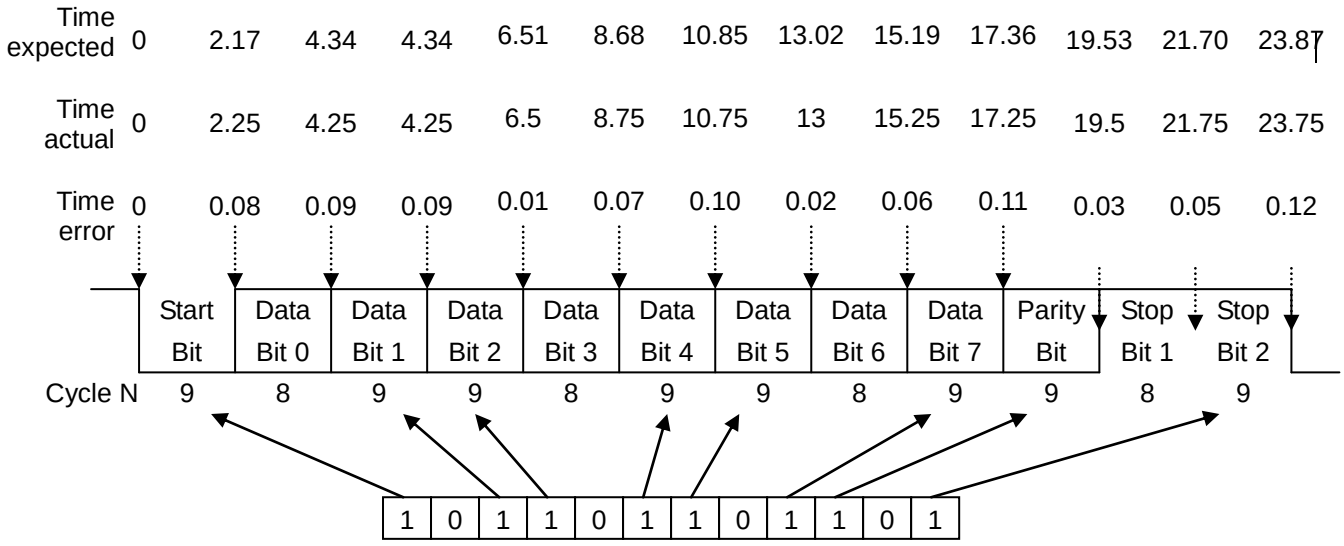
A 12-bit register is used to indicate where to insert the extra cycles.

The first line is the time expected

The second line is the time actual

The third line is the time error

Copyright © 2005-2015 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 © 2005-2015 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29 MMC/SD CE-ATA Controller

29.1 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 MultiMediaCard 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.

29.2 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
- 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
- Maskable 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

29.3 Pins Description

- MSC_CLK, output, host to card clock signal.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

- MSC_CMD, inout, bidirectional command/response signal.
- MSC_DAT[7:0], inout, bidirectional data bus.

29.4 Block Diagram

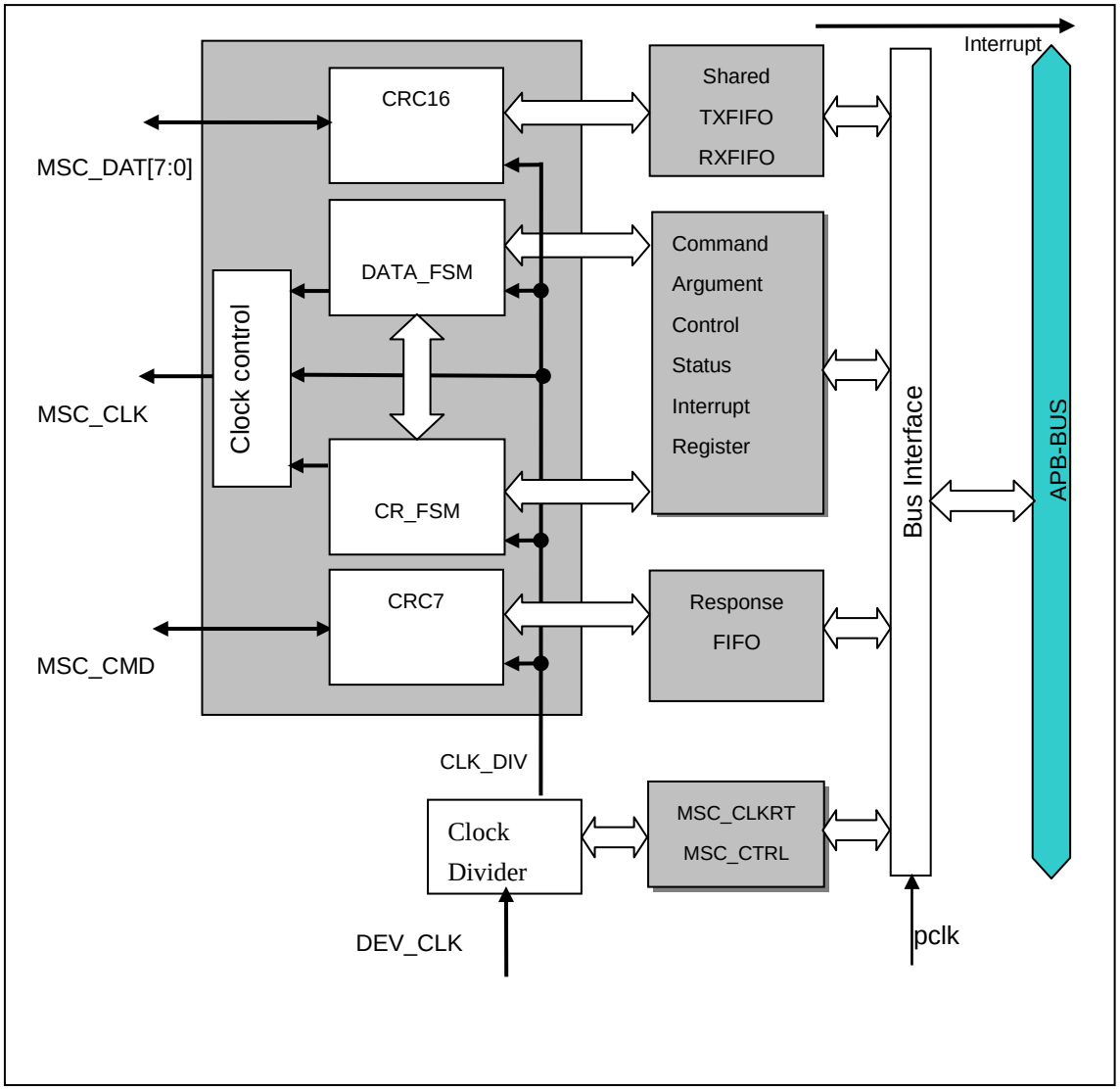


Figure 29-1 MMC/SD Controller Block Diagram

29.5 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

succeeded by an end bit (‘1’). The total length is 48 bits and protected by CRC bits.

Table 29-1 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 29-2 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

29.6 Register Description

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 29-3 MMC/SD Controller Registers Map

Name	RW	Reset Value	Address Offset	Access Size
MSC_CTRL	W	0x0000	0x00	16
MSC_STAT	R	0x00000040	0x04	32
MSC_CLKRT	RW	0x0000	0x08	16
MSC_CMDAT	RW	0x00000000	0x0c	32
MSC_RESTO	RW	0x100	0x10	16
MSC_RDTO	RW	0xFFFFFF	0x14	32
MSC_BLKLEN	RW	0x0000	0x18	16
MSC_NOB	RW	0x0000	0x1c	16

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

MSC_SNOB	R	0x????	0x20	16
MSC_IMASK	RW	0xFFFFFFFF	0x24	32
MSC_IFLG	RW	0x2000	0x28	32
MSC_CMD	RW	0x00	0x2c	8
MSC_ARG	RW	0x00000000	0x30	32
MSC_RES	R	0x????	0x34	16
MSC_RXFIFO	R	0x????????	0x38	32
MSC_TXFIFO	W	0x????????	0x3c	32
MSC_LPM	RW	0x00000000	0x40	32
MSC_DMAC	RW	0x00000000	0x44	32
MSC_DMANDA	RW	0x00000000	0x48	32
MSC_DMADA	RW	0x00000000	0x4c	32
MSC_DMALEN	RW	0x00000000	0x50	32
MSC_DMACMD	RW	0x00000000	0x54	32
MSC_CTRL2	RW	0x00800000	0x58	32
MSC_RTCNT	R	0x00000000	0x5c	32

Note:

There are three MSC controllers in the chip, named MSC0, MSC1 and MSC2.
The base address of MSC0 controller is 0x13450000.
The base address of MSC1 controller is 0x13460000.
The base address of MSC2 controller is 0x13470000.

29.6.1 MSC Control Register (MSC_CTRL)

	MSC_CTRL																											0x00
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_READWAIT	STOP_READWAIT	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	W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		enabled in CE_ATA devices. Once the CCSD pattern is sent to device, host automatically clears the SEND_CCSD bit.	
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, read as zero.	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
5	START_READWAIT	Only used for SDIO ReadWait. Start the ReadWait cycle. 0: No effect 1: Start ReadWait	W
4	STOP_READWAIT	Only used for SDIO ReadWait. Stop the ReadWait cycle. 0: No effect 1: Start ReadWait	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.6.2 MSC Control 2 Register (MSC_CTRL2)

	MSC_CTRL2																														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
	Reserved			PIP					RST_EN	Reserved																		STPRM	SVC	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	RST_EN	Control the level of pin RST_n to reset the card 0: Disable reset or do nothing 1: Enable reset to card	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	SVC	Signal Voltage Change Control 1: Start signal voltage change from 3.3V to 1.8V. Hardware will stop MSC_CLK at low. 0: Stop signal voltage change from 3.3V to 1.8V. Hardware will start to provide MSC_CLK at 1.8 V. 1.8V regulator output should be stable within 5ms when changing voltage from 3.3V to 1.8V. 3.3V regulator output should be stable within 5ms when changing voltage from 1.8V to 3.3V.	RW
2:0	SMS	Speed Mode Selection 000: Default speed 001: High speed 010: SDR12 011: SDR25 100: SDR50 Others: reserved	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.6.3 MSC Status Register (MSC_STAT)

	MSC_STAT																													0x04
Bit	31	30	29	28	27	26	25	24	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	AUTO_CMD23_DONE	SVS	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_READWAIT	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	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	AUTO_CMD23_DONE	Indicates that the auto command CMD23 is finished 0: Not finished 1: Finished	R
29	SVS	Signal Voltage Change Status 1: MSC_CLK starts stopping at low 0: MSC_CLK has no change of stopping at low	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	Writing has no effect, 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. 1: Boot acknowledge received.	R
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.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Reset has been finished 1: Reset has not been finished	
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_READWAIT	Indicates whether SDIO card has entered ReadWait State. 0: Card has not entered ReadWait 1: Card has entered ReadWait	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
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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		01: Card observed erroneous transmission of data 10: No CRC status is sent back 11: Reserved	
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

29.6.4 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																0x08															
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

29.6.5 MSC Command and Data Control Register (MSC_CMDAT)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MSC_CMDAT																														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
	CCS_EXPECTED	READ_CEATA	Reserved		DIS_BOOT	ENB_BOOT	EXP_BOOT_ACK	BOOT_MODE	Reserved					AUTO_CMD23	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	0	0	0	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	ENB_BOOT	This bit is used for Mandatory boot operation mode only. 0: Do nothing 1: Start Mandatory boot	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: Mandatory boot operation 1: Alternative boot operation	RW
23:19	Reserved	Writing has no effect, read as zero.	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

18	AUTO_CMD23	This field controls use of auto command functions. It can be used only when the card supports CMD23. 0: Auto CMD23 is disabled 1: MSC controller will issue CMD23 automatically before issuing a command specified in MSC_CMD register. when stop command has finished, it is auto cleared 0 by hardware.	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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

8	Reserved	Writing has no effect. Read as zero.	R
7	INIT	80 initialization clocks. 0: Issues command directly 1: Issues 80 clocks before command	W
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

29.6.6 MSC Response Time Out Register (MSC_RESTO)

	MSC_RESTO																0x10													
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 © 2005-2015 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

29.6.7 MSC Read Time Out Register (MSC_RDTO)

	MSC_RDTO																														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
	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

29.6.8 MSC Block Size Register (MSC_BLKLEN)

	MSC_BLKLEN																														0x18
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

29.6.9 MSC Block Count Register (MSC_NOB)

	MSC_NOB																														0x1C
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 © 2005-2015 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

29.6.10 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																0x20															
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

29.6.11 MSC Interrupt Mask Register (MSC_IMASK)

	MSC_IMASK																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
	DMA_DATA_DONE	AUTO_CMD23_DONE	SVS	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	AUTO_CMD23_DONE	Mask the interrupt AUTO_CMD23_DONE. 0: Not masked 1: Masked	RW
29	SVS	Mask the interrupt of SVS	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Not masked 1: Masked	
28:24	PIN_LEVEL	Mask the interrupt of PIN_LEVEL separately 0: Not masked 1: Masked	RW
23	WR_ALL_DONE	Mask the interrupt DMA_DATA_DONE 0: Not masked 1: Masked	RW
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.	WR

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: Not masked 1: Masked	
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 1: Masked	WR
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

29.6.12 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																													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
	DMA_DATA_DONE	AUTO_CMD23_DONE	SVS	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	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	AUTO_CMD23_DONE	Indicates AUTO_CMD23_DONE interrupt.	RW

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		0: the interrupt is not detected 1: the interrupt is detected	
29	SVS	Indicates SVS interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
28:24	PIN_LEVEL	Indicates PIN_LEVEL interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
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 1: the interrupt is detected	RW
22:21	Reserved	Indicates AUTO_CMD23_DONE interrupt. 0: the interrupt is not detected 1: the interrupt is detected	R
20	BCE	Boot CRC error. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
19	BDE	Boot data end. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
18	BAE	Indicates the BAE interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
17	BAR	Indicates the BAR interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
16	DMAEND	Indicates the DMA end interrupt. 0: the interrupt is not detected 1: the interrupt is detected Write 1 to clear.	RW
15	AUTO_CMD12_DONE	Indicates AUTO_CMD12_DONE interrupt. 0: the interrupt is not detected 1: the interrupt is detected	RW
14	DATA_FIFO_FULL	Indicate data FIFO is full interrupt. 0: the interrupt is not detected	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		1: the interrupt is detected	
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	RW
11	CRC_READ_ERR	Indicate CRC read error interrupt. 0: the interrupt is not detected 1: the interrupt is detected	RW
10	CRC_WRITE_ERR	Indicate CRC write error interrupt. 0: the interrupt is not detected 1: the interrupt is detected	RW
9	TIME_OUT_RES	Indicate response time out interrupt. 0: the interrupt is not detected 1: the interrupt is detected	RW
8	TIME_OUT_READ	Indicate read time out interrupt. 0: the interrupt is not detected 1: the interrupt is detected	RW
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.	WR
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.	WR
0	DATA_TRAN_DONE	Indicates whether data transfer is done.	WR

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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.	

29.6.13 MSC Command Index Register (MSC_CMD)

	MSC_CMD																												0x2C	
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

29.6.14 MSC Command Argument Register (MSC_ARG)

	MSC_ARG																												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
	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

29.6.15 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The first half-word read from the response FIFO is the most significant half-word of the received response.

	MSC_RES																0x34															
Bit																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																	DATA															
RST																	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
15:0	DATA	Contains the response to the command that is sent by the MMC/SD controller.	R

29.6.16 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																															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
	DATA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Bits	Name	Description	RW
31:0	DATA	One word of read data.	R

29.6.17 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																															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
	DATA																															
RST	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Bits	Name	Description	RW
31:0	DATA	One word of write data.	W

29.6.18 MSC Low Power Mode Register (MSC_LPM)

The MSC_LPM is used to control whether MSC controller enters Low-Power Mode.

	MSC_LPM																														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
	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

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 start_clock and stop clock can be not use.	RW

29.6.19 MSC DMA Control Register (MSC_DMAC)

The MSC_DMAC register is used to control the DMA transfer.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MSC_DMAC																													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																								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. 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

29.6.20 MSC DMA Descriptor Address Register (MSC_DMANDA)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	MSC_DMANDA																														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
	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

29.6.21 MSC DMA Data Address Register (MSC_DMADA)

	MSC_DMADA																														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
	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

29.6.22 MSC DMA Data Length Register (MSC_DMALEN)

	MSC_DMALEN																														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
	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	R

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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.	

29.6.23 MSC DMA Command Register (MSC_DMACMD)

	MSC_DMACMD																														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
	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

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.6.24 MSC RTFIFO Data Counter Register(MSC_RTCNT)

	MSC_RTCNT																														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
	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

29.7 MMC/SD Functional Description

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.

29.7.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

MSC Card Reset

The command Go_Idle_State, CMD0 is the software reset command for MMC and SD Memory Card, and sets each card into Idle State regardless of the current card state; while in SDIO card, CMD52 is used to write IO reset in CCCR. The cards are initialized with a default relative card address (RCA=0x0000) and with a default driver stage register setting (lowest speed, highest driving current capability).

29.7.2 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 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.

29.7.3 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 bitstream. 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.

29.7.4 Card Access

29.7.4.1 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.7.4.2 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.

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.

29.7.4.3 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.

29.7.4.4 Wide Bus Selection/Deselection

Wide Bus (4 bit bus width) operation mode may be selected / deselected using ACMD6. The default

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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).

29.7.5 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.

29.7.5.1 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.

29.7.5.2 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.

29.7.5.3 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 29-4 Command Data Block Structure

Byte #	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Rsv	Rsv	Rsv	Rsv	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.

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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

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.

29.7.6 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 29-5 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 the command. 0: No Error 1: Error	C
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	C

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

			1: Protected	
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 1: failure	C
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 1: ready 2: ident 3: stby 4: tran 5: data 6: rcv 7: prg 8 : dis (9 – 15) : rsv	B
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	-	-	-

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.7.7 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 29-6 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 operation. 0: Not in the Mode 10: In the mode	A
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.			

29.7.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.7.8.1 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.

29.7.8.2 SDIO Suspend/Resume

Within a multi-function SDIO or a Combo (Mix IO and Memory) card, there are multiple devices (I/O 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.

29.7.8.3 SDIO Read Wait

The optional Read Wait (RW) operation is defined only for the SD 1-bit and 4-bit modes. The read wait

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.7.9 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).

29.7.10 Application Specified Command Handling

The MultiMediaCard/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 MultiMediaCard standard commands and it may have the same CMD number. The card will recognize it 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 MultiMediaCard 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 MultiMediaCard illegal command error.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 ('tran_state') 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).

29.8 MMC/SD Controller Operation

29.8.1 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.

29.8.1.1 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 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.

29.8.1.2 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 underrun. 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.

29.8.2 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.

29.8.2.1 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 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.2.2 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 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

29.8.2.3 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
```


Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.2.4 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.

29.8.3 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.

29.8.4 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:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.5 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.

29.8.5.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.5.2 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 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.

29.8.5.3 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 goto 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, goto 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Step 9. Finish.

29.8.6 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.

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.

29.8.7 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].

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.8 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 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 29-7 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	Stop writing in advance (not	1 Set MSC_CTRL [EXIT_MULTIPLE].

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

infinite	write MSC_NOB blocks)	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.

29.8.9 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.

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 29-8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.8.10 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.
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.

29.8.11 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].

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

29.8.12 Erase, Select/Deselect and Stop

For CMD7 (SELECT/DESELECT_CARD), CMD12 (STOP_TRANSMISSION) and CMD38 (ERASE), 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

29.8.13 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.

29.8.14 SDIO ReadWait

The actual ReadWait 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 ReadWait.
2 Write MSC_CTRL [START_READWAIT] with 1.
3 Wait until MSC_STAT [IS_READWAIT] becomes 1.
4 Send CMD52 to query card status.
5 Write MSC_CTRL [STOP_READWAIT] with 1.

29.8.15 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 29-9 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					

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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				
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				

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30 OTG Controller

30.1 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

30.2 Pin Description

Table 30-1 OTG Pins Description

Name	Type	Description
DP	Inout	Data Positive
DM	Inout	Data Minus
ID	Inout	Identification
DRVVBUS_PE10	Out	Charge pump enable

30.3 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 © 2005-2015 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.

30.4 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.

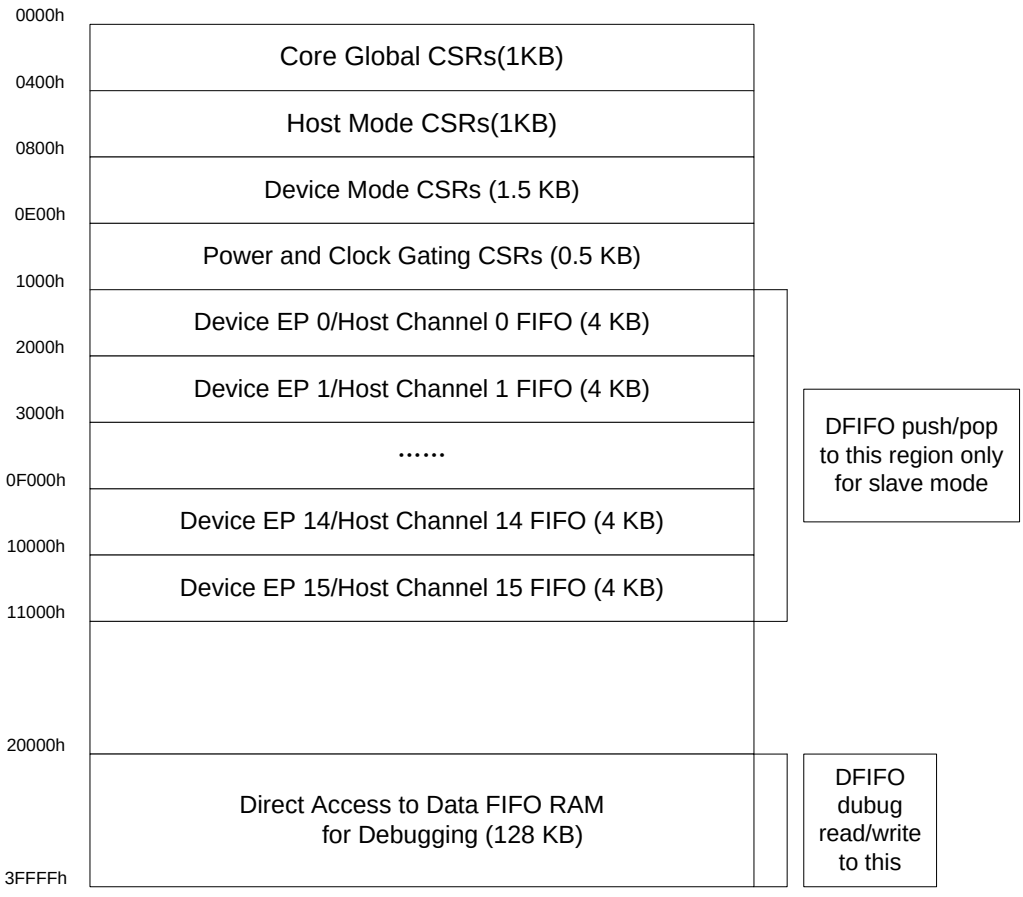


Figure 30-1 OTG CSR Memory Map

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.4.1 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.

30.4.2 Global CSR Map

These registers are available in both Host and Device modes.

Table 30-2 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 © 2005-2015 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

30.4.3 Host Mode CSR Map

These registers must be programmed every time the core changes to Host mode.

Table 30-3 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.4.4 Device Mode CSR Map

These registers must be programmed every time the core changes to Device mode.

Table 30-4 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)

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

DIEPINTn		Device Endpoint-n Interrupt Register (DIEPINTn/DOEPINTn)
	90Ch	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)

30.4.5 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 30-5 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
...	...	...
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

30.4.6 Interrupt Hierarchy

Following figure displays the DWC_otg interrupt hierarchy.

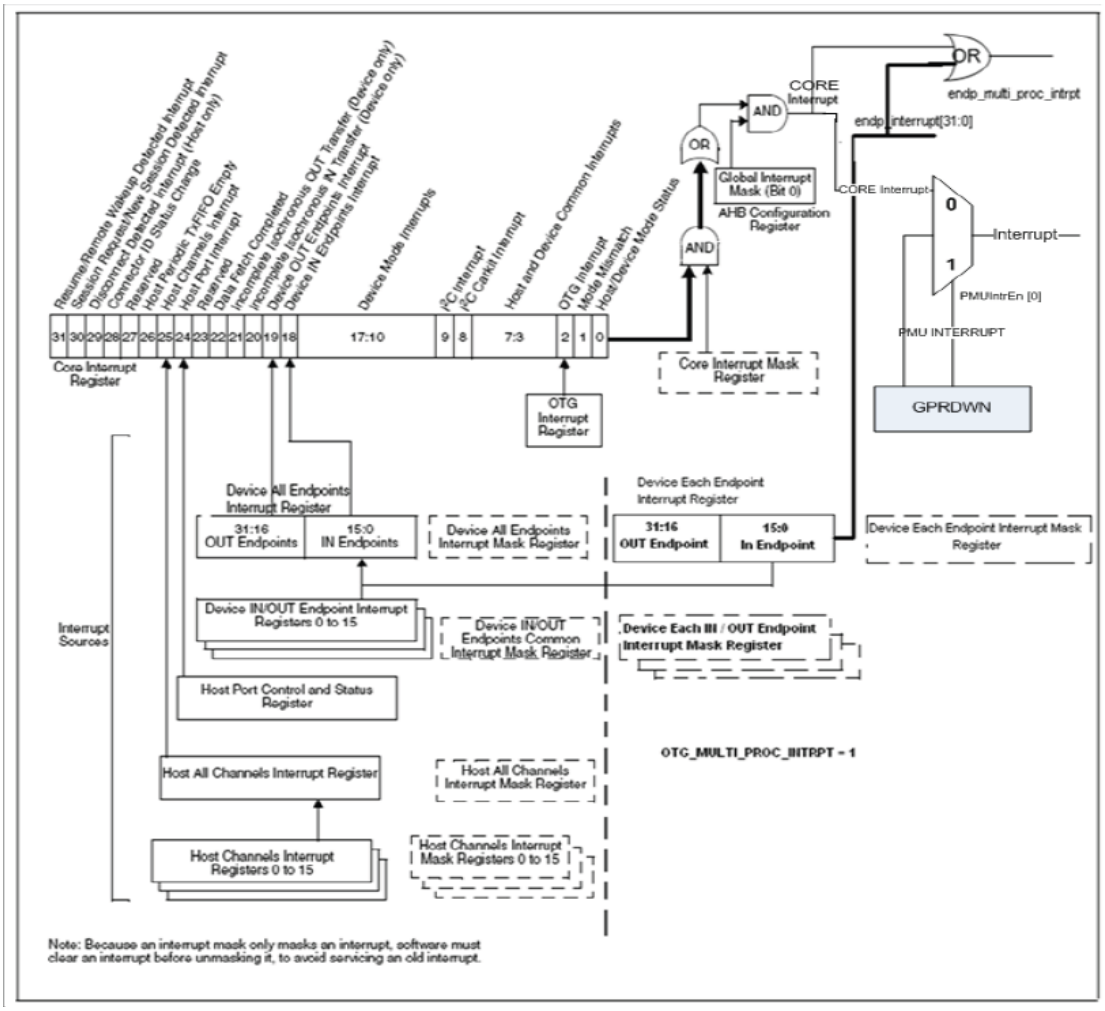


Figure 30-2 Interrupt Hierarchy

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.4.6.1 The Core Interrupt Handler

Following figure illustrates how the core interrupt handler works.

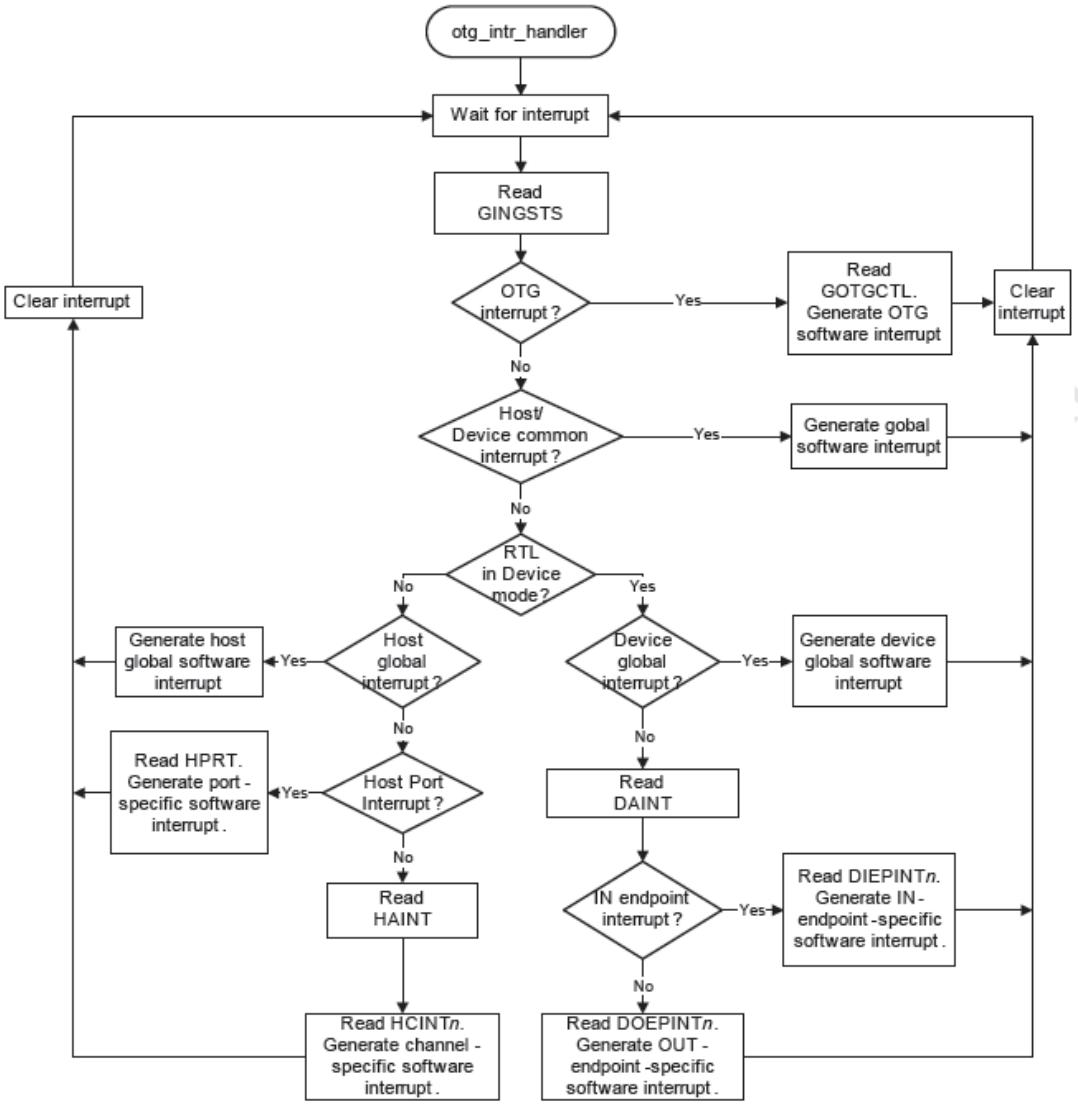


Figure 30-3 Core Interrupt Handler

30.5 Register Descriptions

This section describes Core Global, Device Mode, Host Mode CSRs. Detailed register programming examples are provided in the Programming Guide.

30.5.1 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 © 2005-2015 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).

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.2 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.

Table 30-6 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		▪ 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.
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		▪ GAHBCFG.DMAEn=1, HCFG.DescDMA=1 => Scatter/Gather DMA mode In non Scatter/Gather DMA mode, this bit is 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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.
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-

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		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 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.

30.5.3 Global Registers

These registers are available in both Host and Device modes, and do not need to be reprogrammed when switching between these modes.

30.5.3.1 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 30-7 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	Host and Device	Configurable. The reset value is unknown if OTG_BC_SUPPORT	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	These bits are valid only if OTG_BC_SUPPORT=1, otherwise they are reserved.		=1 and 0 otherwise.	
21	Reserved	Host and Device		RO
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		RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Device		
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	Device Only OTG configurations	1'b0	R_W
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

5	A-Peripheral Session Valid OverrideValue (AvalidOvVal) This bit is used to set the Override value for Avalid signal when GOTGCTL.AvalidOvEn is set. ▪ 1'b0: Avalid value is 1'b0 when GOTGCTL.AvalidOvEn = 1. ▪ 1'b1: Avalid value is 1'b1 when GOTGCTL.AvalidOvEn = 1.	Host only	1'b0	R_W
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

0	Session Request Success (SesReqScs) The core sets this bit when a session request initiation is successful. ▪ 1'b0: Session request failure ▪ 1'b1: Session request success	Device only	1'b0	RO

30.5.3.2 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 30-8 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	Host and Device	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	success or failure.			
8	Session Request Success Status Change (SesReqSucStsChng) 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.	Host and Device	1'b0	R_SS_WC
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

30.5.3.3 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 30-9 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)	Host and	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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.	Device		
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		RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

		Device		
8	Periodic TxFIFO Empty Level (PTxFEmpLvl) Indicates when the Periodic TxFIFO Empty Interrupt bit in 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	Host only	1'b0	R_W
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	Host and Device	4'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 4'b0010: 8 words ▪ 4'b0011: 16 words ▪ 4'b0100: 32 words ▪ 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

30.5.3.4 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 30-10 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	Host and Device	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.			
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	Host Only	1'b0	R_W/RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	in the Vbus State in the RX CMD. Please refer to the ULPI Specification for more detail. ▪ 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	Host and Device	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	register on the ULPIPHY. ▪ 1'b0: PHY does not use AutoResume feature. ▪ 1'b1: PHY uses AutoResume feature. 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 ^2C 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	Device Only	4'h5	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	USB turnaround time is not critical, these bits can be programmed to a larger value.			
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. If HNP 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	RO/R_W
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 takes effect only if GUSBCFG.PHYSel = 1'b0.	Host and Device	1'h0	RO/R_W
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.3.5 Reset Register (GRSTCTL)

✧ Offset: 010h

The application uses this register to reset various hardware features inside the core.

Table 30-11 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	Host and Device	5'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 5'h10: – Flush all the transmit FIFOs in device or host mode.			
5	TxFIFO Flush (TxFFlsh) 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.	Host and Device	1'b0	R_WS_SC
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)	Host and	1'b0	R_WS_S

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Resets the hclk and phy_clock domains as follows: ▪ Clears the interrupts and all the CSR registers except the following register bits: – PCGCCTL.RstPdwnModule – 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	Device		C

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 reset for proper operation.			

30.5.3.6 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 30-12 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)	Host and	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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. 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.	Device		
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	Host Only	1'b0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	HCINTn register to clear this bit.			
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 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.	Host Only	1'b0	RO
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	Device Only	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	IN NAK handshake.			
21	Incomplete Periodic Transfer (incomplP) In Host mode, the core sets this interrupt bit when there are incomplete periodic transactions still pending which are 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 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.	Device only	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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)	Host	1'b0	R_SS_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This field is used only if the Carkit interface was enabled. Otherwise, reads return 0.	and Device		C
7	Global OUT NAK Effective (GOUTNakEff) Indicates that the Set Global OUT NAK bit in the Device 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).	Device only	1'b0	RO
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	Host and Device	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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

30.5.3.7 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 30-13 Interrupt Mask Register: GINTMSK

Field	Description	Mode	Reset	Access
31	Resume/Remote Wakeup Detected Interrupt Mask (WkUpIntMsk)	Host and Device	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

11	USB Suspend Mask (USBSuspMsk)	Device only	1'b0	R_W
10	Early Suspend Mask (ErlySuspMsk)	Device only	1'b0	R_W
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

30.5.3.8 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Following table shows Host mode.

Table 30-14 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 30-15 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'b0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 2'b10: DATA1 ▪ 2'b01: DATA2 ▪ 2'b11: MDATA		
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

30.5.3.9 Receive FIFO Size Register (GRXFSIZ)

✧ Offset: 024h

The application can program the RAM size that must be allocated to the RxFIFO.

Table 30-16 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

30.5.3.10 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 30-17 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	Configurable	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Transmit FIFO RAM. ▪ The application can write a new value in this field. Programmed values must not exceed the power-on value set		

Table 30-18 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

30.5.3.11 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 30-19 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.	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 8'h0: Non-periodic Transmit Request Queue is full ▪ 8'h1: 1 location available ▪ 8'h2: 2 locations available ▪ 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

30.5.3.12 User HW Config1 Register (GHWCFG1)

✧ Offset: 044

This register contains the logical endpoint direction(s) selected.

Table 30-20 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

30.5.3.13 User HW Config2 Register (GHWCFG2)

✧ Offset: 048h

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

This register contains configuration options selected.

Table 30-21 User HW Config2 Register: GHWCFG2

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:	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	OTG_EN_PERIO_HOST".		
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 specifies 16 channels. This field is configured using parameter "Name: OTG_NUM_HOST_CHAN".	Configurable	RO
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)	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 3'b011: SRP-Capable Device ▪ 3'b100: Non-OTG Device ▪ 3'b101: SRP-Capable Host ▪ 3'b110: Non-OTG Host ▪ Others: Reserved This field is configured using parameter "Name: OTG_MODE".		

30.5.3.14 User HW Config3 Register (GHWCFG3)

✧ Offset: 04Ch

This register contains the configuration options selected.

Table 30-22 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		RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1- ADP logic is present along with HS OTG controller.		
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 This field is configured using parameter "Name: OTG_SYNC_RESET_TYPE".	Configurable	RO
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)	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 4'b0000: 11 bits ▪ 4'b0001: 12 bits ... ▪ 4'b1000: 19 bits ▪ Others: Reserved This field is configured using parameter "Name: OTG_TRANS_COUNT_WIDTH".		

30.5.3.15 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 30-23 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	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This field is configured using parameter "Name: OTG_EN_SESSIONEND_FILTER"		
23	"b_valid" Filter Enabled (BValidFltr) ▪ 1'b0: No filter ▪ 1'b1: Filter This field is configured using parameter "Name: OTG_EN_B_VALID_FILTER".	Configurable	RO
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	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 1'b1: Yes This field is configured using parameter "Name: OTG_MIN_AHB_FREQ_LESSTHAN_60".		
4	Enable Partial Power Down ▪ 1'b0: Partial Power Down Not Enabled ▪ 1'b1: Partial Power Down Enabled This field is configured using parameter "Name: OTG_EN_PWROPT".	Configurable	RO
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

30.5.3.16 DFIFO Software Config Register (GDFIFOCFG)

✧ Offset: 05Ch

Table 30-24 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

30.5.3.17 Host Periodic Transmit FIFO Size Register (HPTXFSIZ)

✧ Offset: 100h

This register holds the size and the memory start address of the Periodic TxFIFO.

Table 30-25 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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: 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)	Configurable	RO/R_W

30.5.3.18 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 30-26 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.3.19 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.

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 30-27 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

30.5.4 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:

30.5.4.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-28 Host Configuration Register: HCFG

Field	Description	Reset	Acces
31	Mode Change Ready Timer Enable (ModeChTimEn) 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.	1'b0	R_W
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)	8'd2	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.4.2 Host Frame Interval Register (HFIR)

✧ Offset: 404h

This register stores the frame interval information for the current speed to which the DWC_otg core has enumerated.

Table 30-29 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

30.5.4.3 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 30-30 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	16'h0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		
15:0	Frame Number (FrNum) This field increments when a new SOF is transmitted on the USB, and is reset to 0 when it reaches 16'h3FFF.	16'h3FFF	RO

30.5.4.4 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 30-31 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	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 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		

30.5.4.5 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 30-32 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

30.5.4.6 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 30-33 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

30.5.4.7 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

Table 30-34 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'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 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 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	1'b0	R_W_SS_SC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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

30.5.4.7.1 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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.

30.5.4.8 Host Channel-*n* Characteristics Register (HCCHARn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 500h + (Channel_number * 20h)

Table 30-35 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	2'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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'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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.4.9 Host Channel-n Split Control Register (HCSPLTn)

✧ Channel_number: 0 ≤n ≤15

✧ Offset: 504h + (Channel_number * 20h)

Table 30-36 Host Channel-n Split Control Register: HCSPLTn

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

30.5.4.10 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-37 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. The core sets this bit when the corresponding channel's descriptor list rolls over. For non Scatter/Gather DMA mode, this bit is reserved.	1'b0	R_SS_WC
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	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	masked in the core.		
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 masked in the core.	1'b0	R_SS_WC
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

30.5.4.11 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 30-38 Host Channel-n Interrupt Mask Register: HCINTMSKn

Field	Description	Reset	Acces
31:11	Reserved	1'b0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
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

30.5.4.12 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 30-39 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.	2'b00	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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)		
28:19	Reserved	1'b0	RO
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.	8'h0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Note that this field is applicable only for periodic (Isochronous and Interrupt) channels.		
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. **Note:** Do not set this bit for IN transfers. If this bit is set for IN transfers it disables the	1'b0	R_W
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

30.5.4.13 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 30-40 Host Channel-n DMA Address Register: HCDMAn

Field	Description	Reset	Acces
Buffer DMA Mode			
31:0	DMA Address (DMAAddr)	"X" if not	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

<table>
<tr><td></td><td>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.</td><td>program med as the register is in SPRAM</td><td></td></tr>
<tr><td colspan="4">Descriptor DMA Mode</td></tr>
<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.</td><td>6'h0</td><td>R/W</td></tr>
</table>

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Isochronous: CTD for isochronous is based on the current frame/µframe value. Need to be set to zero by application.		
2:0	Reserved	3'h0	RO

30.5.4.14 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 30-41 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

30.5.4.15 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 30-42 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

30.5.5 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The registers and register fields described in this section can pertain to IN or OUT endpoints, or both, or specific endpoint types as noted.

30.5.5.1 Device Configuration Register (DCFG)

✧ Offset: 800h

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 30-43 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	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=0 => Buffered DMA mode ▪ GAHBCFG.DMAEn=1,DCFG.DescDMA=1 => Scatter/Gather DMA mode		
22:18	IN Endpoint Mismatch Count (EPMisCnt) This field is valid only in shared FIFO operation. 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.	5'h8	R_W
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	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 (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

30.5.5.2 Device Control Register (DCTL)

✧ Offset: 804h

Table 30-44 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.	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		
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) 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.	1'b0	R_W
14:13	Global Multi Count (GMC)	2'h0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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'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'h0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 3'b011: Test_SE0_NAK mode ▪ 3'b100: Test_Packet mode ▪ 3'b101: Test_Force_Enable Others: Reserved		
3	Global OUT NAK Status (GOUTNakSts) ▪ 1'b0: A handshake is sent based on the FIFO Status and 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.	1'b0	WO
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 1-53. ▪ 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	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		

Following table lists the minimum duration under various conditions for which the Soft Disconnect (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 30-45 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

30.5.5.3 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 30-46 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	1'b0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	(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 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

30.5.5.4 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 30-47 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)	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	This bit is valid only when Device Descriptor DMA is enabled.		
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 (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

30.5.5.5 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 30-48 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)	1'b0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	Applies to control endpoints only.		
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

30.5.5.6 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 30-49 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

30.5.5.7 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 30-50 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	16'h0	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	The value of this field depends on the number of OUT endpoints that are configured.		
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 that are configured.	16'h0	R_W

30.5.5.8 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 30-51 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

30.5.5.9 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 30-52 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:	32'h0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 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		

30.5.5.10 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 30-53 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

30.5.5.11 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 30-54 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

30.5.5.12 Device VBUS Discharge Time Register (DVBUSDIS)

✧ Offset: 0828h

This register specifies the VBUS discharge time after VBUS pulsing during SRP.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 30-55 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: 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.	30 MHz: 16'h0B8F 60 MHz: 16'h17D7	R_W

30.5.5.13 Device VBUS Pulsing Time Register (DVBUSPULSE)

✧ Offset: 082Ch

This register specifies the VBUS pulsing time during SRP.

Table 30-56 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

30.5.5.14 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 30-57 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)	9'h8	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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).		
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.5.15 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

✧ Unmask interrupt: 1'b1

Table 30-58 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

30.5.5.16 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 30-59 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.5.17 Device Each Endpoint Interrupt Register Mask (DEACHINTMSK)

✧ Offset: 83Ch

Dependency: This register is available only when parameter "Name: OTG_MULTI_PROC_INTRPT".

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 30-60 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

30.5.5.18 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 30-61 Device Each In Endpoint-n Interrupt Register: DIEPEACHMSKn

Field	Description	Reset	Acces
31:14	Reserved		RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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
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

30.5.5.19 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 30-62 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

30.5.5.20 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 30-63 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)	1'b0	WO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	A write to this bit clears the NAK bit for the endpoint.		
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 assigned to IN Endpoint 0.	4'h0	R_W
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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

30.5.5.21 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.

Table 30-64 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

19:18	Endpoint Type (EPType) Hardcoded to 2'b00 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 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.	1'b0	RO
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

30.5.5.22 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 30-65 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.	1'b0	R_WS_SC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ 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. – 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	1'b0	WO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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	1'h0	R_W R_WS_SC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		
20	Snoop Mode (Snp) 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.	1'b0	R_W
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)	1'b0	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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 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

30.5.5.23 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Interrupt register. The application must clear the appropriate bit in this register to clear the corresponding bits in the DAINT and GINTSTS registers.

Table 30-66 Device Endpoint-n Interrupt Register: DIEPINTn/DOEPINTn

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	1'b0	R_SS_WC R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	▪ Thresholding is enabled.		
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 the TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl)).	1'b1	RO
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.	1'b0	R_SS_WC

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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.		
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

30.5.5.24 Device Endpoint 0 Transfer Size Register (DIEPTSIZ0/DOEPTSIZ0)

✧ Offset for IN endpoints: 910h

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

✧ 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.

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 30-67 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 30-68 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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

30.5.5.25 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 30-69 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 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). 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	2'b0	R_W RO RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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
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

30.5.5.26 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 30-70 Device Endpoint-n DMA Address Register: DIEPDMAn/DOEPDMAn

Field	Description	Reset	Acces
31:0	DMA Address (DMAAddr)	"X" if not	R_W

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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. ▪ 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.	programmed as the register is in SPRAM	

30.5.5.27 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 30-71 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

30.5.5.28 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 30-72 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.	Configurable	RO

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

	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		

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Section 7

BOOT

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

31 XBurst Boot ROM Specification

The JZ4775 contains an internal 16KB boot ROM. The CPU boots from the boot ROM after reset.

31.1 Boot Select

The boot sequence of the JZ4775 is controlled by boot_sel [2:0]. The configuration is shown as follow:

Table 31-1 Boot Configuration of JZ4775

boot_sel[2:0]	Boot method
111	USB boot @ USB 2.0 device, EXTCLK=24MHz
110	NAND boot @ CS1
101	SD boot @ MSC0 (MMC/SD use GPIO Port A)
100	SD boot @ MSC1 (MMC/SD use GPIO Port E)
011	eMMC boot @ MSC0 (use GPIO Port A)
010	NOR boot @ CS4 (just for FPGA testing)
001	USB boot @ USB 2.0 device, EXTCLK=26MHz
000	SPI boot @ SPI0/CE0

31.2 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, 6 flags 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 14KB code are read out from NAND to tcsm, if the 14KB reading failed, the next 14KB backup in NAND will be read. Then branch to tcsm at 256 bytes offset.

3 There 14KB backup reading failed, the 14KB backup at 128th, 256th , …, 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 14KB code from MMC/SD card to tcsm and jump to it. Only one data bus which is MSC0_D0 is used. The clock EXTCLK/128 is used initially. When reading data, the clock EXTCLK/4 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 14KB code from eMMC boot partition1 to tcsm and jump to it. Only one data bus which is MSC0_D0 is used. The clock EXTCLK/4 is used.

6 If it is boot from USB, a block of code will be received through USB cable connected with host

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

PC and be stored in tcsm. Then branch to this area in tcsm.

7 If it is boot from MMC/SD card at MSC1, its function pins MSC1_D0,D1,D2,D3, MSC1_CLK, MSC1_CMD are initialized, the boot program loads the 14KB code from MMC/SD card to tcsm and jump to it.

NOTE: The JZ4775's tcsm is 16KB, its address is from 0xf4000000 to 0xf4004000.

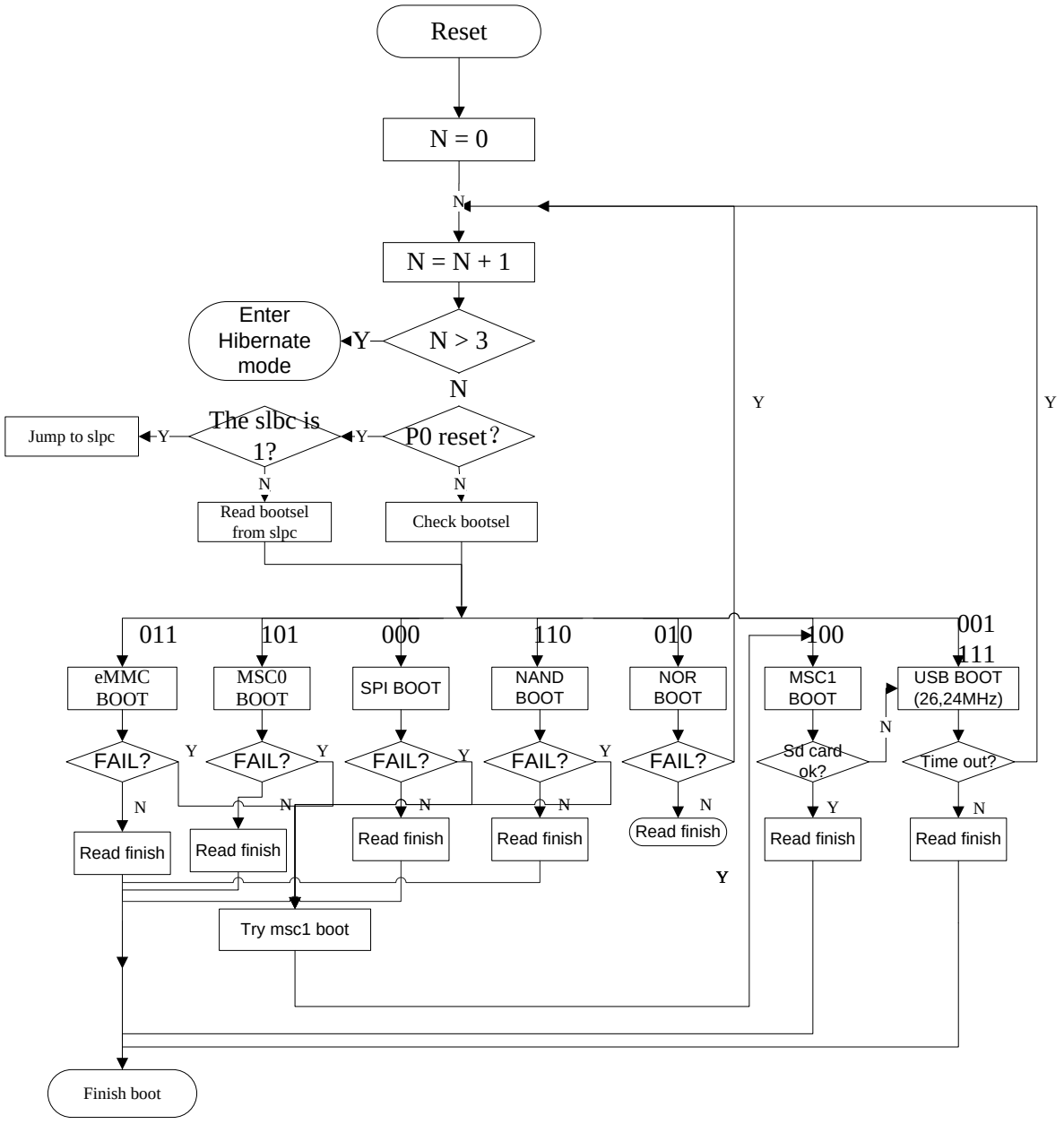


Figure 31-1 Boot sequence diagram of JZ4775

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.3 NAND Boot Specification

If CPU boots from NAND flash (CS1), the boot ROM will read 6 flags from NAND flash to know the NAND information including row cycle(2 or 3) and its page size(512, 2KB, 4KB, 8KB or 16KB bytes).

The content and definition of the 6 flags are shown as follow:

Table 31-2 The definition of 5 flags in NAND flash

<table>
<tr><th>Name</th><th>Location (in byte)</th><th>length (in byte)</th><th>Value</th><th>Description</th></tr>
<tr><td>BusWidth_flag</td><td>0 - 63</td><td>64</td><td>0x55
or
0xaa</td><td>Bus width
0x55: 8 bit NAND
0xaa: 16 bit NAND</td></tr>
<tr><td>NandType_flag</td><td>64 - 127</td><td>64</td><td>0x55
or
0xaa</td><td>NAND type.
0x55: common NAND
0xaa: toggle NAND</td></tr>
<tr><td>RowCycle_flag</td><td>128 - 159</td><td>32</td><td>0x55
or
0xaa</td><td>The number of row cycles.
0x55: 2-byte row cycles
0xaa: 3-byte row cycles</td></tr>
<tr><td>PageSize_flag2</td><td>160 - 191</td><td>32</td><td>0x55
or
0xaa</td><td rowspan="3">flag2 flag1 flag0 pagesize(byte).
0x55 0x55 0x55 512
0x55 0xaa 0x55 2048
0xaa 0x55 0x55 4096
0xaa 0xaa 0x55 8192
0xaa 0xaa 0xaa 16384</td></tr>
<tr><td>PageSize_flag1</td><td>192 - 223</td><td>32</td><td>0x55
or
0xaa</td></tr>
<tr><td>PageSize_flag0</td><td>224 - 255</td><td>32</td><td>0x55
or
0xaa</td></tr>
</table>

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 RowCycle_flag containing 32 bytes locates behind the NandType_flag, if the number of bytes of row cycles is 2, the flag should be filled with 0x55 for all 32 bytes, or else it should be filled with 0xaa;The PageSize_flag2, PageSize_flag1 and PageSize_flag0 each containing 32 bytes locate behind the RowCycle_flag, which value should be filled is determined by the page size of NAND. Please refer to table 2. Totally, 256 bytes are allocated for the 6 flags.

At first, the first 256 bytes (which is a PN* unit) in NAND containing 6 flags 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

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

nand.then, the RowCycle_flag will be obtained from the buffer to know the number of row cycles. At last, PageSize_flag2, PageSize_flag1 and PageSize_flag0 will be obtained from the buffer to know precise page size.

14KB codes in NAND will be loaded up to tcsm and branch to it at 256 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 14KB 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 is shown as following figure.
The procedure of the JZ4775 NAND boot is shown as Figure 31-3.

NOTE: PN is short for pseudorandom noise which is used for supporting TLC (three-level cell) NAND.

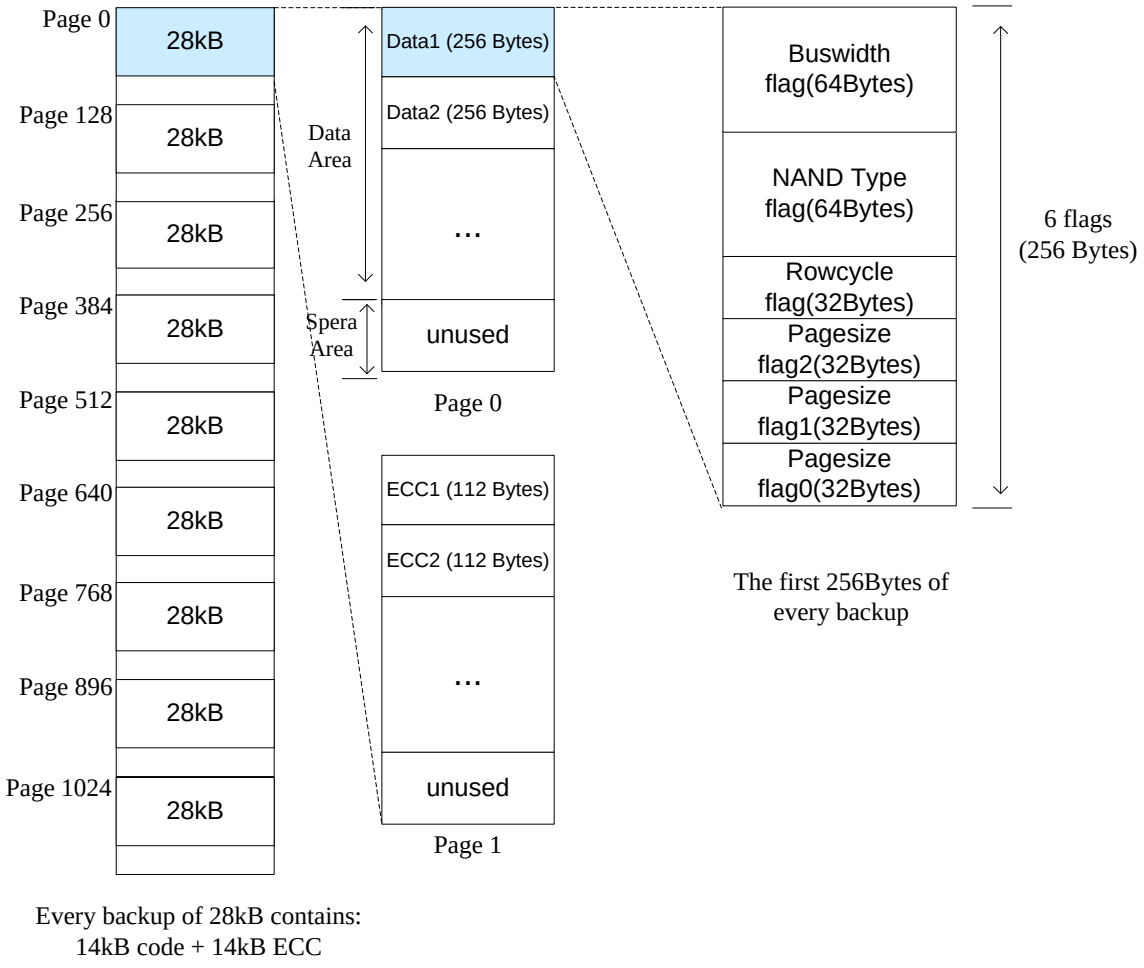
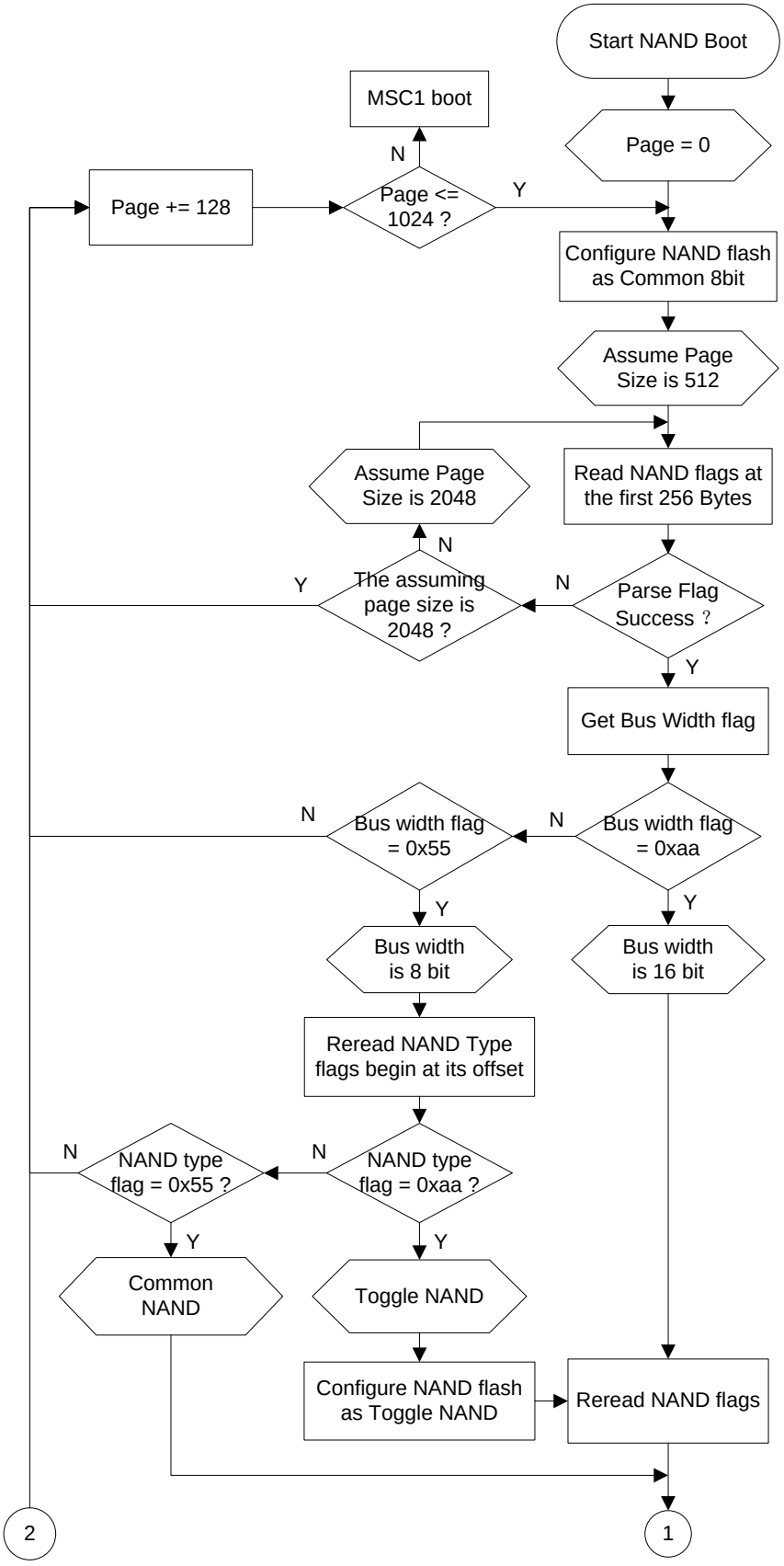


Figure 31-2 the distribution and structure of the boot code in NAND

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.



Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

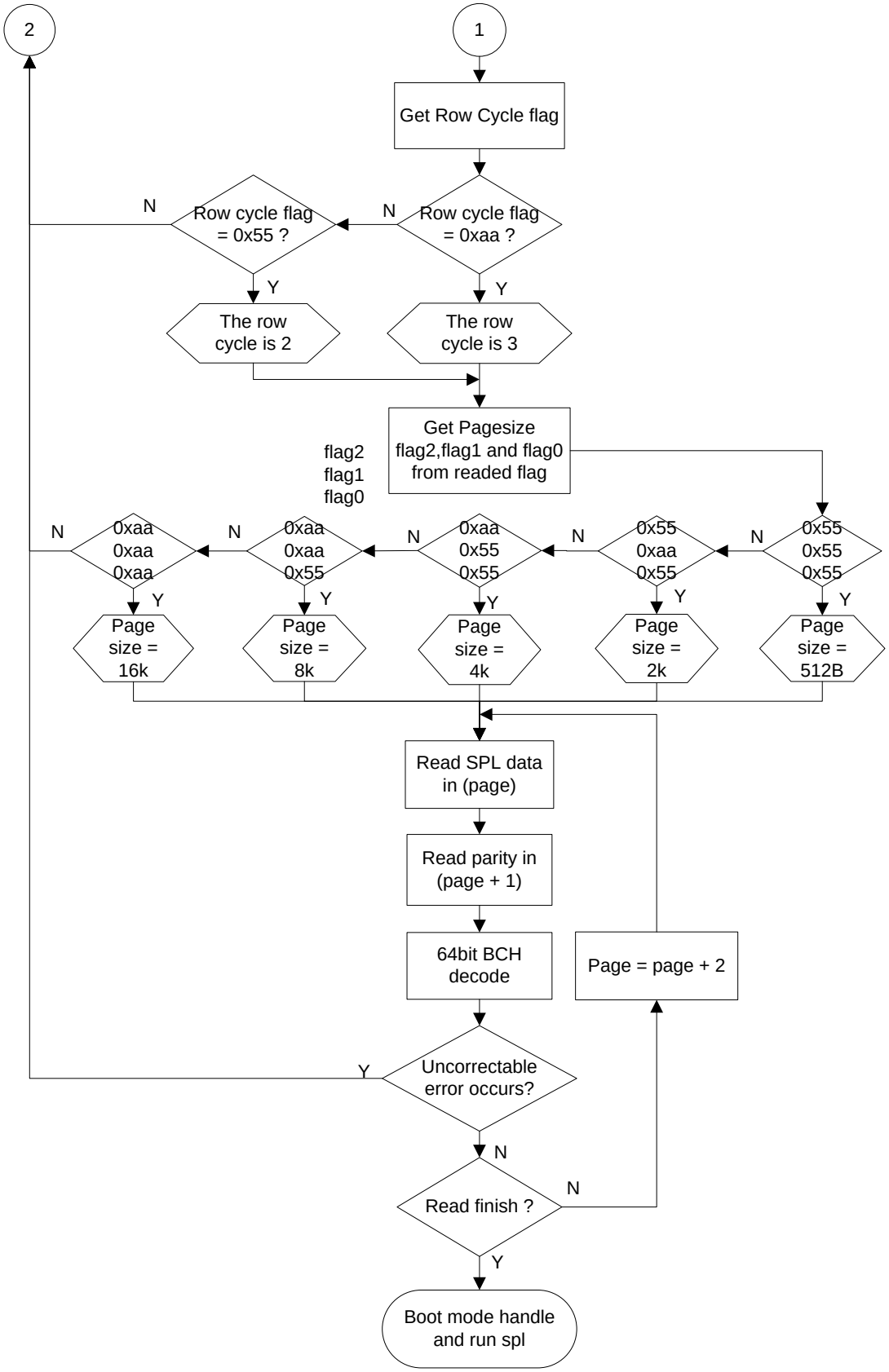


Figure 31-3 JZ4775 NAND Boot Procedure

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.4 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.

JZ4775 supports the external main crystal whose frequency is 24MHz.
The boot program supports both high-speed (480MHz) and full-speed (12MHz) transfer modes. The boot program uses the following two transfer types.

Table 31-3 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.

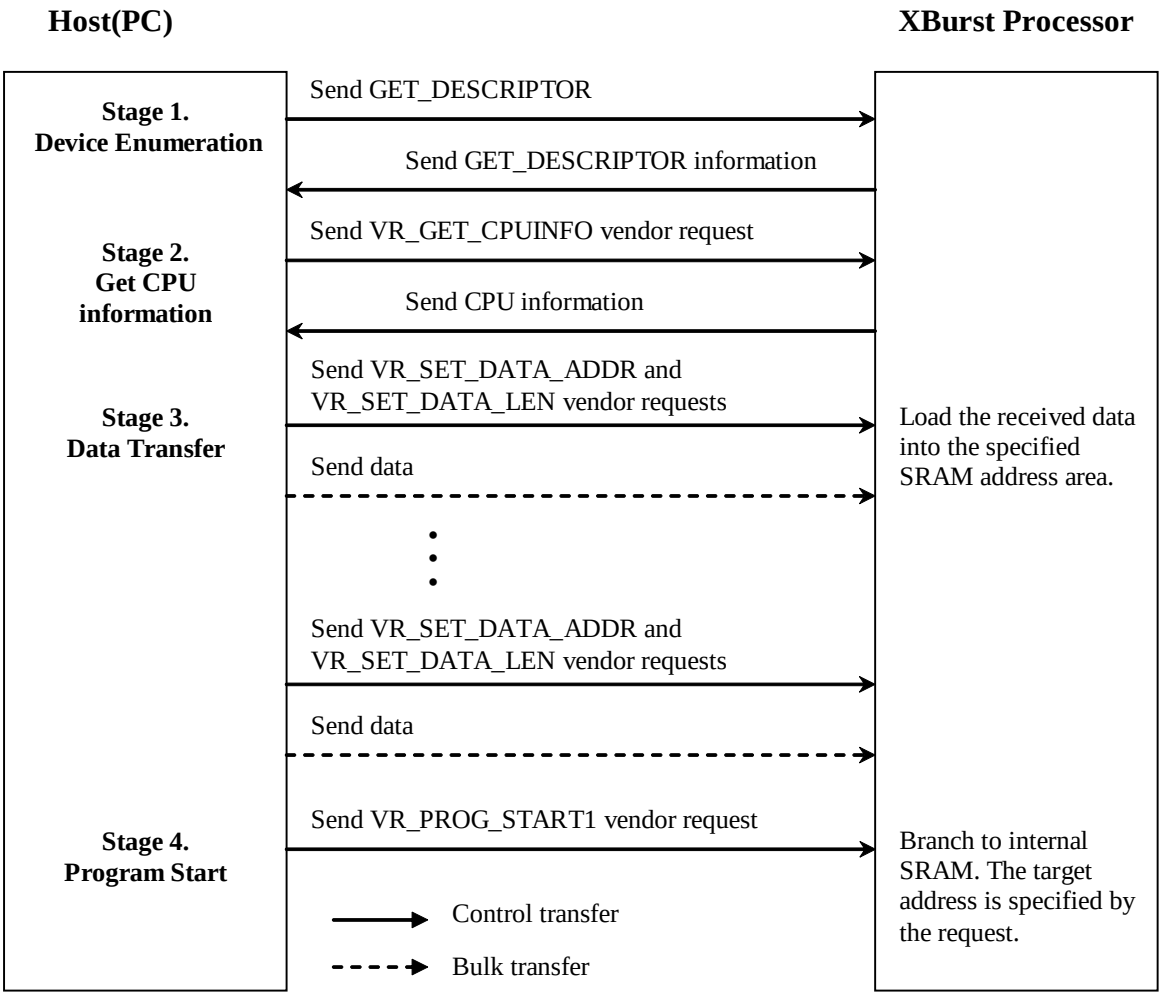


Figure 31-4 USB Communication Flow

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

The vendor ID and product ID for the USB boot device are 0xa108 and 0x4775 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 JZ4775 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 14KB 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 © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

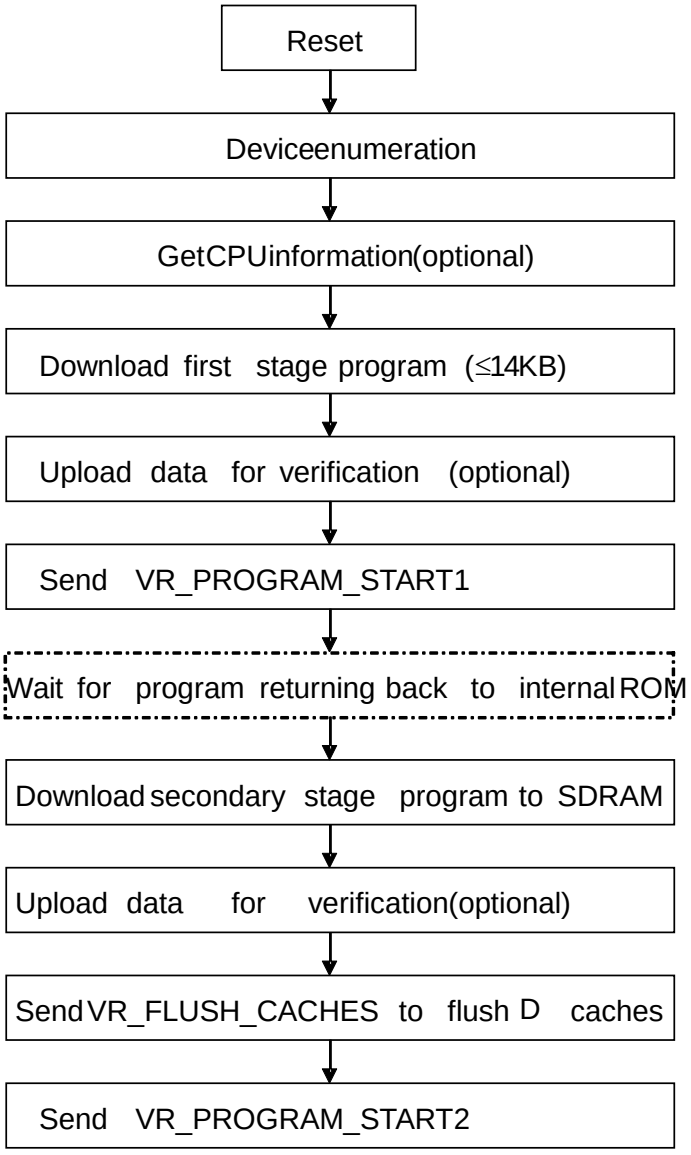


Figure 31-5 Typical Procedure of USB Boot

Following tables list all the vendor requests that USB boot program supports:

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

Table 31-4 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 31-5 Vendor Request 1 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	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 31-6 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 31-7 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.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 31-8 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.
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 31-9 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.

31.5 MSC0 Boot Specification

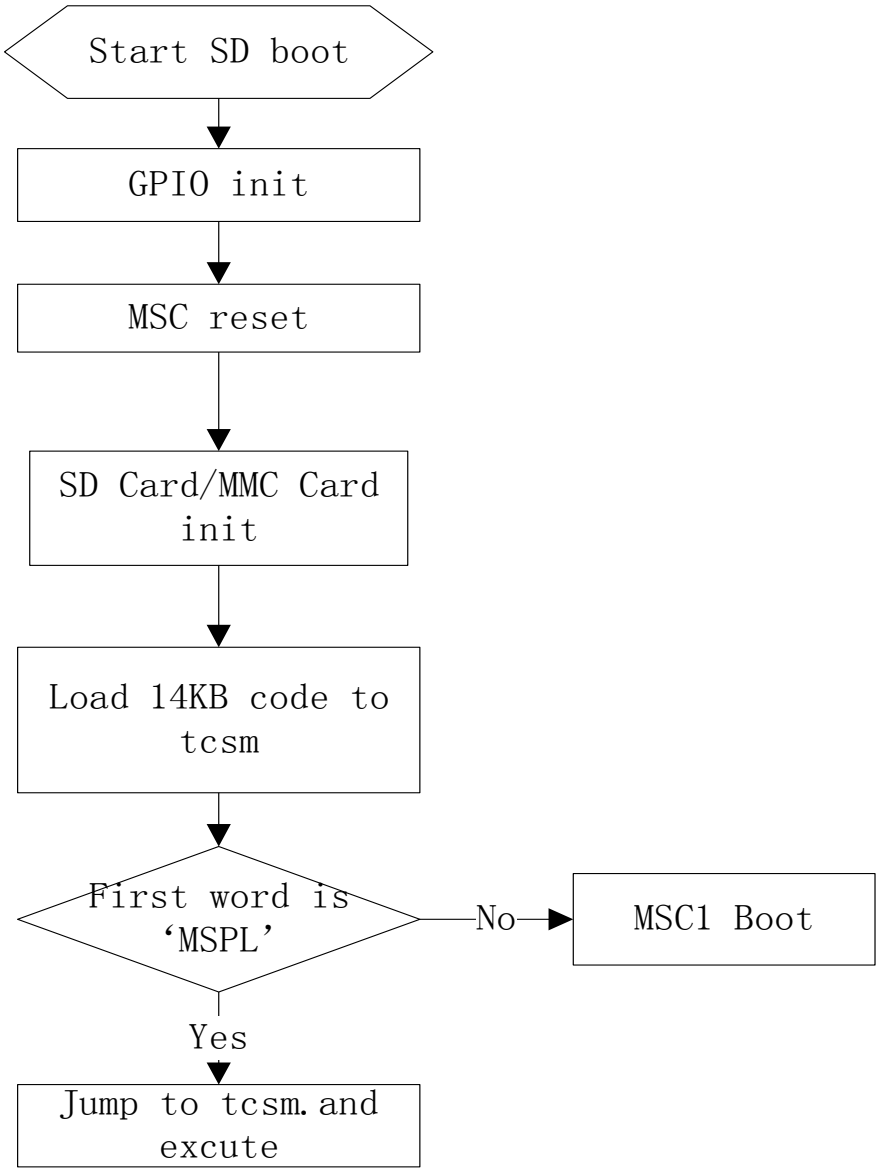
All cards can boot from MMC/SD Boot from MSC0, the boot program will load 14KB code starting at sector 0 from MMC/SD card to tcsm. First the boot program initializes MSC0_D0, MSC0_CLK, MSC0_CMD as function pins. Only one data pin MSC0_D0 is used. Then the boot program sends

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

CMD55 to test if it's SD or MMC card and initializes the card. At last it loads 14KB code from the card to tcsm and branches to execute the code in tcsm.

When initializing the card, the clock of EXTCLK/128 is used. And when reading data, the clock of EXTCLK/4 is used.

The procedure of the JZ4775 MMC/SD boot is shown as follow:



31.6 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 14KB code from eMMC boot partition1 area to tcsm. First the boot program initializes MSC0_D0, MSC0_CLK, MSC0_CMD as

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

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 14KB code from the card to tcsm and branches to execute the code in tcsm. and the clock of EXTCLK/4 is used.

The procedure of the JZ4775 eMMC boot is shown as follow:

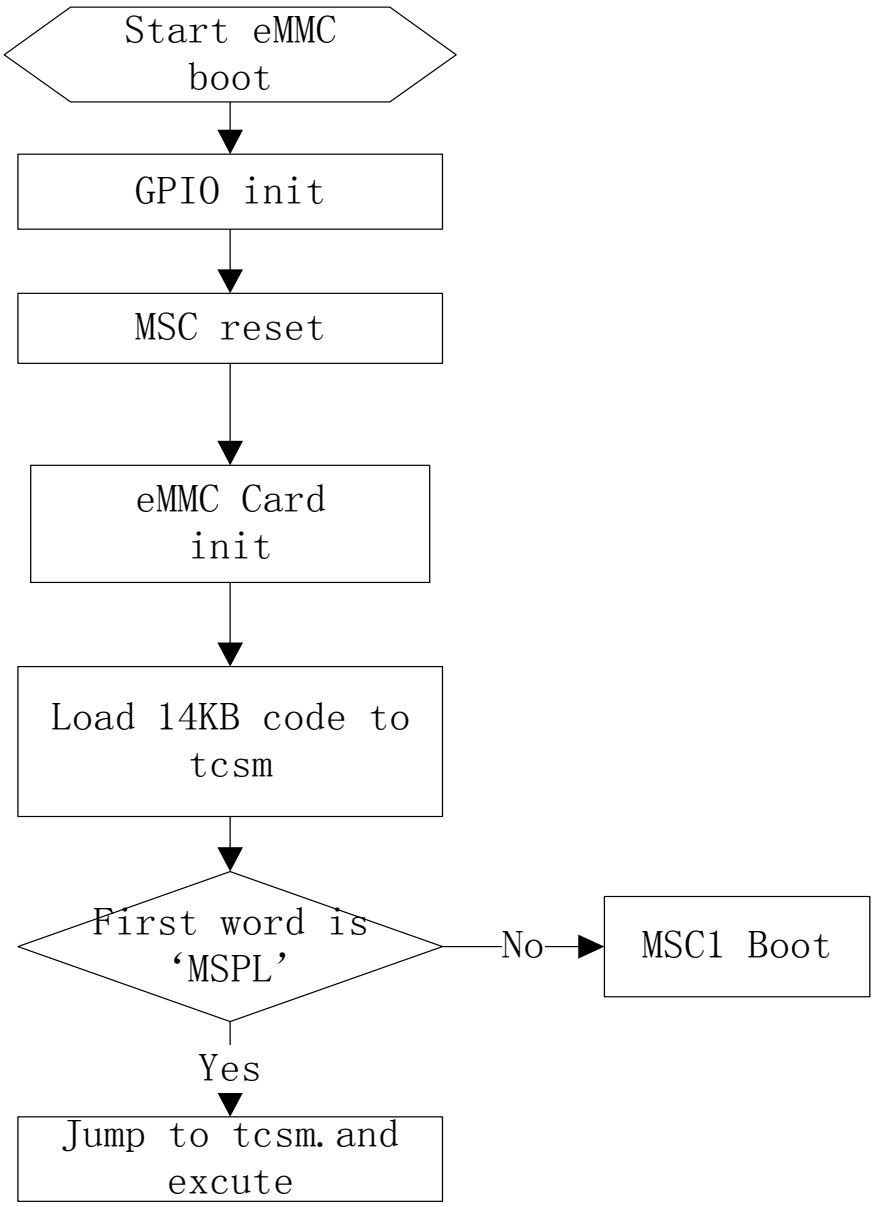


Figure 31-6 JZ4775 eMMC Boot Procedure

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

31.7 msc1 boot Specification

It is the same as msc0 boot. The purpose of msc1 boot is that,you can use it to burn other devices.

31.8 Spi boot Specification

When boot_sel[2:0] is selected as SPI boot(boot_sel = 000), CPU will boot from SPI nor flash by General-Purpose I/O Port-A 18, 20, 21, 23 pin. In spi nor flash address 0x0~0xF, this space will store 16 bytes that agreed SPI_boot flag. SPI_boot flag shown “SPI nor flash boot flag informations” table. SPI_boot detailed process shown “SPI Boot Procedure”

Note: Any irregularity in the above steps, SPI_boot will disable SSI controller and jump to MSC1 boot.

The SPI_boot flag information table and procedure of the JZ4775 SPI boot is shown as follow:

Table 31-10 SPI nor flash boot flag informations

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:

0x00~0x06 :Must store 0x04,0x03,0x02,0x55,0xAA,0x55,0xAA..

0x07 :The value of SSI Clock Generator Register.

0x08~0x0B :The length of copy data.

0x0C~0x0F :Copy data checksum.

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.

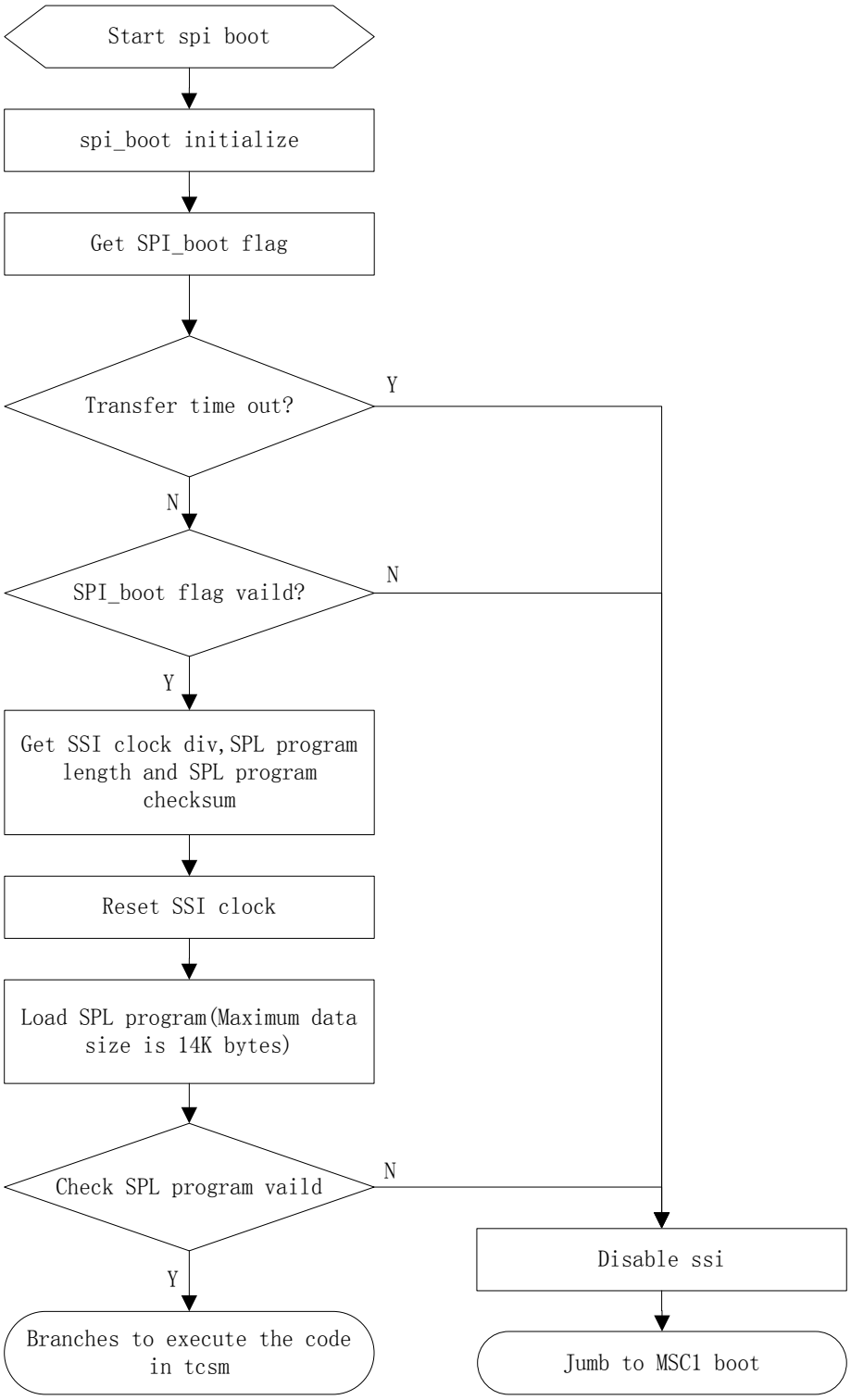


Figure 31-7 JZ4775 SPI Boot Procedure

Copyright © 2005-2015 Ingenic Semiconductor Co., Ltd. All rights reserved.