V8130L

Datasheet

Specifications are subject to change without notice.

© 2021Vango Technologies, Inc.
This document contains information that is proprietary to Vango Technologies, Inc.
Unauthorized reproduction of this information in whole or in part is strictly prohibited.

Vangotech

Declaration

THIS DOCUMENTATION IS THE ORIGINAL WORK AND COPYRIGHTED PROPERTY OF VANGO ("VANGO"). REPRODUCTION IN WHOLE OR IN PART MUST OBTAIN THE WRITTEN APPROVAL OF VANGO AND GIVE CLEAR ACKNOWLEDGEMENT TO THE COPYRIGHT OWNER.

THE PURCHASED PRODUCTS, SERVICES AND FEATURES ARE STIPULATED BY THE CONTRACT MADE BETWEEN VANGO AND THE CUSTOMER. PLEASE READ THE TERMS AND CONDITIONS OF THE CONTRACT AND RELEVANT INSTRUCTIONS CAREFULLY BEFORE USING, AND FOLLOW THE INSTRUCTIONS IN THIS DOCUMENTATION STRICTLY. VANGO ASSUMES NO RESPONSIBILITY FOR THE CONSEQUENCES OF IMPROPER USE(INCLUDING BUT NOT LIMITED TO OVERVOLTAGE, OVERCLOCK, OR EXCESSIVE TEMPERATURE).

THE INFORMATION FURNISHED BY VANGO IS PROVIDED JUST AS A REFERENCE OR TYPICAL APPLICATIONS, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. VANGO RESERVES THE RIGHT TO MAKE CHANGES IN CIRCUIT DESIGN AND/OR SPECIFICATIONS AT ANY TIME WITHOUT NOTICE.

NOR FOR ANY INFRINGEMENTS OF PATENTS OR OTHER RIGHTS OF THE THIRD PARTIES WHICH MAY RESULT FROM ITS USE. NO LICENSE IS GRANTED BY IMPLICATION OR OTHERWISE UNDER ANY PATENT OR PATENT RIGHTS OF VANGO. THIRD PARTY LICENCES MAY BE REQUIRED TO IMPLEMENT THE SOLUTION/PRODUCT. CUSTOMERS SHALL BE SOLELY RESPONSIBLE TO OBTAIN ALL APPROPRIATELY REQUIRED THIRD PARTY LICENCES. VANGO SHALL NOT BE LIABLE FOR ANY LICENCE FEE OR ROYALTY DUE IN RESPECT OF ANY REQUIRED THIRD PARTY LICENCE. VANGO SHALL HAVE NO WARRANTY, INDEMNITY OR OTHER OBLIGATIONS WITH RESPECT TO MATTERS COVERED UNDER ANY REQUIRED THIRD PARTY LICENCE.

Vangotech

Revision History

Revision	Date	Description
1.0	Oct.15,2020	Initial Version
1.1	Dec.15,2020	Modify watchdog number; Modify DLL control register description; Modify power supply dor DLL0 and PLL7; Modify Power-Up Sequence;
1.2	Mar.12,2021	Modify signal name in clock diagram;
1.3	Mar.24,2021	Modify the description of GMAC and USB2.0 OTG;
1.4	Aprl.21,2021	Add DDR, TRNG modules
1.5	Dec.21,2021	Modify USB DP/DM signal description; Modify Bit 7 description of SSP_SPI MUX SEL Register

Vangotech

Contents

TABLE OF CONTENTS

Declaration 2
Revision History 3
Contents 4
Figures 23
Tables 27
About This Documentation 46
1.1. Purpose 47
1.2. Intended Audience 47
1.3. Symbol Conventions 47
1.4. Acronyms and Abbreviations 47
Chapter 2. Overview 50
2.1. Processor Overview 51
2.2. Features 52
2.2.1. CPU Architecture 52
2.2.2. Memory 52
2.2.2.1. Boot ROM 52
2.2.2.2. Embedded SRAM 52
2.2.2.3. SDRAM 52
2.2.2.4. eMMC 52
2.2.3. System Peripherals 52
2.2.3.1. Timer 52
2.2.3.2. GIC 53
2.2.3.3. DMA 53
2.2.3.4. SCU 53
2.2.3.5. RTC 53
2.2.3.6. PWM 53
2.2.3.7. Thermal Sensor 53
2.2.3.8. Crypto Engine 53
2.2.3.9. eFuse 54
2.2.4. Display Subsystem 54
2.2.4.1. Video Output 54
2.2.5. External Peripherals 54
2.2.5.1. USB 54
2.2.5.2. MAC 54
2.2.5.3. SPI 54
2.2.5.4. UART 55
2.2.5.5. I2C 55
2.2.5.6. I2S 55
2.2.5.7. CAN 55
2.2.5.8. Keypad 56
2.2.6. Package 56
2.3. Block Diagram 57
Chapter 3. Pin Description 58
3.1. Pin Characteristics 59
3.2. GPIO Multiplex Function 64
3.3. PWM timer output signals 88
3.4. Signal Descriptions 88
Chapter 4. System 95
4.1. Memory Mapping 96
4.2. Clock 98
4.2.1. System Clocking 98
4.2.2. External Clock Sources 98
4.2.3. Internal Ring OSC 98
4.2.4. PLL Programming Values 98
4.2.5. Summary Clock information 99
4.2.6. Top Level Clocking Diagram 102
4.2.7. Special Clock Specification 105
4.2.7.1. UART Baud Rate Specification and Configuration of UART 106

vangotech

4.2.7.2. IrDA Data Rate Specification and Clock Configuration of UART 106
4.2.8. V8130L Core and IO Voltages 106
4.2.9. Clock Gating for Dynamic Power Reduction 106
4.2.9.1. Power Rail Scheme 106
4.2.9.2. Power Domains 107
4.3. Reset 108
4.3.1. Power-On Reset (Cold Reset) and Power Enable Operation 108
4.4. Global Programming View (GPV) 110
4.5. DDR 113
4.5.1. Overview 113
4.5.2. Features 113
4.5.3. Block Diagram 114
4.5.4. Connection between DDRx Controller and DDRx PHY 118
4.5.5. Register List 118
4.5.6. Register Description 121
4.5.6.1. Memory Controller Configuration Register 121
4.5.6.2. Memory Controller State Control Register 126
4.5.6.3. Mode Register Set Value Register of MR and EMR (MRSVR0, Offset = 0x08) 129
4.5.6.4. Mode Register Set Value Register of EMR 2 and EMR 3 (MRSVR1, Offset = 0x0C) 130
4.5.6.5. External Rank0/Rank1 Register (EXRANKR, Offset = 0x10) 131
4.5.6.6. Timing Parameter 0 Register (TMPR0, Offset = 0x14) 148
4.5.6.7. Timing Parameter 1 Register (TMPR1, Offset = 0x18) 149
4.5.6.8. Timing Parameter 2 Register (TMPR2, Offset = 0x1C) 152
4.5.6.9. DDRx PHY Command and Data Block Control Register (PHYCR0, Offset = 0x20) 155
4.5.6.10. DDRx PHY Read Path DLL Delay Tuning Register (PHYRDTR, Offset = 0x24) 160
4.5.6.11. COMPBLK Control Register (COMPBLKCR, Offset = 0x28) 161
4.5.6.12. Automatic Power-down/Self-refresh Control Register (APDCR, Offset = 0x2C) 162
4.5.6.13. Channel Arbitration Setup Register A (CHARBRA, Offset = 0x30) 163
4.5.6.14. Channel Arbiter Grant Count Register - A (CHGNTRA, Offset = 0x34) 166
4.5.6.15. Channel Arbiter Grant Count Register - B (CHGNTRB, Offset = 0x38) 168
4.5.6.16. DDRx PHY Write/Read Data Timing Control Register (PHYWRTMR, Offset = 0x3C) 169
4.5.6.17. Command Flush Control Register (FLUSHCR, Offset = 0x40) 173
4.5.6.18. Command Flush Status Register (FLUSHSR, Offset = 0x44) 174
4.5.6.19. AHB SPLIT Control Register (SPLITCR, Offset = 0x48) 175
4.5.6.20. DDRx PHY and DDR3 Update Control Register (UPDCR, Offset = 0x4C) 177
4.5.6.21. Revision Register (REVR, Offset = 0x50) 178
4.5.6.22. Feature Register 1 (FEATR1, Offset = 0x54) 178
4.5.6.23. Feature Register 2 (FEATR2, Offset = 0x58) 180
4.5.6.24. User Defined Register (UDEFR, Offset = 0x5C) 181
4.5.6.25. Write-leveling Control Register (WLEVELCR, Offset = 0x60) (Only for DDR3 Combo PHY) 181
4.5.6.26. Write-leveling Byte 7 ~ Byte 4 Control Register (WLEVELBHR, Offset = 0x64) (Only for DDR3 Combo PHY) 183
4.5.6.27. Write-leveling Byte 3 ~ Byte 0 Control Register (WLEVELBLR, Offset = 0x68) (Only for DDR3 Combo PHY) 184
4.5.6.28. DDR3 PHY MISC Control Register 1 (PHYMISCR1, Offset = 0x6C) 185
4.5.6.29. Read-leveling Control Register (RLEVELCR, Offset = 0x70) (Only for DDR3 Combo PHY) 187
4.5.6.30. msdly Byte Control Register (MSDLYCR, Offset = 0x74) 189
4.5.6.31. wrdllsel Control Register (WRDLLCR, Offset = 0x78) 190
4.5.6.32. Traffic Monitor Clock Cycle Register (TRAFMR, Offset = 0x7C) 191
4.5.6.33. Command Count Register for Channel 0 (CMDCNTR0, Offset = 0x80) 191
4.5.6.34. Command Count Register for Channel 1 (CMDCNTR1, Offset = 0x84) 191
4.5.6.35. Command Count Register for Channel 2 (CMDCNTR2, Offset = 0x88) 192
4.5.6.36. Command Count Register for Channel 3 (CMDCNTR3, Offset = 0x8C) 192
4.5.6.37. Command Count Register for Channel 4 (CMDCNTR4, Offset = 0x90) 192
4.5.6.38. Command Count Register for Channel 5 (CMDCNTR5, Offset = 0x94) 192
4.5.6.39. Command Count Register for Channel 6 (CMDCNTR6, Offset = 0x98) 193
4.5.6.40. Command Count Register for Channel 7 (CMDCNTR7, Offset = 0x9C) 193
4.5.6.41. AHB INCR Read Prefetch Length 1 (AHBRPRER1, Offset = 0xA0) 193
4.5.6.42. AHB INCR Read Prefetch Length 2 (AHBRPRER2, Offset = 0xA4) 195
4.5.6.43. Initialization of Waiting Cycle Count 1 (INITWCR1, Offset = 0xA8) 196
4.5.6.44. Initialization of Waiting Cycle Count 2 (INITWCR2, Offset = 0xAC) 197
4.5.6.45. QoS Control Register (QOSCR, Offset = 0xB0) 197
4.5.6.46. QoS Command Count Register A (QOSCNTRA, Offset = 0xB4) 198

Vango Technologies

4.5.6.47. QoS Command Count Register B (QOSCNTRB, Offset = 0xB8) 199
4.5.6.48. QoS Command Count Register C (QOSCNTRC, Offset = 0xBC) 199
4.5.6.49. QoS Command Count Register D (QOSCNTRD, Offset = 0xC0) 200
4.5.6.50. Channel Arbitration Setup Register B (CHARBRB, Offset = 0xC4) 201
4.5.6.51. Channel Arbiter Grant Count Register - C (CHGNTRC, Offset = 0xC8) 204
4.5.6.52. Channel Arbiter Grant Count Register - D (CHGNTRD, Offset = 0xCC) 205
4.5.6.53. Debug Address (DBGADR, Offset = 0xD0) 206
4.5.6.54. Debug Address Mask (DBGADMR, Offset = 0xD4) 206
4.5.6.55. Debug Write Data (DBGWDR, Offset = 0xD8) 206
4.5.6.56. Debug Write Data Mask (DBGWDMR, Offset = 0xDC) 207
4.5.6.57. Debug Master Control (DBGMSTCR, Offset = 0xE0) 207
4.5.6.58. Debug Access Control (DBGACCR, Offset = 0xE4) 207
4.5.6.59. Debug Policy Control (DBGPCR, Offset = 0xE8) 208
4.5.6.60. Debug Control (DBGENR, Offset = 0xEC) 208
4.5.6.61. Debug Address Status (DBGADSR, Offset = 0xF0) 209
4.5.6.62. Debug Write Data Status (DBGWDSR, Offset = 0xF4) 209
4.5.6.63. Debug Master Status (DBGMSTSR, Offset = 0xF8) 209
4.5.6.64. Debug Access Status (DBGACCSR, Offset = 0xFC) 209
4.5.6.65. LPDDR2 Mode Register MR1/MR2/MR3 (LP2MRA, Offset = 0x100) 209
4.5.6.66. LPDDR2 Mode Register MR9/MR10 (LP2MRB, Offset = 0x104) 210
4.5.6.67. LPDDR2 Mode Register MR16/MR17 (LP2MRC, Offset = 0x108) 210
4.5.6.68. LPDDR2 Mode Register MR41/MR42/MR48 (LP2MRD, Offset = 0x10C) 210
4.5.6.69. LPDDR2 Mode Register MR63 (LP2MRE, Offset = 0x110) 210
4.5.6.70. LPDDR2 MRW and MRR Command Register (LP2MRCR, Offset = 0x114) 211
4.5.6.71. LPDDR2 MRR Command Read Value Register (LP2MRVR, Offset = 0x118) 212
4.5.6.72. LPDDR2 Added Latency Register (LP2ADLR, Offset = 0x11C) 212
4.5.6.73. LPDDR2 1-µs Wait Cycle Register (LP2WCR1, Offset = 0x120) 213
4.5.6.74. LPDDR2 10-µs Wait Cycle Register (LP2WCR2, Offset = 0x124) 213
4.5.6.75. Reserved Register (Offset = 0x128) 213
4.5.6.76. Re-arbitration Disable Register (Offset = 0x12C) 213
4.5.6.77. DDR PHY Read Path DLL Delay Tuning for Falling Edge Register (PHYRDTFR, Offset = 0x130) 216
4.5.6.78. DDR PHY MISC2 Control Register (PHYMISCR2, Offset = 0x134) 216
4.5.6.79. DDR ELASTIC FIFO Control Register (EFIFOCR, Offset = 0x138) 219
4.5.6.80. Byte n Read Deskew Register (RBnDSKW, Offset = 0x160+4*n) 220
4.5.6.81. Byte n Write Deskew Register (WBnDSKW, Offset = 0x180+4*n) 220
4.5.6.82. Write Data Mask Deskew Register (WDMDSKW, Offset = 0x1A0) 221
4.5.6.83. Byte 8 Read Deskew Register (RB8DSKW, Offset = 0x1A4) 222
4.5.6.84. Byte 8 Write Deskew Register (WB8DSKW, Offset = 0x1A8) 222
4.5.6.85. Byte 8 PHY Control Register (B8_PHYCR, Offset = 0x1AC) 223
4.5.6.86. BIST Control Register (BIST_CTRL, Offset = 0x1B0) 225
4.5.6.87. BIST Start Address Register (BIST_START_ADDR, Offset = 0x1B4) 226
4.5.6.88. BIST Test Size Register (BIST_SIZE, Offset = 0x1B8) 226
4.5.6.89. Reserved Register (Reserved, Offset = 0x1BC) 227
4.5.6.90. BIST Result Register (BIST_RESULT, Offset = 0x1C0) 227
4.5.6.91. BIST Fail Address Register (BIST_FAIL_ADDR, Offset = 0x1C4) 227
4.5.6.92. BIST Fail Data Low Register (BIST_FAIL_DATA_LOW, Offset = 0x1C8) 228
4.5.6.93. BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x1CC) 228
4.5.6.94. BIST Expect Data Low Register (BIST_EXP_DATA_LOW, Offset = 0x1D0) 228
4.5.6.95. BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x1D4) 229
4.5.6.96. Reserved Register (Reserved, Offset = 0x1D8) 229
4.5.6.97. Reserved Register (Reserved, Offset = 0x1DC) 229
4.5.6.98. ECC Control Register (ECC_CTRLR, Offset = 0x1E0) 229
4.5.6.99. ECC Interrupt Control Register (ECC_INTCR, Offset = 0x1E4) 230
4.5.6.100. ECC Error Channel Register (ECC_CHR, Offset = 0x1E8) 231
4.5.6.101. ECC Error Address Register (ECC_EAR, Offset = 0x1EC) 231
4.5.6.102. ECC Error Bit Position Low Byte Register (ECC_EBPLR, Offset = 0x1F0) 232
4.5.6.103. ECC Error Bit Position High Byte Register (ECC_EBPHR, Offset = 0x1F4) 232
4.5.6.104. ECC One-bit error Count Register (ECC_ERR1_COUNT, Offset = 0x1F8) 232
4.5.6.105. ECC Two-bit Error Count Register (ECC_ERR2_COUNT, Offset = 0x1FC) 233
4.5.7. Memory Address Table (MA Table) 233
4.5.7.1. AHB/AXI Address to Memory Address Mapping 234
4.5.7.2. MA Table1 235

Vango Tech

4.5.7.3. MA Table2 236
4.5.7.4. MA Table3 238
4.5.7.5. MA Table4 239
4.6. SCU 241
4.6.1. Overview 241
4.6.2. Features 242
4.6.3. Operations and Functional Descriptions 242
4.6.3.1. Block Diagram 242
4.6.3.2. Real-Time Clock (RTC) Module 243
4.6.3.3. Power Control Module (PCM) 243
4.6.3.4. Clock Control Module (CCM) 244
4.6.3.5. Power Domain Partition 244
4.6.3.6. SCU Boot Flow 245
4.6.4. Register List 246
4.6.5. Register Description 247
4.6.5.1. Boot-up Status Register 247
4.6.5.2. Boot-up Control Register 248
4.6.5.3. Power Control Register 249
4.6.5.4. Power Start-up Sequence Control Register 250
4.6.5.5. Chip ID Register 250
4.6.5.6. System Control Unit Version Register 251
4.6.5.7. Strap Value Register 251
4.6.5.8. OSC Control Register 251
4.6.5.9. Power Mode Register 252
4.6.5.10. Interrupt Status Register 253
4.6.5.11. Interrupt Enable Register 254
4.6.5.12. Software Reset Control Register 255
4.6.5.13. PLL0 Control Register 255
4.6.5.14. PLL4 Control Register 256
4.6.5.15. DLL Control Register 256
4.6.5.16. AHB Clock Control Register 256
4.6.5.17. AHB Clock Sleep Control Register 257
4.6.5.18. APB Clock Control Register 257
4.6.5.19. APB Clock Sleep Control Register 258
4.6.5.20. AXI Clock Control Register 260
4.6.5.21. AXI Clock Sleep Control Register 260
4.6.5.22. Multi-Core WFI/WFE Control Register 261
4.6.5.23. Wakeup Event Status Register 261
4.6.5.24. Wakeup Event Status Register 262
4.6.5.25. IDLE Control Register 262
4.6.5.26. Real-Time Clock (RTC) Function 263
4.6.5.27. RTC Time Register 263
4.6.5.28. RTC Alarm Time Register 264
4.6.5.29. RTC Control Register 266
4.6.5.30. RTC Tick Trim Control Register 267
4.6.5.31. RTC Time BCD Register 268
4.6.5.32. RTC Alarm BCD Register 269
4.6.6. Extension Register Description 271
4.6.6.1. CA7 Configuration 1 Register 278
4.6.6.2. CA7 Configuration 2 Register 278
4.6.6.3. CA7 Configuration 3 Register 278
4.6.6.4. CA7 Configuration 4 Register 278
4.6.6.5. CA7 Configuration 5 Register 279
4.6.6.6. CA7 Configuration 6 Register 279
4.6.6.7. CA7's SPI Host Magic Number Register 279
4.6.6.8. PLL Lock Status Register 279
4.6.6.9. DDR CTRL 0 Register 280
4.6.6.10. DDR CTRL 1 Register 281
4.6.6.11. DDR CTRL 2 Register 282
4.6.6.12. SBS_OTG CTRL Register 282
4.6.6.13. SBS_OTG_1 CTRL Register 284
4.6.6.14. PLL1 Control Register 286
4.6.6.15. PLL1 Lock Timer Register 287

Vangotech

4.6.6.16. PLL2 Control Register 287
4.6.6.17. PLL2 Lock Timer Register 287
4.6.6.18. PLL3 Control Register 287
4.6.6.19. PLL3 Lock Timer Register 288
4.6.6.20. PLL5 Control Register 288
4.6.6.21. PLL5 Lock Timer Register 288
4.6.6.22. DLL0 Control Register 288
4.6.6.23. DLL1 Control Register 289
4.6.6.24. Clock Enable 0 Register 289
4.6.6.25. Clock Enable 1 Register 290
4.6.6.26. Clock Enable 2 Register 291
4.6.6.27. Clock Mux Selection Register 291
4.6.6.28. Clock Divider 1 Register 292
4.6.6.29. Clock Divider 2 Register 292
4.6.6.30. Clock Divider 3 Register 293
4.6.6.31. Clock Divider 4 Register 293
4.6.6.32. MISC. Register 293
4.6.6.33. Software Reset Mask 0 (swrst_msk0) Register 294
4.6.6.34. Software Reset Mask 1 (swrst_msk1) Register 295
4.6.6.35. Software Reset Mask 2 (swrst_msk2) Register 295
4.6.6.36. Software Reset Mask 3 (swrst_msk3) Register 296
4.6.6.37. Software Reset Register 297
4.6.6.38. PLL6 Control Register 297
4.6.6.39. PLL6 Lock Timer Register 297
4.6.6.40. Clock Divider 5 Register 297
4.6.6.41. Efuse Read Password Register 298
4.6.6.42. WDT Tickle Feature Register 298
4.6.6.43. EFUSE Feature Read Register 298
4.6.6.44. EFUSE Feature Read Register 298
4.6.6.45. SBS_OTG_2 CTRL Register 299
4.6.6.46. Clock Divider 6 Register 301
4.6.6.47. GPIO0 bps Enable Register 301
4.6.6.48. GPIO0 bps Data Register 301
4.6.6.49. GPIO1 bps Enable Register 301
4.6.6.50. GPIO1 bps Data Register 302
4.6.6.51. RGMII_gtx_clk_delay Register 302
4.6.6.52. RGMII_rx_clk_delay Register 302
4.6.6.53. TZC Configure Register 302
4.6.6.54. SPI020 Configure Register 303
4.6.6.55. SSP_SPI MUX SEL Register 303
4.6.6.56. PLL7 Control Register 303
4.6.6.57. PLL7 Lock Timer Register 304
4.6.6.58. Pin Mux with Mode X 304
4.6.6.59. IO Control Register 305
4.6.6.60. SYS_XTAL Control Register 306
4.7. System Boot 307
4.7.1. Overview 307
4.7.2. System Boot Diagram 307
4.7.3. Hardware Bootstrap Pin 307
4.8. WDT 309
4.8.1. Overview 309
4.8.2. Features 309
4.8.3. Block Diagram 309
4.8.4. Register List 309
4.8.5. Register Description 310
4.8.5.1. Key Register 310
4.8.5.2. Prescaler Register 310
4.8.5.3. Reload Register 311
4.8.5.4. Status Register 311
4.8.5.5. Interrupt Status Register 311
4.8.5.6. Control Register 312
4.8.5.7. Interrupter Length Register 313
4.8.5.8. Under Flow Value Register 313

Vangotech

4.8.5.9. WdRevision Register ... 313
4.9. PWMTMR ... 313
4.9.1. Overview ... 313
4.9.2. Features ... 314
4.9.3. Block Diagram ... 314
4.9.4. Operations and Functional Descriptions ... 314
4.9.4.1. Basic Timer Operation ... 314
4.9.4.2. Auto-reload and Double Buffering ... 315
4.9.4.3. PWM Function ... 316
4.9.4.4. Dead-zone Generation ... 316
4.9.4.5. DMA Request Mode ... 317
4.9.5. Register List ... 317
4.9.6. Register Description ... 318
4.9.6.1. Interrupt Status and Control Register ... 318
4.9.6.2. Timer Control Register ... 319
4.9.6.3. Timer Counting Value Buffer Register ... 320
4.9.6.4. Timer Counter Compare Buffer Register ... 320
4.9.6.5. Timer Count Observation Register ... 320
4.9.6.6. Revision Register ... 320
4.10. GIC ... 321
4.11. DMAC for AXI ... 325
4.11.1. Overview ... 325
4.11.2. Features ... 326
4.11.3. Block Diagram ... 326
4.11.4. Programming Guidelines ... 326
4.11.4.1. AXI Master and SFIFO ... 326
4.11.4.2. APB Slave ... 328
4.11.4.3. AXI Slave and Local Descriptor Memory ... 328
4.11.4.4. Channel Arbiter ... 329
4.11.4.5. Channel Controller ... 330
4.11.4.6. Link List Pointer ... 331
4.11.4.7. DMA Peripheral Interface ... 332
4.11.4.8. Example of One-dimension Transfer ... 332
4.11.4.9. Example of Two-dimension Transfer ... 333
4.11.4.10. Write-only Mode ... 335
4.11.4.11. Unalign Transfer Mode ... 335
4.11.4.12. Examples of Channel Control Settings ... 335
4.11.4.13. Usage of Event ... 336
4.11.5. Register List ... 336
4.11.6. Register Description ... 339
4.11.6.1. Interrupt Register ... 339
4.11.6.2. Terminal Count Interrupt Register ... 339
4.11.6.3. Terminal Count Interrupt Clear Register ... 339
4.11.6.4. Error/Abort Interrupt Register ... 339
4.11.6.5. Error/Abort Interrupt Clear Register ... 340
4.11.6.6. Terminal Count Status Register ... 340
4.11.6.7. Error/Abort Status Register ... 340
4.11.6.8. Channel Enable Register ... 340
4.11.6.9. Synchronization of Peripheral Interface Register ... 341
4.11.6.10. Local Descriptor Memory Base Register ... 341
4.11.6.11. Watchdog Timer Register ... 342
4.11.6.12. Global Event Register ... 342
4.11.6.13. PSlvErr Enable Register ... 342
4.11.6.14. Revision Number Register ... 343
4.11.6.15. Hardware Feature Register ... 343
4.11.6.16. Local Descriptor Memory Free Flag Register ... 344
4.11.6.17. Endian Conversion Register ... 345
4.11.6.18. Write-only Mode Constant Value Register ... 346
4.11.6.19. DMA Channel N Control Register ... 346
4.11.6.20. DMA Channel N Configuration Register ... 349
4.11.6.21. DMA Channel N Source Address Register ... 351
4.11.6.22. DMA Channel N Destination Address Register ... 351
4.11.6.23. DMA Channel N Link List Pointer Register ... 351

Vangotech

4.11.6.24. DMA Channel N Transfer Size Register ... 352
4.11.6.25. DMA Channel N Source/Destination Stride Register ... 352
4.11.6.26. DMA Channel N Source Address and Destination Address Expand Register ... 353
4.11.6.27. DMA Channel N Link List Pointer and UserUser-defined Signal Expand Register ... 354
4.12. DMAC for AHB ... 355
4.12.1. Overview ... 355
4.12.2. Features ... 355
4.12.3. Block Diagram ... 355
4.12.4. Programming Guidelines ... 356
4.12.4.1. General Description ... 356
4.12.4.2. Chain Transfer ... 356
4.12.4.3. DMA Hardware Handshake Mode ... 359
4.12.4.4. DMA Normal Mode ... 361
4.12.5. Register List ... 362
4.12.6. Register Description ... 362
4.12.6.1. Interrupt Status Register ... 362
4.12.6.2. Terminal Count Interrupt Status Register ... 363
4.12.6.3. Terminal Count Interrupt Status Clear Register ... 364
4.12.6.4. Error/Abort InterruptStatus Register ... 365
4.12.6.5. Error/Abort Interrupt Status Clear Register ... 366
4.12.6.6. Terminal Count Status Register ... 368
4.12.6.7. Error/Abort Status Register ... 368
4.12.6.8. Channel Enable Status Register ... 370
4.12.6.9. Channel Busy Status Register ... 371
4.12.6.10. Main Configuration Status Register ... 372
4.12.6.11. Synchronization Register ... 372
4.12.6.12. DMAC Revision Register ... 373
4.12.6.13. FTDMAC020 Feature Register ... 373
4.12.6.14. Channel Control Register ... 374
4.12.6.15. Channel Configuration Register ... 379
4.12.6.16. Channel Source Address Register ... 381
4.12.6.17. Channel Destination Address Register ... 381
4.12.6.18. Link List Descriptor Pointer ... 381
4.12.6.19. Transfer Size Register ... 381
4.13. ADC ... 382
4.13.1. Overview ... 382
4.13.2. Features ... 382
4.13.3. Block Diagram ... 382
4.13.4. Functional Descriptions ... 382
4.13.4.1. Sampling Modes ... 382
4.13.4.2. Discharge Time ... 383
4.13.4.3. Power Enable Time ... 384
4.13.4.4. Sampling Rate ... 385
4.13.4.5. Conversion Mode ... 385
4.13.4.6. DMA Function ... 387
4.13.4.7. Threshold Detection ... 388
4.13.4.8. Auto Power-down Mode ... 388
4.13.4.9. Temperature Sensing Mode ... 389
4.13.5. Register List ... 389
4.13.6. Register Description ... 390
4.13.6.1. Sensing Data Register ... 390
4.13.6.2. ADC Threshold Detect Programming Registers ... 390
4.13.6.3. Temperature Data Register ... 392
4.13.6.4. Temperature Threshold Detect Programming Register ... 392
4.13.6.5. ADC Control Register ... 393
4.13.6.6. ADC Miscellaneous Control Register ... 394
4.13.6.7. ADC Interrupt Enable Register ... 395
4.13.6.8. ADC Interrupt Status Register ... 395
4.13.6.9. ADC Timing Parameter Register ... 396
4.13.6.10. ADC Sampling Timing Parameter Control Register ... 397
4.13.6.11. ADC Sensing Discharge Time Register ... 398
4.13.6.12. ADC Clock Pre-scaler Control Register ... 399
4.13.6.13. ADC Scan Sequence Register ... 399

Vangotech

4.13.6.14. ADC DMAn Data Port Register 399
4.13.6.15. ADC DMAn Function Control Register 400
4.13.6.16. ADC DMAn Interrupt Enable Register 400
4.13.6.17. ADC DMAn Interrupt Status Register 400
4.14. TDC 402
4.14.1. Overview 402
4.14.2. Features 402
4.14.3. Block Diagram 402
4.14.4. Electrical Specification 402
4.14.4.1. Electrical Characteristics 402
4.14.4.2. Timing Diagramof Sampling Clock 403
4.14.4.3. Wake-up Timing Requirement 404
4.14.4.4. Trimming Table 404
4.14.5. Functional Descriptions 405
4.14.6. Register List 406
4.14.7. Register Description 406
4.14.7.1. TDC sensing data registers 406
4.14.7.2. TDC Threshold registers 406
4.14.7.3. TDC control register 407
4.14.7.4. TDC Trim register 407
4.14.7.5. TDC Interrupt enable register 407
4.14.7.6. TDC Interrupt status register 407
4.14.7.7. TDC Timing Parameter 0 control register 408
4.14.7.8. TDC Timing Parameter 1 control register 408
4.14.7.9. TDC clock pre-scaler register 408
4.14.7.10. IP revision register 408
4.15. eFUSE 409
4.15.1. Overview 409
4.15.2. Features 409
4.15.3. Register List 409
4.15.4. Register Description 409
4.15.4.1. eFuse data register 409
4.15.4.2. Timing parameter 1 register 410
4.15.4.3. Timing parameter 2 register 410
4.15.4.4. eFuse control and timing parameter 3 register 410
4.15.4.5. Page data status register 411
4.15.4.6. Protect key and status register 411
4.15.4.7. Unlock register 412
4.16. Port Controller 413
4.17. TRNG 419
4.17.1. Block diagram 419
4.17.2. Register map 419
4.17.3. Register Description 420
4.17.3.1. Control register 420
4.17.3.2. FIFOLevel register 421
4.17.3.3. FIFOThresh register 421
4.17.3.4. FIFODepth register 422
4.17.3.5. Key registers 422
4.17.3.6. TestData register 422
4.17.3.7. Threshold registers 422
4.17.3.8. Status register 423
4.17.3.9. InitWaitVal register 423
4.17.3.10. DisableOsc registers 423
4.17.3.11. SwOffTmrVal register 424
4.17.3.12. ClkDiv register 424
4.17.3.13. AIS31Conf registers 424
4.17.3.14. AIS31Status register 425
4.17.3.15. HwConf register 425
4.18. Crypto Engine 427
4.18.1. Overview 427
4.18.2. Features 427
4.18.3. Block Diagram 427
4.18.4. Functional Descriptions 427

Vangotech

4.18.4.1. Control bits ... 427
4.18.4.2. Using the DMA ... 428
4.18.5. Register List ... 428
4.18.6. Register Description ... 429
4.18.6.1. Fetcher / pusher registers ... 429
4.18.6.2. Interrupts ... 430
4.18.6.3. Control / status register ... 432
4.18.6.4. Scatter-gather descriptor ... 433
4.19. Secure Design ... 434
4.19.1. Secure JTAG ... 434
4.19.2. Secure SPI Host ... 434
4.19.3. Secure Efuse Controller ... 435
4.19.4. Application Note ... 435
4.19.5. TZC NSAID Mapping Table ... 436
4.20. Internal Interrupt Vector Generation ... 437
Chapter 5. Memory ... 444
5.1. EMC ... 446
5.1.1. Overview ... 446
5.1.3. Block Diagram ... 446
5.1.4. Operations and Functional Descriptions ... 447
5.1.4.1. Read Un-pack ... 447
5.1.4.2. Memory Retention Function Control ... 447
5.1.5. Register List ... 447
5.1.6. Register Definition ... 447
5.1.6.1. ECC Error Address Register ... 447
5.1.6.2. Memory Control Register ... 448
5.1.6.3. APB Control Register ... 450
5.1.6.4. IP Feature Register ... 450
5.1.6.5. IP Revision Register ... 450
5.2. eMMC ... 451
5.2.1. Overview ... 451
5.2.3. Block Diagram ... 451
5.2.4. Functional Descriptions ... 452
5.2.4.1. Block Description ... 452
5.2.4.2. DMA Transaction ... 453
5.2.4.3. SD Command and Data Input/Output Timing ... 456
5.2.4.4. CPRM Operation ... 458
5.2.5. Register List ... 460
5.2.6. Register Description ... 462
5.2.6.1. SDMA System Address Register ... 462
5.2.6.2. Block Size Register ... 462
5.2.6.3. Block Count Register ... 463
5.2.6.4. Argument 1 Register ... 464
5.2.6.5. Transfer Mode Register ... 464
5.2.6.6. Command Register ... 465
5.2.6.7. Response Registers 0 ~ 3 ... 467
5.2.6.8. Buffer Data Port Register ... 468
5.2.6.9. Present State Register ... 468
5.2.6.10. Host Control 1 Register ... 472
5.2.6.11. Power Control Register ... 474
5.2.6.12. Block Gap Control Register ... 474
5.2.6.13. Clock Control Register ... 476
5.2.6.14. Timeout Control Register ... 477
5.2.6.15. Software Reset Register ... 477
5.2.6.16. Normal Interrupt Status Register ... 478
5.2.6.17. Error Interrupt Status Register ... 480
5.2.6.18. Normal Interrupt Status Enable Register ... 482
5.2.6.19. Error Interrupt Status Enable Register ... 483
5.2.6.20. Normal Interrupt Signal Enable Register ... 484
5.2.6.21. Error Interrupt Signal Enable Register ... 484
5.2.6.22. Auto CMD12 Error Status Register ... 485
5.2.6.23. Host Control 2 Register ... 486
5.2.6.24. Capabilities 0/1 Register ... 488

Vangotech

5.2.6.25. Maximum Current Capabilities Register ... 492
5.2.6.26. Force Event Register for Auto CMD Error Status ... 493
5.2.6.27. Force Event Register for Error Interrupt Status ... 493
5.2.6.28. ADMA Error Status Register ... 494
5.2.6.29. ADMA System Address Register ... 495
5.2.6.30. Preset Value Register ... 495
5.2.6.31. Host Controller Version Register ... 496
5.2.6.32. Vendor-defined Register 0 ... 497
5.2.6.33. Vendor-defined Register 1 ... 498
5.2.6.34. Vendor-defined Register 2 ... 499
5.2.6.35. Vendor-defined Register 3 ... 499
5.2.6.36. Vendor-defined Register 4 ... 500
5.2.6.37. Vendor-defined Register 5 ... 501
5.2.6.38. Vendor-defined Register 6 ... 501
5.2.6.39. Vendor-defined Register 7 ... 501
5.2.6.40. Vendor-defined Register 8 ... 502
5.2.6.41. Vendor-defined Register 9 ... 502
5.2.6.42. DMA Handshake Enable Register ... 503
5.2.6.43. Hardware Attributes Register ... 504
5.2.6.44. IP Revision Register ... 504
5.2.6.45. Cipher Mode Control Register ... 504
5.2.6.46. Cipher Mode Status Register ... 505
5.2.6.47. Cipher Mode Signal Enable Register ... 506
5.2.6.48. Low Word of Input Data Register ... 506
5.2.6.49. High Word of Input Data Register ... 506
5.2.6.50. Low Word of Input Key Register ... 506
5.2.6.51. High Word of Input Key Register ... 506
5.2.6.52. Low Word of Output Data Register ... 506
5.2.6.53. High Word of Output Data Register ... 507
5.2.6.54. Secret Constant Table Data Port ... 507
Chapter 6. Display ... 508
6.1. LCDC ... 509
6.1.1. Overview ... 509
6.1.2. Block Diagram ... 511
6.1.3. Operations and Functional Descriptions ... 512
6.1.3.1. Application Connection ... 512
6.1.4. Register List ... 512
6.1.5. Register Descriptions ... 514
6.1.5.1. LCD Function Enable Parameter ... 514
6.1.5.2. LCD Panel Pixel Parameter ... 517
6.1.5.3. LCD Interrupt Enable Mask Parameter ... 519
6.1.5.4. LCD Interrupt Status Clear Parameter ... 519
6.1.5.5. LCD Interrupt Status Parameter ... 520
6.1.5.6. Frame Buffer Parameter ... 520
6.1.5.7. LCD Panel Image0 Frame0 Base Address ... 521
6.1.5.8. LCD Panel Image1 Frame0 Base Address ... 522
6.1.5.9. LCD Panel Image2 Frame0 Base Address ... 522
6.1.5.10. LCD Panel Image3 Frame0 Base Address ... 522
6.1.5.1. PatGen Pattern Bar Distance Parameter ... 523
6.1.5.2. FIFO Threshold Control Parameter ... 523
6.1.5.3. Bus Bandwidth Control Parameter ... 524
6.1.5.4. Revision Register ... 525
6.1.5.1. LCD Panel VBI Base Address ... 525
6.1.5.2. FIFO Threshold Control Parameter ... 526
6.1.5.3. LCD Horizontal Timing Control Parameter ... 526
6.1.5.4. LCD Vertical Timing Control Parameter ... 527
6.1.5.5. LCD Vertical Back Porch Parameter ... 528
6.1.5.6. LCD Polarity Control Parameter ... 528
6.1.5.7. PiP Blending Parameter ... 529
6.1.5.8. PiP Sub-Picture1 Position Parameter ... 530
6.1.5.9. PiP Sub-Picture1 Dimension Parameter ... 530
6.1.5.10. PiP Sub-Picture2 Position Parameter ... 531
6.1.5.11. PiP Sub-Picture2 Dimension Parameter ... 531

Vangotech

6.1.5.12. PiP Image Priority Parameter 532
6.1.5.13. PiP/PoP Image Format 1 Parameter 532
6.1.5.14. PiP/PoP Image Format 2 ParameParameter 535
6.1.5.15. PiP Color Key1 537
6.1.5.16. PiP Color Key2 537
6.1.5.17. PiP Color Key3 537
6.1.5.18. PiP Sub-Picture3 Position 537
6.1.5.19. PiP Sub-Picture3 Dimension Parameter 538
6.1.5.20. LCD Palette RAM Write Accessing Port 538
6.1.5.21. LCD Color Management Parament0 Parameter 539
6.1.5.22. LCD Color Management Parament1 Parameter 539
6.1.5.23. LCD Color Management Parament2 Parameter 540
6.1.5.24. LCD Color Management Parament3 Parameter 540
6.1.5.25. Lookup Table of LCD Gamma Red 540
6.1.5.26. Lookup Table of LCD Gamma Green 541
6.1.5.27. Lookup Table of LCD Gamma Blue 542
6.1.5.28. Scaler Control Registers 542
6.1.5.29. Horizontal Resolution Register of Scaler Input 542
6.1.5.30. Vertical Resolution Register of Scaler Input 542
6.1.5.31. Horizontal Resolution Register of Scaler Output 543
6.1.5.32. Vertical Resolution Register of Scaler Output 543
6.1.5.33. Scaler Control Register 543
6.1.5.34. Horizontal High Threshold Register 544
6.1.5.35. Horizontal Low Threshold Register 545
6.1.5.36. Vertical High Threshold Register 545
6.1.5.37. Vertical Low Threshold Register 545
6.1.5.38. Scaler Resolution Parameter 545
6.1.5.39. Cursor Position Control Register 546
6.1.5.40. Cursor Color Control Register 546
6.1.5.41. 32 * 32 Cursor Font Database Write Accessing Port 547
6.1.5.42. 64 * 64 Cursor Font Database Write Accessing Port 549
6.1.5.43. LCM Function Registers 554
6.1.5.44. Virtual Screen Control Regi ster 556
6.1.5.45. Virtual Screen1 Control Register 557
6.1.5.46. Virtual Screen2 Control Register 557
6.1.5.47. Virtual Screen3 Control Register 557
6.1.5.48. OSD Control Registers with Complex Configuration 558
6.1.5.49. OSD Windows On/Off Switch Register 558
6.1.5.50. OSD Font Control Register 0 558
6.1.5.51. OSD Font Control Register 1 560
6.1.5.52. OSD Windows 1 to 4 Control Register 0 (Offset 0x200C, 0x2018, 0x2024, 0x2030) 560
6.1.5.53. OSD Windows 1 to 4 Control Register 1 (Offset 0x2010, 0x201C, 0x2028, 0x2034) 561
6.1.5.54. OSD Window 1 to 4 Control Register 2 (Offset 0x2014, 0x2020, 0x202C, 0x2038) 562
6.1.5.55. OSD Multi Multi-Color Font Base Address (Offset 0x203C) 562
6.1.5.56. OSD Palette Color Registers 0 ~ 15 (Offset 0x2040 ~ 0x207C) 562
6.1.5.57. OSD Font RAM Data (Offset 0x10000 ~ 0x147FC) 563
6.1.5.58. OSD Display Display-RAM Data (Offset 0x20000 ~ 0x207FC) 567
6.1.5.59. OSD Dimension Control (Offset 0x2000) 567
6.1.5.60. OSD Position Control (Offset 0x2004) 568
6.1.5.61. OSD Foreground Color Control (Offset 0x2008) 568
6.1.5.62. OSD Background Color C ontrol (Offset 0x200C) 568
6.1.5.63. OSD Font Database Write Accessing Port (Offset 0x8000 ~ 0xBFFC) 569
6.1.5.64. OSD Window Attribute Write Accessing Port (Offset 0xC000H ~ 0xC7FC) 570
Chapter 7. Audio 571
7.1. I2S 572
Chapter 8. Interface 573
8.1. USB2.0 575
8.1.1. Overview 575
8.1.2. Features 575
8.1.3. Block Diagram 575
8.1.4. Timing Diagram 576
8.1.5. Register List 576
8.1.6. Register Description 578

Vangotech

8.1.6.1. HC Capability Register 578
8.1.6.2. HC Structural Parameters 579
8.1.6.3. HC Capability Parameters 579
8.1.6.4. HC USB Command Register 580
8.1.6.5. HC USB Status Register 581
8.1.6.6. HC USB Interrupt Enable Register 582
8.1.6.7. HC Frame Index Register 583
8.1.6.8. HC Periodic Frame List Base Address Register 584
8.1.6.9. HC Current Asynchronous List Address Register 584
8.1.6.10. HC Port Status and Control Register 584
8.1.6.11. HC Miscellaneous Register 587
8.1.6.12. HC Full-Speed (FS) EOF1 Timing Point Register 589
8.1.6.13. HC High-Speed (HS) EOF1 Timing Point Register 591
8.1.6.14. OTG Control Status Register 592
8.1.6.15. OTG Interrupt Status Register 595
8.1.6.16. OTG Interrupt Enable Register 596
8.1.6.17. Global HC/OTG/DEV Interrupt Status Register 596
8.1.6.18. Global Mask of HC/OTG/DEV Interrupt Register 597
8.1.6.19. Revision Information Register 597
8.1.6.20. Configured Information Register 598
8.1.6.21. Device Main Control Register 598
8.1.6.22. Device Address Register 600
8.1.6.23. Device Test Register 601
8.1.6.24. Device SOF Frame Number Register 601
8.1.6.25. Device SOF Mask Timer Register 602
8.1.6.26. PHY Test Mode Selector Register 602
8.1.6.27. Device Vendor-Specific I/O Control Register 603
8.1.6.28. Device CX Configuration Status Register 603
8.1.6.29. Device CX Configuration and FIFO Empty Status Register 603
8.1.6.30. Device Idle Counter Register 604
8.1.6.31. Device Mask of Interrupt Group Register 605
8.1.6.32. Device Mask of Interrupt Source Group 0 Register 605
8.1.6.33. Device Mask of Interrupt Source Group 1 Register 606
8.1.6.34. Device Mask of Interrupt Source Group 2 Register 607
8.1.6.35. Device Interrupt Group Register 609
8.1.6.36. Device Interrupt Source Group 0 Register 609
8.1.6.37. Device Interrupt Source Group 1 Register 610
8.1.6.38. Device Interrupt Source Group 2 Register 612
8.1.6.39. Device Receive Zero-Length Data Packet Register 614
8.1.6.40. Device Transfer Zero-length Data Packet Register 614
8.1.6.41. Device Isochronous Sequential Error/Abort Register 615
8.1.6.42. Device IN Endpoint n MaxPacketSize Register 616
8.1.6.43. Device OUT Endpoint n MaxPacketSize Register 617
8.1.6.44. Device Endpoint 1 ~ 4 Map Register 617
8.1.6.45. Device Endpoint 5 ~ 8 Map Register 619
8.1.6.46. Device FIFO Map Register 620
8.1.6.47. Device FIFO Configuration Register 621
8.1.6.48. Device FIFO n Instruction and Byte Count Register 624
8.1.6.49. Device DMA Target FIFO Number Register 624
8.1.6.50. Device DMA Controller Parameter Setting 0 Register 625
8.1.6.51. Device DMA Controller Parameter Setting 1 Register 626
8.1.6.52. Device DMA Controller Parameter Setting 2 Register 628
8.1.6.53. Device DMA Controller Parameter Setting 3 Register 628
8.1.6.54. Device DMA Controller Parameter Setting 4 Register 629
8.1.6.55. Device Virtual DMA CXF Parameter Setting 1 Register 629
8.1.6.56. Device Virtual DMA CXF Parameter Setting 2 Register 630
8.1.6.57. Device Virtual DMA FIFO0 Parameter Setting 1 Register 630
8.1.6.58. Device Virtual DMA FIFO0 Parameter Setting 2 Register 631
8.1.6.59. Device Virtual DMA FIFO1 Parameter Setting 1 Register 632
8.1.6.60. Device Virtual DMA FIFO1 Parameter Setting 2 Register 633
8.1.6.61. Device Virtual DMA FIFO2 Parameter Setting 1 Register 633
8.1.6.62. Device Virtual DMA FIFO2 Parameter Setting 2 Register 634
8.1.6.63. Device Virtual DMA FIFO3 Parameter Setting 1 Register 634

Vangotech

8.1.6.64. Device Virtual DMA FIFO3 Parameter Setting 2 Register 635
8.1.6.65. Device Interrupt Source Group 3 Register 636
8.1.6.66. Device Mask of Interrupt Source Group 3 Register 638
8.1.6.67. Device Virtual DMA Control Register 639
8.1.6.68. Device LPM Capability Register 640
8.1.6.69. Device Interrupt Source Group 4 Register 640
8.1.6.70. Device Mask of Interrupt Source Group 4 Register 640
8.2. SSP 641
8.2.1. Overview 641
8.2.2. Features 641
8.2.3. Functional Block Diagram 641
8.2.4. Initialization/Application Information 642
8.2.4.1. Initialization in Master Mode 643
8.2.4.2. Initialization in Slave Mode 643
8.2.4.3. Disable/Enable Program Sequence 644
8.2.4.4. SPI Flash Controller Write Command Program Sequence 644
8.2.4.5. SPI Flash Controller Read Command Program Sequence 644
8.2.4.6. SPI Flash Controller Read Command Control by Write Program Sequence 644
8.2.4.7. LCD Driver 3-line Serial Interface Write Command & D/CX Control by Write Data Program Sequence 645
8.2.4.8. LCD Driver 3-line Serial Interface Write Command & D/CX Control by Register Program Sequence 645
8.2.4.9. LCD Driver 4-line Serial Interface Write Command & D/CX Control by Write Data Program Sequence 646
8.2.4.10. LCD Driver 4-line Serial Interface Write Command & D/CX Control by Register Program Sequence 646
8.2.4.11. LCD Driver 3-line Serial Interface Read Command Program Sequence 647
8.2.4.12. LCD Driver 4-line Serial Interface Read Command Program Sequence 647
8.2.5. Register List 647
8.2.6. Register Description 648
8.2.6.1. SSP Control Register 0 648
8.2.6.2. SSP Control Register 1 652
8.2.6.3. SSP Control Register 2 652
8.2.6.4. SSP Status Register 655
8.2.6.5. Interrupt Control Register 655
8.2.6.6. Interrupt Status Register 657
8.2.6.7. SSP Transmit/Receive Data Register 657
8.2.6.8. SSP Control Register 3 658
8.2.6.9. SSP Revision Register 658
8.2.6.10. SSP Feature Register 659
8.3. SPIF 660
8.3.1. Overview 660
8.3.2. Features 660
8.3.3. Block Diagram 660
8.3.4. Operations and Functional Descriptions 660
8.3.4.1. Command Slave Port 660
8.3.4.2. PIO Mode 661
8.3.4.3. XIP Slave Port 661
8.3.4.4. XIP Prefetch Function 661
8.3.4.5. SPI Flash Operation 662
8.3.5. Initialization/Application Information 667
8.3.5.1. Data Transfer in PIO Mode at Command-based Slave Port 667
8.3.5.2. Read Data Transfer at XIP Slave Port (Port Switching Sequence) 668
8.3.5.3. Recommended Boot Sequence 669
8.3.5.4. Other Operations 669
8.3.6. Register List 669
8.3.7. Register Description 670
8.3.7.1. Command Word 0 ~ 3 670
8.3.7.2. Control Register 674
8.3.7.3. AC Timing Register 676
8.3.7.4. Status Register 676
8.3.7.5. Interrupt Control Register 676
8.3.7.6. Interrupt Status Register 677

Vangotech

8.3.7.7. SPI Read Status Register ... 677
8.3.7.8. SPI Flash Size Register ... 677
8.3.7.9. XIP Command Word ... 677
8.3.7.10. XIP Prefetch Function Option ... 679
8.3.7.11. Revision Register ... 679
8.3.7.12. Feature Register ... 679
8.3.7.13. Data Port Register ... 680
8.4. SPI2AHB ... 680
8.4.1. Overview ... 680
8.4.2. Features ... 680
8.4.3. Functional Block Diagram ... 680
8.4.4. Function Description ... 681
8.4.4.1. SPI Timing ... 681
8.4.4.2. SPI Write Protocol ... 682
8.4.4.3. SPI Single Write ... 682
8.4.4.4. SPI Burst Write ... 682
8.4.5. SPI Read Protocol ... 682
8.4.5.1. SPI Single Read ... 683
8.4.5.2. SPI Burst Read ... 683
8.5. I2C ... 684
8.5.1. Overview ... 684
8.5.2. Features ... 684
8.5.3. Block Diagram ... 684
8.5.3.1. Glitch Suppression Circuit ... 684
8.5.3.2. I2C_MnS Circuit ... 684
8.5.4. Software Programming Guide ... 685
8.5.4.1. Programming Guide for TX/RX Slave Mode ... 685
8.5.4.2. Programming Guide for RX in Slave Mode with Repeat-Start ... 685
8.5.4.3. Programming Guide for TX in Master Mode ... 686
8.5.4.4. Programming Guide for RX in Master Mode ... 686
8.5.4.5. Programming Guide for TX in Master Mode with HS-Mode (or START Byte) ... 687
8.5.4.6. Programming Guide for RX in Master Mode with HS-Mode (or START Byte) ... 688
8.5.4.7. Programming Guide for TX/RX in Slave Mode with HS-Mode (or START Byte) ... 688
8.5.4.8. Programming Guide for Master TX Burst Mode ... 689
8.5.4.9. Programming Guide for Master RX Burst Mode ... 690
8.5.5. Register List ... 690
8.5.6. Register Description ... 690
8.5.6.1. I2C Control Register ... 691
8.5.6.2. I2C Status Registers ... 693
8.5.6.3. I2C Clock Division Register ... 695
8.5.6.4. I2C Data Register ... 696
8.5.6.5. I2C Address Register ... 696
8.5.6.6. I2C Set/Hold Time and Glitch Suppression Setting Register ... 697
8.5.6.7. I2C Bus Monitor Register ... 698
8.5.6.8. I2C Burst Mode Register ... 698
8.5.6.9. I2C Revision Register ... 698
8.6. UART ... 699
8.6.1. Overview ... 699
8.6.2. Features ... 699
8.6.3. Block Diagram ... 699
8.6.3.1. APB Interface ... 700
8.6.3.2. Prescaler ... 700
8.6.3.3. Baud Rate Generator ... 700
8.6.3.4. TX FIFO and RX FIFO ... 700
8.6.3.5. Status FIFO(ST FIFO) ... 700
8.6.3.6. Configuration and Status Registers ... 700
8.6.3.7. Modem Control and Flags ... 700
8.6.3.8. SIR Logic ... 700
8.6.3.9. FIR Logic ... 700
8.6.3.10. SIP Generator ... 701
8.6.4. Operations and Functional Descriptions ... 701
8.6.4.1. UART Mode ... 701
8.6.4.2. IrDA 1.3 SIR Mode ... 702

Vangotech

8.6.4.3. IrDA 1.3 FIR Mode 703
8.6.4.4. DMA Operation in FIR Mode 705
8.6.5. Programming Guidelines 707
8.6.5.1. UART Mode 707
8.6.5.2. SIR Mode 708
8.6.5.3. FIR Mode 710
8.6.5.4. Clock Frequency Limitation 712
8.6.6. Register List 713
8.6.7. Register Description 715
8.6.7.1. Receiver Buffer Register 715
8.6.7.2. Transmit Holding Register 715
8.6.7.3. Interrupt Enable Register 716
8.6.7.4. Interrupt Identification Register 717
8.6.7.5. FIFO Control Register 720
8.6.7.6. Line Control Register 721
8.6.7.7. Modem Control Register 723
8.6.7.8. Line Status Register 725
8.6.7.9. Testing Register 727
8.6.7.10. Modem Status Register 727
8.6.7.11. Scratch Pad Register 728
8.6.7.12. Baud-Rate Divisor Latch 728
8.6.7.13. Prescaler Register 729
8.6.7.14. Mode Definition Register 729
8.6.7.15. Auxiliary Control Register 730
8.6.7.16. Transmit Frame Frame-Length Register 731
8.6.7.17. Maximum Receiver Frame Frame-Length 731
8.6.7.18. Preamble Length Register (PLR , Offset: 0x38) 732
8.6.7.19. Interrupt Identification Register in FIR Mode 732
8.6.7.20. Interrupt Identification Register in FIR Mode 733
8.6.7.21. Interrupt Identification Enable Register in FIR Mode 734
8.6.7.22. Interrupt Identification Enable Register in FIR Mode 734
8.6.7.23. Status FIFO Line Status Register 735
8.6.7.24. Status FIFO Received Frame-Length Register Low 735
8.6.7.25. Status FIFO Received Frame-Length Register High 735
8.6.7.26. Link Status Register in FIR Mode 736
8.6.7.27. Link Status Interrupt Enable Register in FIR Mode 736
8.6.7.28. RX FIFO Count Register 737
8.6.7.29. Last Frame-Length Register Low 737
8.6.7.30. Last Frame Length Register High 737
8.6.7.31. Feature Register 738
8.6.7.32. Revision Register Digit 1 739
8.6.7.33. Revision Register Digit 2 739
8.6.7.34. Revision Register Digit 3 739
8.7. MAC 740
8.7.1. Overview 740
8.7.2. Features: 740
8.7.3. Block Diagram 741
8.7.3.1. AHB/AXI Master 741
8.7.3.2. AHB/APB Slave 741
8.7.3.3. DMA Arbitrator 742
8.7.3.4. TXDMA 742
8.7.3.5. RXDMA 742
8.7.3.6. TXMAC 742
8.7.3.7. RXMAC 742
8.7.3.8. REG 743
8.7.3.9. TSU 743
8.7.3.10. IFSEL 743
8.7.3.11. MII2RMII 743
8.7.3.12. GMII2RGMII 743
8.7.3.13. TXFIFO and RXFIFO 743
8.7.3.14. xMII PHY 743
8.7.4. Operations and Functional Descriptions 743
8.7.4.1. Transmit Descriptors and Data Buffers 743

Vangotech

8.7.4.2. Receive Descriptors and Data Buffers ... 747
8.7.4.3. Transmitting Packets ... 747
8.7.4.4. Receiving Packets ... 748
8.7.4.5. Ethernet Address Filtering ... 748
8.7.4.6. DMA Arbitration Scheme ... 748
8.7.4.7. Wake-On-LAN ... 749
8.7.4.8. Flow Control ... 750
8.7.4.9. Supported Ethernet Frame Type for Checksum Offload ... 750
8.7.4.10. IEEE 1588 PTP Frame Reorganization ... 751
8.7.5. Initialization/Application Information ... 751
8.7.5.1. Frame Transmitting Procedure ... 752
8.7.5.2. Frame Receiving Procedure ... 753
8.7.5.3. Procedures of Entering and Exiting Wake-On-LAN ... 754
8.7.6. Programming Guidelines ... 754
8.7.6.1. Ethernet Frame Formats ... 754
8.7.6.2. MII Management Interface ... 755
8.7.7. Register List ... 756
8.7.8. Register Definition ... 759
8.7.8.1. Interrupt Status Register ... 759
8.7.8.2. Interrupt Enable Register ... 760
8.7.8.3. Most Significant Address Register ... 760
8.7.8.4. Least Significant Address Register ... 761
8.7.8.5. Multicast Address Hash Table 0 Register ... 761
8.7.8.6. Multicast Address Hash Table 1 Register ... 761
8.7.8.7. Normal Priority Transmit Poll Demand Register ... 761
8.7.8.8. Receive Poll Demand Register ... 761
8.7.8.9. Normal Priority Transmit Ring Base Address Register ... 761
8.7.8.10. Receive Ring Base Address Register ... 762
8.7.8.11. High Priority Transmit Poll Demand Register ... 762
8.7.8.12. High Priority Transmit Ring Base Address Register ... 762
8.7.8.13. TX Interrupt Timer Control Register ... 762
8.7.8.14. RX Interrupt Timer Control Register ... 764
8.7.8.15. Automatic Polling Timer Control Register ... 766
8.7.8.16. DMA Burst Length and Arbitration Control Register ... 767
8.7.8.17. DMA/FIFO State Register, DMAFIFOS ... 769
8.7.8.18. Transmit Priority Arbitration and FIFO Control Register ... 770
8.7.8.19. Receive Buffer Size Register ... 771
8.7.8.20. MAC Control Register ... 771
8.7.8.21. MAC Status Register ... 773
8.7.8.22. Test Mode Register ... 773
8.7.8.23. PHY Control register ... 774
8.7.8.24. PHY Data Register ... 775
8.7.8.25. Flow Control Register ... 775
8.7.8.26. Back Pressure Register ... 776
8.7.8.27. Wake-On-LAN Control Register ... 776
8.7.8.28. Wake-On-LAN Status Register ... 777
8.7.8.29. Wake-up Frame CRC Register ... 778
8.7.8.30. Wake-up Frame Byte Mask 1st Double-word Register ... 778
8.7.8.31. Wake-up Frame Byte Mask 2nd Double-word Register ... 779
8.7.8.32. Wake-up Frame Byte Mask 3rd Double-word Register ... 779
8.7.8.33. Wake-up Frame Byte Mask 4th Double-word Register ... 780
8.7.8.34. Normal Priority Transmit Ring Pointer Register ... 780
8.7.8.35. High Priority Transmit Ring Pointer Register ... 781
8.7.8.36. Receive Ring Pointer Register ... 781
8.7.8.37. TPKT_CNT Counter Register ... 781
8.7.8.38. TXMCOL_CNT and TXSCOL_CNT Counter Register, TX_CNT2 (Offset: 0xA4) ... 781
8.7.8.39. TXECOL_CNT and TXFAIL_CNT Counter Register ... 782
8.7.8.40. TXLCOL_CNT and TXUNDERUN_CNT Counter Register ... 782
8.7.8.41. RPKT_CNT Counter Register ... 782
8.7.8.42. BROPKT_CNT Counter Register ... 782
8.7.8.43. MULPKT_CNT Counter Register ... 782
8.7.8.44. RPF_CNT and AEP_CNT Counter Register ... 782
8.7.8.45. RUNT_CNT Counter Register ... 783

Vangotech

8.7.8.46. CRCER_CNT and FTL_CNT Counter Register ... 783
8.7.8.47. RCOL_CNT and RLOST_CNT Counter Register ... 783
8.7.8.48. BIST Pattern Control Register ... 783
8.7.8.49. Broadcast and Multicast Receiving Control Register ... 784
8.7.8.50. Error Response Control Register ... 785
8.7.8.51. AXI Additional Control Information Register ... 785
8.7.8.52. MAC Interface Selection Register ... 786
8.7.8.53. SW Reset Cycle Count Register ... 786
8.7.8.54. EEE Control Register ... 786
8.7.8.55. Revision Register ... 786
8.7.8.56. Feature Register ... 787
8.7.8.57. PTP RX Unicast IP Destination Address Register ... 788
8.7.8.58. PTP TX Unicast IP Destination Address Register ... 788
8.7.8.59. PTP TX Event Frame Register ... 788
8.7.8.60. PTP RX Event Frame Register ... 789
8.7.8.61. PTP TX Peer Frame Register ... 789
8.7.8.62. PTP RX Peer Frame Register ... 789
8.7.8.63. PTP Timer Register, PTP_TMR ... 790
8.7.8.64. PTP Period Increment 0 Register ... 790
8.7.8.65. PTP Period Increment 1 Register ... 791
8.7.8.66. PTP Period Offset Increment Register ... 791
8.7.8.67. PTP Timer Adjustment Register ... 791
8.8. GPIO ... 792
8.8.1. Overview ... 792
8.8.2. Block Diagram ... 792
8.8.2.1. Bypass Architecture Diagram ... 792
8.8.2.2. Connection Diagram ... 793
8.8.3. Programming Guidelines ... 793
8.8.4. Register List ... 794
8.8.5. Register Description ... 796
8.8.5.1. GPIO Data-out Register ... 796
8.8.5.2. GPIO Data-in Register ... 796
8.8.5.3. Pin Direction Register ... 796
8.8.5.4. Pin Bypass Mode Register ... 797
8.8.5.5. GPIO Data Bit Set Register ... 797
8.8.5.6. GPIO Data Bit Clear Register ... 797
8.8.5.7. Pin Pull Enable Register ... 797
8.8.5.8. Pin Pull Type Register ... 797
8.8.5.9. Interrupt Enable Register ... 798
8.8.5.10. Interrupt Raw State Register ... 798
8.8.5.11. Interrupt Masked State Register ... 798
8.8.5.12. Interrupt Mask Register ... 798
8.8.5.13. Interrupt Clear Register ... 799
8.8.5.14. IntrTrigger Register ... 799
8.8.5.15. Interrupt Both Edge Trigger Register ... 799
8.8.5.16. Interrupt Rise or Neg Edge Trigger Register ... 799
8.8.5.17. Bounce Enable Register ... 799
8.8.5.18. Bounce Clock Pre-scale Register ... 800
8.9. CAN ... 801
8.9.1. Overview ... 801
8.9.2. Features ... 801
8.9.3. Block Diagram ... 801
8.9.4. Operations and Functional Descriptions ... 801
8.9.4.1. Operation Mode ... 801
8.9.4.2. Transmit and Receive Buffer ... 802
8.9.4.3. Clock and Reset ... 803
8.9.5. Programming Guidelines ... 804
8.9.6. Register List ... 804
8.9.7. Register Definition ... 810
8.9.7.1. Command Request Register ... 810
8.9.7.2. Interrupt Request Enable Register ... 811
8.9.7.3. Control Enable 0 Register ... 812
8.9.7.4. Control Enable 1 Register ... 812

Vangotech

8.9.7.5. Bus Status Register ... 813
8.9.7.6. Transmit and Receive Buffer Status Register ... 814
8.9.7.7. Transmit Status Register ... 814
8.9.7.8. Reserved ... 815
8.9.7.9. Transmit Flexible Data-Rate Register 0 ... 815
8.9.7.10. Transmit Remote and Extended Identifier Enable Register 0 ... 816
8.9.7.11. Transmit Extended Identifier Low Register 0 ... 816
8.9.7.12. Transmit Extended Identifier Middle Register 0 ... 816
8.9.7.13. Transmit Extended Identifier High Register 0 ... 816
8.9.7.14. Transmit Base Identifier Low and Data Length Register 0 ... 817
8.9.7.15. Transmit Base Identifier High Register 0 ... 817
8.9.7.16. Reserved ... 817
8.9.7.17. Transmit Flexible Data-Rate Register 1 ... 817
8.9.7.18. Transmit Remote and Extended Identifier Enable Register 1 ... 818
8.9.7.19. Transmit Extended Identifier Low Register 1 ... 818
8.9.7.20. Transmit Extended Identifier Middle Register 1 ... 818
8.9.7.21. Transmit Extended Identifier High Register 1 ... 819
8.9.7.22. Transmit Base Identifier Low and Data Length Register 1 ... 819
8.9.7.23. Transmit Base Identifier High Register 1 ... 819
8.9.7.24. Reserved ... 820
8.9.7.25. Transmit Flexible Data-rate Register 2 ... 820
8.9.7.26. Transmit Remote and Extended Identifier Enable Register 2 ... 820
8.9.7.27. Transmit Extended Identifier Low Register 2 ... 821
8.9.7.28. Transmit Extended Identifier Middle Register 2 ... 821
8.9.7.29. Transmit Extended Identifier High Register 2 ... 821
8.9.7.30. Transmit Base Identifier Low and Data Length Register 2 ... 821
8.9.7.31. Transmit Base Identifier High Register 2 ... 822
8.9.7.32. Reserved ... 822
8.9.7.33. Acceptance Filter for Extended Identifier Low Bit Register 0 ... 822
8.9.7.34. Acceptance Filter for Extended Identifier Middle Bit Register 0 ... 822
8.9.7.35. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 0 ... 823
8.9.7.36. Acceptance Filter for Base Identifier High Bit Register 0 ... 823
8.9.7.37. Acceptance Filter for Extended Identifier Low Bit Register 1 ... 823
8.9.7.38. Acceptance Filter for Extended Identifier Middle Bit Register 1 ... 823
8.9.7.39. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 1 ... 823
8.9.7.40. Acceptance Filter for Base Identifier High Bit Register 1 ... 823
8.9.7.41. Acceptance Filter for Extended Identifier Low Bit Register 2 ... 823
8.9.7.42. Acceptance Filter for Extended Identifier Middle Bit Register 2 ... 824
8.9.7.43. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 2 ... 824
8.9.7.44. Acceptance Filter for Base Identifier High Bit Register 2 ... 824
8.9.7.45. Acceptance Filter for Extended Identifier Low Bit Register 3 ... 824
8.9.7.46. Acceptance Filter for Extended Identifier Middle Bit Register 3 ... 824
8.9.7.47. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 3 ... 824
8.9.7.48. Acceptance Filter for Base Identifier High Bit Register 3 ... 825
8.9.7.49. Acceptance Filter for Extended Identifier Low Bit Register 4 ... 825
8.9.7.50. Acceptance Filter for Extended Identifier Middle Bit Register 4 ... 825
8.9.7.51. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 4 ... 825
8.9.7.52. Acceptance Filter for Base Identifier High Bit Register 4 ... 825
8.9.7.53. Acceptance Filter for Extended Identifier Low Bit Register 5 ... 825
8.9.7.54. Acceptance Filter for Extended Identifier Middle Bit Register 5 ... 826
8.9.7.55. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 5 ... 826
8.9.7.56. Acceptance Filter for Base Identifier High Bit Register 5 ... 826
8.9.7.57. Acceptance Filter Data Byte0 Register 0 ... 826
8.9.7.58. Acceptance Filter Data Byte1 Register 0 ... 826
8.9.7.59. Acceptance Filter Data Byte0 Register 1 ... 826
8.9.7.60. Acceptance Filter Data Byte1 Register 1 ... 826
8.9.7.61. Acceptance Filter Control Register 0 ... 826
8.9.7.62. Acceptance Filter Control Register 1 ... 827
8.9.7.63. Reserved ... 828
8.9.7.64. Reserved ... 828
8.9.7.65. Filter Mask Extended Identifier Low Register 0 ... 828
8.9.7.66. Filter Mask Extended Identifier Middle Register 0 ... 828
8.9.7.67. Filter Mask Base Identifier Low and Extended Identifier High 0 ... 828

Vangotech

8.9.7.68. Filter Mask Base Identifier High Register 0 ... 829
8.9.7.69. Filter Mask Extended Identifier Low Register 1 ... 829
8.9.7.70. Filter Mask Extended Identifier Middle Register 1 ... 829
8.9.7.71. Filter Mask Base Identifier Low and Extended Identifier High 1 ... 829
8.9.7.72. Filter Mask Base Identifier High Register 1 ... 829
8.9.7.73. Filter Mask Data Byte0 Register 0 ... 829
8.9.7.74. Filter Mask Data Byte1 Register 0 ... 829
8.9.7.75. Filter Mask Data Byte0 Register 0 ... 829
8.9.7.76. Filter Mask Data Byte1 Register 1 ... 830
8.9.7.77. Filter Mask Control Register ... 830
8.9.7.78. Reserved ... 830
8.9.7.79. Reserved ... 831
8.9.7.80. Reserved ... 831
8.9.7.81. Received Flexible Data-Rate Register 0 ... 831
8.9.7.82. Received Remote and Extended Identifier Enable Register 0 ... 831
8.9.7.83. Received Extended Identifier Low Register 0 ... 831
8.9.7.84. Received Extended Identifier Middle Register 0 ... 832
8.9.7.85. Received Extended Identifier High Register 0 ... 832
8.9.7.86. Received Base Identifier Low and Data Length Register 0 ... 832
8.9.7.87. Received Base Identifier High Register 0 ... 833
8.9.7.88. Reserved ... 833
8.9.7.89. Received Flexible Data-Rate Register 1 ... 833
8.9.7.90. Received Remote and Extended Identifier Enable Register 1 ... 833
8.9.7.91. Received Extended Identifier Low Register 1 ... 833
8.9.7.92. Received Extended Identifier Middle Register 1 ... 834
8.9.7.93. Received Extended Identifier High Register 1 ... 834
8.9.7.94. Received Base Identifier Low and Data Length Register 1 ... 834
8.9.7.95. Received Base Identifier High Register 1 ... 834
8.9.7.96. Reserved ... 835
8.9.7.97. Nominal Bit Timing Configuration 0 Register ... 835
8.9.7.98. Nominal Bit Timing Configuration 1 Register ... 835
8.9.7.99. Nominal Bit Timing Configuration 2 Register ... 835
8.9.7.100. Nominal Bit Timing Configuration 3 Register ... 835
8.9.7.101. Nominal Bit Timing Configuration 4 Register ... 835
8.9.7.102. Reserved ... 836
8.9.7.103. Reserved ... 836
8.9.7.104. Reserved ... 836
8.9.7.105. Data Bit Timing Configuration 0 Register ... 836
8.9.7.106. Data Bit TimingConfiguration 1 Register ... 836
8.9.7.107. Data Bit Timing Configuration 2 Register ... 837
8.9.7.108. Data Bit Timing Configuration 3 Register ... 837
8.9.7.109. Receiver Error Counter Low Register ... 837
8.9.7.110. Receiver Error Counter HighRegister ... 837
8.9.7.111. Transmitter Error Counter Low Register ... 837
8.9.7.112. Transmitter Error Counter High Register ... 837
8.9.7.113. Error Type Register ... 837
8.9.7.114. Transmitter and Receive Error Detected Register ... 838
8.9.7.115. Reserved ... 838
8.9.7.116. Reserved ... 838
8.9.7.117. Receive Buffer Time Stamp Low 0 ... 839
8.9.7.118. Receive Buffer Time Stamp High 0 ... 839
8.9.7.119. Receive Buffer Time Stamp Low 1 ... 839
8.9.7.120. Receive Buffer Time Stamp High 1 ... 839
8.9.7.121. Interrupt Request Register ... 839
8.9.7.122. Transmit Buffer0 Data Register ... 840
8.9.7.123. Transmit Buffer1 Data Register ... 840
8.9.7.124. Transmit Buffer2 Data Register ... 841
8.9.7.125. Receive Buffer0 Data Register ... 841
8.9.7.126. Receive Buffer1 Data Register ... 841
8.10. KBC ... 842
8.10.1. Overview ... 842
8.10.2. Features ... 842
8.10.3. Block Diagram ... 842

Vangotech

8.10.3.1. APB Interrupt Status and Control Register ... 842
8.10.3.2. Divider ... 842
8.10.3.3. Receiver ... 842
8.10.3.4. Transmitter ... 843
8.10.3.5. Sync.Logic ... 843
8.10.3.6. KMI Controller Logic ... 843
8.10.3.7. Scan Matrix ... 843
8.10.4. Operation Procedure ... 843
8.10.4.1. Serial Data Frame on the Keyboard/Mouse Interface ... 843
8.10.4.2. Keypad Interface ... 844
8.10.5. Register List ... 844
8.10.6. Register Description ... 845
8.10.6.1. Control Register ... 845
8.10.6.2. Sampling Rate Division Registers ... 846
8.10.6.3. Request to Send Counter Register ... 846
8.10.6.4. Status Register (Offset: 0x0C) ... 846
8.10.6.5. Interrupt Status Register (Offset: 0x10) ... 846
8.10.6.6. Keyboard Receive Register ... 847
8.10.6.7. Keyboard Transmit Register ... 847
8.10.6.8. Interrupt Mask Register (Offset 0x1C) ... 847
8.10.6.9. Keypad Code X Register ... 847
8.10.6.10. Keypad Code Y Register ... 847
8.10.6.11. Auto-Scan Period Register ... 848
8.10.6.12. Revision Register ... 848
8.10.6.13. Feature Register ... 848
Chapter 9. Electrical Characteristics ... 849
9.1. Absolute Maximum Ratings ... 849
9.2. Recommended Operating Conditions ... 850
9.3. DC Electrical Characteristics ... 851
9.4. SDRAM I/O DC Electrical Characteristics ... 852
9.5. Oscillator Electrical Characteristics ... 852
9.6. External Peripherals Electrical Characteristics ... 853
9.6.1. LCD AC Electrical Characteristics ... 853
9.6.2. SPI AC Electrical Characteristics ... 855
9.6.3. UART AC Electrical Characteristics ... 855
9.6.4. SCR AC Electrical Characteristics ... 860
9.7. Power-up and Power-down Sequence ... 861
Chapter 10. Package Thermal Characteristics ... 862
Chapter 11. Pin Assignment ... 863
11.1. Pin Map ... 863
11.2. Package Dimension ... 864
Chapter 12. Carrier, Storage and Baking Information ... 865
12.1. Carrier ... 865
12.1.1. Matrix Tray Information ... 865
12.2. Storage ... 867
12.2.1. Matrix Tray Information ... 867
12.2.2. Bagged Storage Conditions ... 867
12.2.3. Out-of-bag Duration ... 868
12.3. Baking ... 868
Chapter 13. Reflow Profile ... 869
Chapter 14. Part Marking ... 871

Figures

Figure 2-1. V8130L Block Diagram ... 57
Figure 4-1. Memory Space Block Diagram of CA7 ... 96
Figure 4-2. Memory Space Block Diagram of CA7 Non-Secure APB ... 97
Figure 4-3. Clock Distribution Diagram (For CA7) ... 103
Figure 4-4. Clock Distribution Diagram (For Peripheral Part 1) ... 103
Figure 4-5. Clock Distribution Diagram (For Peripheral Part 2) ... 104
Figure 4-6. Clock Distribution Diagram (For Peripheral Part 3) ... 104
Figure 4-7. Clock Distribution Diagram (eMMC/LCDC Controller Clock Generation) ... 105

Vangotech

Figure 4-8. Clock Distribution Diagram (MAC Controller Clock) 105
Figure 4-9. Power Domain Diagram of V8130L 107
Figure 4-10. Power-on Reset Timing when scu_clk is from Built-in RC Ring Oscillator 108
Figure 4-11. Power-on Reset Timing 109
Figure 4-12. Reset control diagram 109
Figure 4-13 Functional Block Diagram 115
Figure 4-14 Logic Symbol between DDRx SDRAM Controller and DDRx PHY 118
Figure 4-15. DDR PHY I/O Control Timing when Entering into Self-refresh Mode 160
Figure 4-16. DDR PHY I/O Control Timing when Exiting from Self-refresh Mode 160
Figure 4-17. tphy_wrlat and tphy_wrdata Timing in DFI Specification 2.1 169
Figure 4-18. trddata_en and tphy_rdlat in DFI Specification 2.1 169
Figure 4-19. Functional Block Diagram 243
Figure 4-20. V8130L power domain block diagram 244
Figure 4-21. Power States and Transitions 245
Figure 4-22. Power mode mapping table 245
Figure 4-23. SCU Boot Flow Chart 246
Figure 4-24. CHIPID Protected Register Lock and Unlock 250
Figure 4-25. RTC Trimming Protected Register Lock and Unlock 268
Figure 4-26. System Boot Diagram 307
Figure 4-27. Functional Block Diagram 309
Figure 4-28. PWMTMR Block Diagram 314
Figure 4-29. Example of Timer Operation 315
Figure 4-30. Example of Double Buffering Function 316
Figure 4-31. Example of PWM Function 316
Figure 4-32. Waveform of Enabled Dead-zone Feature 317
Figure 4-33. DMA Request and ACK 317
Figure 4-34. DMA Block Diagram 326
Figure 4-35. SFIFO Data Movement 328
Figure 4-36. Descriptors Stored in LDM 329
Figure 4-37. Two-level Round-robin Arbitration Mechanism 330
Figure 4-38. One-dimension Descriptor Structure 330
Figure 4-39. Two-dimension Descriptor Structure 331
Figure 4-40. Usage of Link List Pointer 331
Figure 4-41. Peripheral Interface Protocol 332
Figure 4-42. One-dimension DMA Transfer 333
Figure 4-43. Two-dimension DMA Transfer 335
Figure 4-44. DMA Controller Configuration 355
Figure 4-45. Link List Structure of Chain Transfer Operation 356
Figure 4-46. Link List Descriptor 357
Figure 4-47. Example of Interrupt Conflict during Chain Transfer Operation 358
Figure 4-48. LLP_CNT Operation 358
Figure 4-49. Example of Hardware Handshake Mode Transfer (SRC_WIDTH[2] = DST_WIDTH[3]) 360
Figure 4-50. DMA Hardware Handshake Protocol 360
Figure 4-51. ERROR Response of Transfer in Hardware Handshake Mode 361
Figure 4-52. Abort Operation with Active Channel 361
Figure 4-53. Abort Operation with Inactive Channel 361
Figure 4-54. Transfer Example in DMA Normal Mode 361
Figure 4-55. FTADCC010 Block Diagram 382
Figure 4-56. Timing Diagram of External Sampling Clock (External Sampling Mode) 383
Figure 4-57. Timing Diagram of Internal Sampling Clock (Internal Sampling Mode) 383
Figure 4-58. Timing Diagram of Scan Discharging Time 384
Figure 4-59. Timing Diagram of Channel Discharging Time when AFE_XEOC is unready 384
Figure 4-60. Timing Diagram of FTADCC010 Power Enable 385
Figure 4-61. Timing Diagram of FTADCC010 Channel Sensing 385
Figure 4-62. Timing Diagram of FTADCC010 Scan 386
Figure 4-63. FTADCC010 Single Step Conversion Mode 386
Figure 4-64. FTADCC010 Single Scan Conversion Mode 387

Vangotech

Figure 4-65. FTADCC010 Continuous Scan Conversion Mode 387
Figure 4-66. FTADCC010 Channel Conversion and DMA Function 388
Figure 4-67. FTADCC010 Threshold Function 388
Figure 4-68. FTADCC010 Auto Power-down Function 389
Figure 4-69. FTADCC010 Temperature Function 389
Figure 4-70. Guarded Area between Higher Threshold and Lower Threshold 391
Figure 4-71. Functional Block Diagram 402
Figure 4-72. Timing Diagram of Sampling Clock 403
Figure 4-73. Wake-up Timing Requirement 404
Figure 4-74. Entropy source block diagram 419
Figure 4-75. BA431 block diagram 419
Figure 4-76. CE Block Diagram 427
Figure 4-77. Secure JATG Scheme 434
Figure 4-78. Secure SPI Host Scheme 435
Figure 4-79. Secure Efuse Controller Scheme 435
Figure 5-1. Functional Block Diagram 446
Figure 5-2. Block Diagram 452
Figure 5-3. Block Diagram of ADMA 453
Figure 5-4. 32-bit Descriptor Table 454
Figure 5-5. State Diagram of ADMA 455
Figure 5-6. SD Output Timing 457
Figure 5-7. Host Controller with CPRM Function 459
Figure 5-8. Encryption in ECB Mode 459
Figure 5-9. Encryption in C-CBC Mode 460
Figure 6-1. Block Diagram 512
Figure 6-2. Example of LCDC Application 512
Figure 6-3. Write Operation in 6800 Mode 555
Figure 6-4. Read Operation in 6800 Mode 555
Figure 6-5. Write Operation in 8080 Mode 555
Figure 6-6. Read Operation in 8080 Mode 556
Figure 8-1. Functional Block Diagram 576
Figure 8-2. Functional Block Diagram 642
Figure 8-3. LCD Driver 3-line Serial Write Transaction 645
Figure 8-4. LCD Driver 4-line Serial Write Transaction 646
Figure 8-5. SPI Block Diagram 660
Figure 8-6. Write Enable 662
Figure 8-7. Page Program 662
Figure 8-8. Read Status 663
Figure 8-9. Read Data 663
Figure 8-10. Fast Read Dual Output Command 664
Figure 8-11. Fast Read Dual I/O 664
Figure 8-12. Fast Read Quad Output 665
Figure 8-13. Fast Read Quad I/O Command 665
Figure 8-14. Continuous Read Mode Reset 666
Figure 8-15. Fast Dual I/O DT Read (2DTRD) Sequence (Command BD) 666
Figure 8-16. CS Active Setup Time relative to SCK 667
Figure 8-17. CS Active Setup Time relative to SCK 667
Figure 8-18. Block Diagram 681
Figure 8-19. SPI Timing Chart 681
Figure 8-20. SPI Single Write Transfer 682
Figure 8-21. SPI Burst Write Transfer 682
Figure 8-22. SPI Single Read Transfer 683
Figure 8-23. SPI Burst Read Transfer 683
Figure 8-24. Functional Block Diagram 684
Figure 8-25. Example of Repeat-Start Sequence 686
Figure 8-26. Relationship among TSR, SCL, and SDA 698
Figure 8-27. Block Diagram of UART and IrDA Communication Controller 699
Vangotech

Figure 8-28. UART Data Representation and Sampling 702
Figure 8-29. SIR Encoding 703
Figure 8-30. Serial Infrared Interaction Pulse 705
Figure 8-31. Hardware Flow Control between Two Similar Devices 707
Figure 8-32. Interconnection between MCR and MSR in Loopback Mode 725
Figure 8-33. Block Diagram 741
Figure 8-34. Transmit Ring Descriptor Structure 744
Figure 8-35. Receive Ring Descriptor Structure 747
Figure 8-36. Block Diagram 792
Figure 8-37. Data Bypass Logic 793
Figure 8-38. Connection Diagram 793
Figure 8-39. Functional Block Diagram 801
Figure 8-40. Transmit Buffer Structure 802
Figure 8-41. Receive Buffer0's Filter and Registers 803
Figure 8-42. Receive Buffer1's Filter and Registers 803
Figure 8-43. Transmit Programming Flow Chart 804
Figure 8-44. Receive Programming Flow Chart 804
Figure 8-45. Functional Block Diagram 842
Figure 8-46. Serial Data Frame Format 843
Figure 9-1. HV_IF Interface Vertical Timing 853
Figure 9-2. HV_IF Interface Parallel Mode Horizontal Timing 854
Figure 9-3. HV_IF Interface Serial Mode Horizontal Timing 854
Figure 9-4. SPI MOSI Timing 855
Figure 9-5. SPI MISO Timing 855
Figure 9-6. UART RX Timing 856
Figure 9-7. UART nCTS Timing 856
Figure 9-8. UART nRTS Timing 856
Figure 9-9. AC-link Low Power Mode Timing 857
Figure 9-10. BIT_CLK and SYNC Timing 858
Figure 9-11. AC-link Data Transmission Output and Input Timing 858
Figure 9-12. Signal Rise and Fall Timing 858
Figure 9-13. SCR Activation and Cold Reset Timing 860
Figure 9-14. SCR Warm Reset Timing 860
Figure 9-15. V8130L Power Up Sequence 861
Figure 11-1. V8130L Pin Map 863
Figure 11-2. SCM701 Package Dimension 864
Figure 12-1. Tray Dimension Drawing 867
Figure 13-1. Typical Lead-free Reflow Profile 869
Figure 13-2. Measuring the Reflow Soldering Process 870
Figure 14-1. V8130L Marking 871

Vangotech

Tables

Table 3-1. Pin Characteristics 59
Table 3-2. GPIO Multiplex Function 64
Table 3-3. Signal Descriptions 88
Table 4-1. External Clock Sources (XTAL) 98
Table 4-2. Internal Clock Sources 98
Table 4-3. PLL Default Frequencies and Programming Values 98
Table 4-4. Clock Summary 100
Table 4-5. Power Rail Scheme 106
Table 4-6. NIC Bus 1 Security Attribution Register (Base Address = 0x1050_0000) 110
Table 4-7. NIC Bus 0 Security Attribution Register (Base Address = 0x1060_0000) 110
Table 4-8. NIC Bus 1 GPV Offset (Base Address = 0x1050_0000) 110
Table 4-9. NIC Bus 0 GPV Offset (Base Address = 0x1060_0000) 111
Table 4-10 Memory Controller Configuration Register (MCCR, Offset = 0x00) 121
Table 4-11 Memory Controller State Control Register (MCSR, Offset = 0x04) 126
Table 4-12 Mode Register Set Value Register of MR and EMR (MRSVR0, Offset = 0x08) 130
Table 4-13 Mode Register Set Value Register of EMR 2 and EMR 3 (MRSVR1, Offset = 0x0C) 131
Table 4-14 DDR RNK_TYPE Summary 131
Table 4-15 External Rank0/Rank1 Register (EXRANKR, Offset = 0x10) when FTDDR3030_LOGIC_ADDR_YES is Configured 132
Table 4-16 External Rank0/Rank1 Register (EXRANKR, Offset = 0x10) when FTDDR3030_LOGIC_ADDR_NO is Configured 134
Table 4-17 DDR1 Memory Width = 8 Bits 137
Table 4-18 DDR1 Memory Width = 16 Bits 138
Table 4-19 DDR1 Memory Width = 32 Bits 138
Table 4-20 DDR1 Memory Width = 64 Bits 138
Table 4-21 DDR2 Configuration by Different MA Tables 139
Table 4-22 DDR2 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 139
Table 4-23 DDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 139
Table 4-24 DDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 140
Table 4-25 DDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 140
Table 4-26 DDR2 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 141
Table 4-27 DDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 141
Table 4-28 DDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 141
Table 4-29 DDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 142
Table 4-30 DDR3 Configuration by Different MA Tables 142
Table 4-31 DDR3 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 142
Table 4-32 DDR3 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 143
Table 4-33 DDR3 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 143
Table 4-34 DDR3 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 143
Table 4-35 DDR3 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 144
Table 4-36 DDR3 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 144
Table 4-37 DDR3 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 144
Table 4-38 DDR3 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 145
Table 4-39 LPDDR2 Configuration by Different MA Tables 145
Table 4-40 LPDDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 146
Table 4-41 LPDDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 146
Table 4-42 LPDDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined 146
Table 4-43 LPDDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 147
Table 4-44 LPDDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 147
Table 4-45 LPDDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined 147
Table 4-46 Timing Parameter 0 Register (TMPR0, Offset = 0x14) 148
Table 4-47 Timing Parameter 1 Register (TMPR1, Offset = 0x18) 149
Table 4-48 Timing Parameter 2 Register (TMPR2, Offset = 0x1C) 153
Table 4-49 DDRx PHY Command and Data Block Control Register (PHYCR0, Offset = 0x20) 156
Table 4-50 DDRx PHY Read Path DLL Delay Tuning Register (PHYRDTR, Offset = 0x24) 160

Vantotech

Table 4-51 COMPBLK Control Register (COMPBLKCR, Offset = 0x28) 161
Table 4-52 Automatic Power-down Control Register (AODCR, Offset = 0x2C) 162
Table 4-53 Setting for Read/Write Operation 163
Table 4-54 Channel Arbitration Setup Register (CHARBRA, Offset = 0x30) 164
Table 4-55 Channel Arbiter Grant Count Register - A (CHGNTRA, Offset = 0x34) 167
Table 4-56 Channel Arbiter Grant Count Register - B (CHGNTRB, Offset = 0x38) 168
Table 4-57 DDRx PHY Write/Read Data Timing Control Register (PHYWRTMR, Offset = 0x3C) 169
Table 4-58 FXDDRII17x DDR PHY Timing Parameter 171
Table 4-59 FXDDR2A/D 200 DDR12 Combo PHY Timing Parameter 172
Table 4-60 FXDDR10x, 30x, 40x, 50x DDR23/LPDDR2 Combo PHY Timing Parameter 172
Table 4-61 Command Flush Control Register (FLUSHCR, Offset = 0x40) 173
Table 4-62 Command Flush Status Register (FLUSHSR, Offset = 0x44) 174
Table 4-63 AHB SPLIT Control Register (SPLITCR, Offset = 0x48) 175
Table 4-64 DDRx PHY and DDR3 Update Control Register (UPDCR, Offset = 0x4C) 177
Table 4-65 Revision Register (REVR, Offset = 0x50) 178
Table 4-66 Feature Register 1 (FEATR1, Offset = 0x54) 178
Table 4-67 Feature Register 2 (FEATR2, Offset = 0x58) 180
Table 4-68 User Defined Register (UDEFR, Offset = 0x5C) 181
Table 4-69 Write-leveling Control Register (WLEVELCR, Offset = 0x60) (Only for DDR3 Combo PHY) 181
Table 4-70 Write-leveling Byte 7 ~ Byte 4 Control Register (WLEVELBHR, Offset = 0x64) (Only for DDR3 Combo PHY) 183
Table 4-71 Write-leveling Byte 3 ~ Byte 0 Control Register (WLEVELBLR, Offset = 0x68) (Only for DDR3 Combo PHY) 184
Table 4-72 DDR3 PHY MISC Control Register 1 (PHYMISCR1, Offset = 0x6C) 185
Table 4-73 Read-leveling Control Register (Offset = 0x70) (RLEVELCR, Only for DDR3 Combo PHY) 187
Table 4-74 msdly Byte Control Register (MSDLYCR, Offset = 0x74) 189
Table 4-75 wrdllsel Control Register (WRDLLCR, Offset = 0x78) 190
Table 4-76 Traffic Monitor Clock Cycle Register (TRAFMR, Offset = 0x7C) 191
Table 4-77 Command Count Register for Channel 0 (CMDCNTR0, Offset = 0x80) 191
Table 4-78 Command Count Register for Channel 1 (CMDCNTR1, Offset = 0x84) 191
Table 4-79 Command Count Register for Channel 2 (CMDCNTR2, Offset = 0x88) 192
Table 4-80 Command Count Register for Channel 3 (CMDCNTR3, Offset = 0x8C) 192
Table 4-81 Command Count Register for Channel 4 (CMDCNTR4, Offset = 0x90) 192
Table 4-82 Command Count Register for Channel 5 (CMDCNTR5, Offset = 0x94) 192
Table 4-83 Command Count Register for Channel 6 (CMDCNTR6, Offset = 0x98) 193
Table 4-84 Command Count Register for Channel 7 (CMDCNTR7, Offset = 0x9C) 193
Table 4-85 AHB INCR Read Prefetch Length 1 (AHBRPRER1, Offset = 0xA0) 194
Table 4-86 AHB INCR Read Prefetch Length 2 (AHBRPRER2, Offset = 0xA4) 195
Table 4-87 Initialization of Waiting Cycle Count 1 (INITWCR1, Offset = 0xA8) 196
Table 4-88 Initialization of Waiting Cycle Count 2 (INITWCR2, Offset = 0xAC) 197
Table 4-89 Setting for QoS Command Count 197
Table 4-90 QoS Control Register (QOSCR, Offset = 0xB0) 197
Table 4-91 QoS Command Count Register A (QOSCNTRA, Offset = 0xB4) 198
Table 4-92 QoS Command Count Register B (QOSCNTRB, Offset = 0xB8) 199
Table 4-93 QoS Command Count Register C (QOSCNTRC, Offset = 0xBC) 199
Table 4-94 QoS Command Count Register D (QOSCNTRD, Offset = 0xC0) 200
Table 4-95 Channel Arbitration Setup Register B (CHARBRB, Offset = 0xC4) 201
Table 4-96 Channel Arbiter Grant Count Register - C (CHGNTRC, Offset = 0xC8) 204
Table 4-97 Channel Arbiter Grant Count Register - D (CHGNTRD, Offset = 0xCC) 205
Table 4-98 Debug Address (DBGADR, Offset = 0xD0) 206
Table 4-99 Debug Address Mask (DBGADMR, Offset = 0xD4) 206
Table 4-100 Debug Write Data (DBGWDR, Offset = 0xD8) 206
Table 4-101 Debug Write Data Mask (DBGWDMR, Offset = 0xDC) 207
Table 4-102 Debug Master Control (DBGMSTR, Offset = 0xE0) 207
Table 4-103 Debug Access Control (DBGACCR, Offset = 0xE4) 207
vangotech

Table 4-104 Debug Policy Control (DBGPCR, Offset = 0xE8) 208
Table 4-105 Debug Control (DBGENR, Offset = 0xEC) 208
Table 4-106 Debug Address Status (DBGADSR, Offset = 0xF0) 209
Table 4-107 Debug Write Data Status (DBGWDSR, Offset = 0xF4) 209
Table 4-108 Debug Master Status (DBGMSTSR, Offset = 0xF8) 209
Table 4-109 Debug Access Status (DBGACCSR, Offset = 0xFC) 209
Table 4-110 LPDDR2 Mode Register MR1/MR2/MR3 (LP2MRA, Offset = 0x100) 209
Table 4-111 LPDDR2 Mode Register MR9/MR10 (LP2MRB, Offset = 0x104) 210
Table 4-112 LPDDR2 Mode Register MR16/MR17 (LP2MRC, Offset = 0x108) 210
Table 4-113 LPDDR2 Mode Register MR41/MR42/MR48 (LP2MRD, Offset = 0x10C) 210
Table 4-114 LPDDR2 Mode Register MR63 (LP2MRE, Offset = 0x110) 210
Table 4-115 LPDDR2 MRW and MRR Command Register (LP2MRCR, Offset = 0x114) 211
Table 4-116 LPDDR2 MRR Read Value Register (LP2MRVR, Offset = 0x118) 212
Table 4-117 LPDDR2 Write Latency and Read Latency Register (LP2ADLR, Offset = 0x11C) 212
Table 4-118 LPDDR2 1-μs Wait Cycle Register (LP2WCR1, Offset = 0x120) 213
Table 4-119 LPDDR2 10-μs Wait Cycle Register (LP2WCR2, Offset = 0x124) 213
Table 4-120 Re-arbitration Disable Register (REARBDISR, Offset = 0x12C) 213
Table 4-121 DDR PHY Read Path DLL Delay Tuning for Falling Edge Register (PHYRDTFR, Offset = 0x130) 216
Table 4-122 DDR PHY MISC2 Register (PHYMISCR2, Offset = 0x134) 217
Table 4-123 DDR ELASTIC FIFO Control Register (EFIFOCR, Offset = 0x138) 219
Table 4-124 Byte n Read Deskew Register (RBnDSKW, Offset = 0x160+4 * n, n = 0 ~ 7) 220
Table 4-125 Byte n Write Deskew Register (WBnDSKW, Offset = 0x180+4*n, n = 0 ~ 7) 220
Table 4-126 Write Data Mask Deskew Register (WDMDSKW, Offset = 1A0) 221
Table 4-127 Byte 8 Read Deskew Register (RB8DSKW, Offset = 0x1A4) 222
Table 4-128 Byte 8 Write Deskew Register (WB8DSKW, Offset = 0x1A8) 222
Table 4-129 Byte8 PHY Control Register (B8_PHYCR, Offset = 0x1AC) 223
Table 4-130 BIST Control Register (BISTCR, Offset = 0x01B0) 225
Table 4-131 BIST Start Address Register (BIST_START_ADDR, Offset = 0x01B4) 226
Table 4-132 BIST Test Size Register (BIST_Test_Size, Offset = 0x01B8) 226
Table 4-133 BIST Result Register (BIST_RESULT, Offset = 0x01C0) 227
Table 4-134 BIST Fail Address Register (BIST_FAIL_ADDR, Offset = 0x01C4) 227
Table 4-135 BIST Fail Data Low Register (BIST_FAIL_DATA_LOW, Offset = 0x01C8) 228
Table 4-136 BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x01CC) 228
Table 4-137 BIST Expect Data Low Register (BIST_EXP_DATA_LOW, Offset = 0x01D0) 228
Table 4-138 BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x01D4) 229
Table 4-139 ECC Control Register (ECC_CTRLR, Offset = 0x01E0) 229
Table 4-140 ECC Interrupt Control Register (ECC_INTCR, Offset = 0x01E4) 230
Table 4-141 ECC Error Channel Register (ECC_CHR, Offset = 0x01E8) 231
Table 4-142 ECC Error Address Register (ECC_EAR, Offset = 0x01EC) 231
Table 4-143 ECC Error Bit Position Low Byte Register (ECC_EBPLR, Offset = 0x01F0) 232
Table 4-144 ECC Error Bit Position High Byte Register (ECC_EBPHR, Offset = 0x01F4) 232
Table 4-145 ECC One-bit Error Count Register (ECC_ERR1_COUNT, Offset = 0x01F8) 233
Table 4-146 ECC Two-bit Error Count Register (ECC_ERR2_COUNT, Offset = 0x01FC) 233
Table 4-147 AHB/AXI Address for DDR Address Mapping when FTDDR3030_LOGIC_ADDR_YES is Configured 235
Table 4-148 AHB/AXI Address for DDR Address Mapping when FTDDR3030_LOGIC_ADDR_NO is Configured 235
Table 4-149 MA Table1, AMTSEL = 0 235
Table 4-150 MA Table1, AMTSEL = 1 235
Table 4-151 MA Table1, AMTSEL = 2 236
Table 4-152 MA Table2, AMTSEL = 0 236
Table 4-153 MA Table2, AMTSEL = 1 237
Table 4-154 MA Table2, AMTSEL = 2 237
Table 4-155 MA Table3, AMTSEL = 0 238
Table 4-156 MA Table3, AMTSEL = 1 238
Table 4-157 MA Table3, AMTSEL = 2 238
vangotech

Table 4-158 MA Table4, AMTSEL = 0 239
Table 4-159 MA Table4, AMTSEL = 1 239
Table 4-160 MA Table4, AMTSEL = 2 240
Table 4-161 Boot-up Status Register (BTUP_STS, Offset: 0x000) 247
Table 4-162 Boot-up Control Register (BTUP_CTRL, Offset: 0x004) 248
Table 4-163. Power Control Register (PWR_CTRL, Offset: 0x008) 249
Table 4-164. *Power Control Register (*PWRUP_SEQ*, Offset: 0x00C)* 250
Table 4-165. Chip ID Register (CHIPID, Offset: 0x010) 250
Table 4-166. System Control Unit Version Register (VERID, Offset: 0x014) 251
Table 4-167. Strap Value Register (STRAP, Offset: 0x018) 251
Table 4-168. OSC Control Register (OSC_CTRL, Offset: 0x01C) 251
Table 4-169. Power Mode Register, (PWR_MOD, Offset: 0x020) 252
Table 4-170. Interrupt Status Register (INT_STS, Offset: 0x024) 253
Table 4-171. Interrupt Enable Register (INT_EN, Offset: 0x028) 254
Table 4-172. Software Reset Control Register (SWRST_CTRL, Offset: 0x02C) 255
Table 4-173. PLL0 Control Register (PLL_CRTL, Offset: 0x030) 255
Table 4-174. PLL4 Control Register (PLL4_CTRL, Offset: 0x040) 256
Table 4-175. DLL Control Register (DLL_CTRL, Offset: 0x044) 256
Table 4-176. AHB Clock Control Register0 (AHBCLKG0, Offset: 0x050) 256
Table 4-177. AHB Clock Sleep Control Register0 (SLP_AHBCLKG0, Offset: 0x058) 257
Table 4-178. APB Clock Control Register (APBCLKG0, Offset: 0x060) 257
Table 4-179. APB Clock Control Register (APBCLKG1, Offset: 0x0000_0064) 258
Table 4-180. APB Clock Sleep Control Register (SLP_APBCLKG0, Offset: 0x068) 258
Table 4-181. APB Clock Sleep Control Register (SLP_APBCLKG1, Offset: 0x06C) 259
Table 4-182. AXI Clock Control Register0 (AXICLKG, Offset: 0x080) 260
Table 4-183. AXI Clock Sleep Control Register0 (AXICLK_SLPCTRL, Offset: 0x088) 260
Table 4-184. Multi-Core WFI/WFE Control Register (0x0B4) 261
Table 4-185. Sleep Wakeup Status Register (SLP_WAKUP_ST, Offset: 0x0C0) 261
Table 4-186. Sleep Wakeup Enable Register (SLP_WAKUP_EN, Offset: 0x0C4) 262
Table 4-187. IDLE Control Register(IDLE_CTRL, Offset: 0x128) 262
Table 4-188. RTC Time1 Register (RTC_TIME1, Offset: 0x200) 263
Table 4-189. RTC Time2 Register (RTC_TIME2, Offset: 0x204) 264
Table 4-190. RTC Alarm Time1 Register (RTC_ALM1, Offset: 0x208) 265
Table 4-191. RTC Alarm Time2 Register (RTC_ALM2, Offset: 0x20C) 265
Table 4-192. RTC Control Register (RTC_CTRL, Offset: 0x210) 266
Table 4-193. RTC Tick Trim Control Register (RTC_TRIM, Offset: 0x214) 267
Table 4-194. RTC Time1 BCD Register (BCD_RTC_TIME1, Offset: 0x220) 268
Table 4-195. RTC Time2 BCD Register (BCD_RTC_TIME2, Offset: 0x224) 269
Table 4-196. RTC Alarm Time1 BCD Register (BCD_RTC_ALM1, Offset: 0x228) 269
Table 4-197. RTC Alarm Time2 BCD Register (BCD_RTC_ALM2, Offset: 0x22C) 270
Table 4-198. Extension register offsets for V8130L. 271
Table 4-199. CA7 Configuration 1 Register (Offset: 0x0000_8000) 278
Table 4-200. CA7 Configuration 2 Register (Offset: 0x0000_8004) 278
Table 4-201. CA7 Configuration 3 Register (Offset: 0x0000_8008) 278
Table 4-202. CA7 Configuration 4 Register (Offset: 0x0000_800C) 278
Table 4-203. CA7 Configuration 5 Register (Offset: 0x0000_8010) 279
Table 4-204. CA7 Configuration 6 Register (Offset: 0x0000_8014) 279
Table 4-205. CA7's SPI Host Magic Number Register (Offset: 0x0000_801C) 279
Table 4-206. PLL Lock Status Register (Offset: 0x0000_8020) 279
Table 4-207. DDR CTRL 0 Register (Offset: 0x0000_8024) 280
Table 4-208. DDR CTRL 1 Register (Offset: 0x0000_8028) 281
Table 4-209. DDR CTRL 2 Register (Offset: 0x0000_802C) 282
Table 4-210. SBS_OTG CTRL Register (Offset: 0x0000_8030) 282
Table 4-211. SBS_OTG_1 CTRL Register (Offset: 0x0000_8034) 284
Table 4-212. PLL1 Control Register (Offset: 0x0000_8040) 286
Table 4-213. PLL1 Lock Timer Register (Offset: 0x0000_8044) 287
Table 4-214. PLL2 Control Register (Offset: 0x0000_8048) 287
Vangotech

Table 4-215. PLL2 Lock Timer Register (Offset: 0x0000_804C) 287
Table 4-216. PLL3 Control Register (Offset: 0x0000_8050) 287
Table 4-217. PLL3 Lock Timer Register (Offset: 0x0000_8054) 288
Table 4-218. PLL5 Control Register (Offset: 0x0000_8060) 288
Table 4-219. PLL5 Lock Timer Register (Offset: 0x0000_8064) 288
Table 4-220. DLL0 Control Register (Offset: 0x0000_8068) 288
Table 4-221. DLL1 Control Register (Offset: 0x0000_806C) 289
Table 4-222. Clock Enable 0 Register (Offset: 0x0000_8070) 289
Table 4-223. Clock Enable 1 Register (Offset: 0x0000_8074) 290
Table 4-224. Clock Enable 2 Register (Offset: 0x0000_8078) 291
Table 4-225. Clock Mux Selection Register (Offset: 0x0000_807C) 291
Table 4-226. Clock Divider 1 Register (Offset: 0x0000_8080) 292
Table 4-227. Clock Divider 2 Register (Offset: 0x0000_8084) 292
Table 4-228. Clock Divider 3 Register (Offset: 0x0000_8088) 293
Table 4-229. Clock Divider 4 Register (Offset: 0x0000_808C) 293
Table 4-230. MISC. Register (Offset: 0x0000_8090) 293
Table 4-231. Software Reset Mask 0 (swrst_msk0) Register (Offset: 0x0000_8094) 294
Table 4-232. Software Reset Mask 1 (swrst_msk1) Register (Offset: 0x0000_8098) 295
Table 4-233. Software Reset Mask 2 (swrst_msk2) Register (Offset: 0x0000_809C) 295
Table 4-234. Software Reset Mask 3 (swrst_msk3) Register (Offset: 0x0000_80A0) 296
Table 4-235. Software Reset Register (Offset: 0x0000_80A4) 297
Table 4-236. PLL6 Control Register (Offset: 0x0000_80C8) 297
Table 4-237. PLL6 Lock Timer Register (Offset: 0x0000_80CC) 297
Table 4-238. Clock Divider 5 Register (Offset: 0x0000_80D0) 297
Table 4-239. Efuse Read Password Register (Offset: 0x0000_80D4) 298
Table 4-240. WDT Tickle Feature Register (Offset: 0x0000_80D8) 298
Table 4-241. EFUSE Feature Read Register (Offset: 0x0000_80DC) 298
Table 4-242. EFUSE Feature Read Register (Offset: 0x0000_80E0) 298
Table 4-243. SBS_OTG_2 CTRL Register (Offset: 0x0000_80E4) 299
Table 4-244. Clock Divider 6 Register (Offset: 0x0000_80E8) 301
Table 4-245. GPIO0 bps Enable Register (Offset: 0x0000_80EC) 301
Table 4-246. GPIO0 bps Data Register (Offset: 0x0000_80F0) 301
Table 4-247. GPIO1 bps Enable Register (Offset: 0x0000_80F4) 301
Table 4-248. GPIO1 bps Data Register (Offset: 0x0000_80F8) 302
Table 4-249. RGMII_gtx_clk_delay Register (Offset: 0x0000_8100) 302
Table 4-250. RGMII_rx_clk_delay Register (Offset: 0x0000_8104) 302
Table 4-251. TZC Configure Register (Offset: 0x0000_8108) 302
Table 4-252. SPI020 Configure Register (Offset: 0x0000_810C) 303
Table 4-253. SSP_SPI MUX SEL Register (Offset: 0x0000_8110) 303
Table 4-254. PLL7 Control Register (Offset: 0x0000_8114) 303
Table 4-255. PLL7 Lock Timer Register (Offset: 0x0000_8118) 304
Table 4-256. Pin Mux with Mode X (Offset: 0x0000_8204) 304
Table 4-257. *IO Control Register (Offset: 0x0000_8500~0x0000_87F4)* 305
Table 4-258. SYS_XTAL Control Register (Offset: 0x0000_87F4) 306
Table 4-259. Key Register 310
Table 4-260. Prescaler Register 310
Table 4-261. Reload Register 311
Table 4-262. Status Register 311
Table 4-263. Interrupt Status Register 311
Table 4-264. Control Register 312
Table 4-265. Interrupt Length Register 313
Table 4-266. Reload Register 313
Table 4-267. WdRevision Register 313
Table 4-268.Control Register (Offset: 0x00) 318
Table 4-269. Timer Control Register (TMX_CTRL (X = 1, 2, 3, 4), Offset = 0xX0) 319
Table 4-270. Timer Counting Value Buffer Register (TMX_CNTB (X = 1, 2, 3, 4), Offset = 0xX4) 320
Vangotech

Table 4-271. Timer Counter Compare Buffer Register (TMX_CMPB (X = 1, 2, 3, 4), Offset = 0xX8) 320
Table 4-272. Timer Count Observation Register (TMX_CNTO (X = 1, 2,..., 8), Offset = 0xXC) 320
Table 4-273. Revision Register (TMR_REV, Offset = 0x90) 320
Table 4-274. Internal Interrupt Vectors for CA7 321
Table 4-275. Relationship between AXI ID and DMA Channels 327
Table 4-276. Descriptor Content of One-dimension DMA Transfer 332
Table 4-277. Control Register of One-dimension DMA Transfer 333
Table 4-278. Descriptor Content of Two-dimension DMA Transfer 333
Table 4-279. Control Register of Two-dimension DMA Transfer 334
Table 4-280. Interrupt Register (INT, Offset = 0x00) 339
Table 4-281. Terminal Count Interrupt Register (TCINT, Offset = 0x04) 339
Table 4-282. Terminal Count Interrupt Clear Register (TCINTC, Offset = 0x08) 339
Table 4-283. Error/Abort Interrupt Register (EAINT, Offset = 0x0C) 340
Table 4-284. Error/Abort Interrupt Clear Register (EAINTC, Offset = 0x10) 340
Table 4-285. Terminal Count Status Register (TCS, Offset = 0x14) 340
Table 4-286. Error/Abort Interrupt Register (EAS, Offset = 0x18) 340
Table 4-287. Channel Enable Register (CES, Offset = 0x1C) 340
Table 4-288. Synchronization of Peripheral Interface Register (SPI, Offset = 0x20) 341
Table 4-289. Local Descriptor Memory Base Register (LDMBA, Offset = 0x24) 341
Table 4-290. Watchdog Timer Register (WDT, Offset = 0x28) 342
Table 4-291. Global Event Register (GER, Offset = 0x2C) 342
Table 4-292. PSlvErr Disable Register (PSE, Offset = 0x30) 343
Table 4-293. Revision Number Register (REV, Offset = 0x34) 343
Table 4-294. Hardware Feature Register (FEA, Offset = 0x38) 343
Table 4-295. Hardware Feature Register 2 (FEA2, Offset = 0x54) 344
Table 4-296. Local Descriptor Memory Free Flag Set 0 Register (LMFFS0, Offset = 0x3C) 345
Table 4-297. Local Descriptor Memory Free Flag Set 1 Register* (LMFFS1, Offset = 0x40) 345
Table 4-298. Local Descriptor Memory Free Flag Set 2 Register* (LMFFS2, Offset = 0x44) 345
Table 4-299. Local Descriptor Memory Free Flag Set 3 Register* (LMFFS3, Offset = 0x48) 345
Table 4-300. Endian Conversion Register (EDNC, Offset = 0x4C) 345
Table 4-301. Write-only Mode Constant Value Register (CVWOM, Offset = 0x50) 346
Table 4-302. DMA Channel N Control Register (CnCTL, Offset = 0x100 + 0x20 * N) 346
Table 4-303. DMA Channel N Configuration Register (CnCFG, Offset = 0x0104 + 0x20 * N) 349
Table 4-304. DMA Channel N Source Address Register (CnSRCA, Offset = 0x108 + 0x20 * N) 351
Table 4-305. DMA Channel N Destination Address Register (CnDSTA, Offset = 0x10C + 0x20 * N) 351
Table 4-306. DMA Channel N Link List Pointer Register (CnLNK, Offset = 0x110 + 0x20 * N) 351
Table 4-307. DMA Channel N Transfer Size Register (CnTSIZE, Offset = 0x114 + 0x20 * N, when 2DEn is not set.) 352
Table 4-308. DMA Channel N Transfer Size Register (CnTSIZE, Offset = 0x114 + 0x2 0 *N, when 2DEn is set.) 352
Table 4-309. DMA Channel N Source/Destination Stride Register (CnSSDA, Offset = 0x118 + 0x20 * N) 352
Table 4-310. DMA Channel N Source Address and Destination Address Expand Register (CnSADAEXP, Offset = 0x200 + 0x20 * N) 353
Table 4-311. DMA Channel N Link List Pointer and User-defined Signal Expand Register (CnLLPUSEREXP, Offset = 0x204 + 0x20 * N) 354
Table 4-312. Address Map for Link List Descriptor (Base Address: Cn_LLP[31:2]) 357
Table 4-313. Control Field Definition in Link List Descriptor 357
Table 4-314. Total Transfer Size Definition in Link List Descriptor 357
Table 4-315. Interrupt Status Register (INT, Offset = 0x00) 363
Table 4-316. Terminal Count Interrupt Status Register (INT_TC, Offset = 0x04) 363
Table 4-317. Terminal Count Interrupt Status Clear Register (INT_TC_CLR, Offset = 0x08) 364
Table 4-318. Error/Abort Interrupt Status Register (INT_ERR/ABT, Offset = 0x0C) 365
Table 4-319. Error/Abort Interrupt Status Clear Register (INT_ERR/ABT_CLR, Offset = 0x10) 367
Table 4-320. Terminal Count Status Register (TC, Offset = 0x14) 368

Vangotech

Table 4-321. Error/Abort Status Register (ERR/ABT, Offset = 0x18) 369
Table 4-322. Channel Enable Status Register (CH_EN, Offset = 0x1C) 370
Table 4-323. Channel Busy Status Register (CH_BUSY, Offset = 0x20) 371
Table 4-324. Main Configuration Status Register (CSR, Offset = 0x24) 372
Table 4-325. Synchronization Register (SYNC, Offset = 0x28) 372
Table 4-326. FTDMAC020 Revision Register (DMAC_REVISION, Offset = 0x30) 373
Table 4-327. FTDMAC020 Feature Register (DMAC_FEATURE, Offset = 0x34) 374
Table 4-328. Channel Control Register (Cn_CSR, Offset = 0x100 + 0x20 * n, n = 0 ~ 7) 374
Table 4-329. Channel Configuration Register (Cn_CFG, Offset = 0x104 + 0x20 * n, n = 0 ~ 7) 379
Table 4-330. Channel Source Address Register (Cn_SrcAdd, Offset = 0x108 + 0x20 * n, n = 0 ~ 7) 381
Table 4-331. Channel Destination Address Register (Cn_DstAddr, Offset = 0x10C + 0x20 * n, n = 0 ~ 7) 381
Table 4-332. Link List Descriptor Pointer (Cn_LLP, Offset = 0x110 + 0x20 * n, n = 0 ~ 7) 381
Table 4-333. Transfer Size Register (Cn_SIZE, Offset = 0x114 + 0x20 * n, n = 0 ~ 7) 381
Table 4-334. Sensing Data Register (Offset: 0x000 ~ 0x01F) 390
Table 4-335. ADC Threshold Detect Programming Registers (Offset: 0x080 ~ 0x09F) 391
Table 4-336. Temperature Data Register (Offset: 0x0C0) 392
Table 4-337. Temperature Threshold Detect Programming Register (Offset: 0x0C4) 392
Table 4-338. ADC Control Register (ADC_CTRL, Offset: 0x100) 393
Table 4-339. ADC Miscellaneous Control Register (Offset: 0x104) 394
Table 4-340. ADC Interrupt Enable Register (Offset: 0x108) 395
Table 4-341. ADC Interrupt Status Register (Offset: 0x10C) 395
Table 4-342. ADC Timing Parameter Register (Offset: 0x0110) 397
Table 4-343. ADC Sampling Timing Parameter Control Register (Offset: 0x114) 397
Table 4-344. ADC Sensing Discharge Time Register (Offset: 0x118) 398
Table 4-345. ADC Clock Pre-scaler Control Register (Offset: 0x11C) 399
Table 4-346. ADC Scan Sequence Register (Offset: 0x120) 399
Table 4-347. ADC DMAn Data Port Register (Offset: 0x200, 0x210, 0x220, and 0x230) 399
Table 4-348. ADC DMAn Function Control Register (Offset: 0x204, 0x214, 0x224, and 0x234) 400
Table 4-349. ADC DMAn Interrupt Enable Register (Offset: 0x208, 0x218, 0x228, and 0x238) 400
Table 4-350. ADC DMAn Interrupt Status Register (Offset: 0x20c, 0x21c, 0x22c, and 0x23c) 400
Table 4-351. Electrical Characteristics 402
Table 4-352. Timing Parameters of Sampling Clock 403
Table 4-353. Wake-up Timing Specifications 404
Table 4-354. Trimming Table (1) 405
Table 4-355. Trimming Table (2) 405
Table 4-356. Relationship between Temperature & Output of TDC and Data Format 405
Table 4-357. TDC sensing data registers (TDC_DATA0, offset: 0x000) 406
Table 4-358. Threshold registers (TDC_THRHOLD0, offset: 0x080) 406
Table 4-359. TDC control register (TDC_CTRL, offset: 0x100) 407
Table 4-360. Trim register (TDC_TRIM, offset: 0x104) 407
Table 4-361. Interrupt enable register (TDC_INTEN, offset: 0x108) 407
Table 4-362. Interrupt status register (TDC_INTST, offset: 0x10c) 407
Table 4-363. Timing Parameter 0 control register (TDC_TPARAM0, offset: 0x0110) 408
Table 4-364. Timing Parameter 1 control register (TDC_TPARAM1, offset: 0x114) 408
Table 4-365. TDC clock pre-scaler register (TDC_PRESCAL, offset: 0x11c) 408
Table 4-366. IP revision register (IP_REVISION , offset: 0x13c) 408
Table 4-367. eFuse data register (Offset: 0x000~0xFFF.) 409
Table 4-368. electronic signature 409
Table 4-369. Timing parameter 1 register (Offset: 0x1000) 410
Table 4-370. Timing parameter 2 register (Offset: 0x1004) 410
Table 4-371. eFuse control and timing parameter 3 register (0x1008) 410
Table 4-372. Page data status register (Offset: 0x100c) 411
Table 4-373. Protect key and status register (Offset: CTRL_REG_BASE+0x10) For write operation. 411
vangotech

Table 4-374. Protect key and status register (Offset: CTRL_REG_BASE+0x10) For read operation. 412
Table 4-375. Unlock register (Offset: CTRL_REG_BASE+0x14) 412
Table 4-376. Extension Register Summary Table (Base Address = 0x2800_0000) 413
Table 4-377. IO Control Register (Offset: 0x0000_8500~0x0000_87F4) 418
Table 4-378 Register map 419
Table 4-379 Control register 420
Table 4-380 FifoLevel register 421
Table 4-381 FIFOThresh register 421
Table 4-382 FIFODepth register 422
Table 4-383 Key register 422
Table 4-384 Threshold registers 422
Table 4-385 Control register 423
Table 4-386 InitWaitVal register 423
Table 4-387 DisableOsc0 register 424
Table 4-388 DisableOsc1 register 424
Table 4-389 SwOffTmrVal register 424
Table 4-390 ClkDiv register 424
Table 4-391 AIS31 configuration register 0 424
Table 4-392 AIS31 configuration register 1 424
Table 4-393 AIS31 configuration register 2 425
Table 4-394 AIS31Status register 425
Table 4-395 HwConf register 425
Table 4-396. Fetcher / pusher registers 429
Table 4-397. Interrupt sources 430
Table 4-398. Interrupt registers 431
Table 4-399. Control/status register 432
Table 4-400. Descriptor (32-bit address) 433
Table 4-401. Descriptor (64-bit address) 433
Table 4-402. Internal Interrupt Vectors for CA7 437
Table 5-1. Register Summary Table 447
Table 5-2. ECC Error Address Register (EEAR, Offset = 0x0) 447
Table 5-3. Memory Control Register (MCR, Offset = 0x4) 448
Table 5-4. APB Controller Register (PCR, Offset = 0x8) 450
Table 5-5. IP Feature Register (FEAR, Offset = 0xF8) 450
Table 5-6. IP Revision Register (IVR, Offset = 0xFC) 450
Table 5-7. ADMA Data Length Field 454
Table 5-8. ADMA States 455
Table 5-9. Timing Adjustment of SD Interface Output 456
Table 5-10. C2 Cipher Characteristics 459
Table 5-11. SDMA System Address Register (SDMADR, Offset: 0x00) 462
Table 5-12. Block Size Register (BSR, Offset: 0x04) 462
Table 5-13. Block Count Register (BCR, Offset: 0x06) 463
Table 5-14. Argument 1 Register (ARG1, Offset: 0x08) 464
Table 5-15. Transfer Mode Register (TMR, Offset: 0x0C) 464
Table 5-16. Transfer Command Setting 465
Table 5-17. Command Register (CR, Offset: 0x0E) 466
Table 5-18. Relationship between Parameters and Names of Response Types 467
Table 5-19. Response Register 0 ~ 3 (RESPn, Offset: 0x10 ~ 0x1C, n = 0 ~ 3) 468
Table 5-20. Response Bit Definition of Each Response Type 468
Table 5-21. Buffer Data Port Register (BDP, Offset: 0x20) 468
Table 5-22. Present State Register (PSR, Offset: 0x24) 468
Table 5-23. Host Control 1 Register (HC1, Offset: 0x28) 472
Table 5-24. Power Control Register (PCR, Offset: 0x29) 474
Table 5-25. Block Gap Control Register (BGCR, Offset: 0x2A) 474
Table 5-26. Clock Control Register (CCR, Offset: 0x2C) 476
Table 5-27. Timeout Control Register (TCR, Offset: 0x2E) 477

vangotech

Table 5-28. Software Reset Register (SRR, Offset: 0x2F) 477
Table 5-29. Normal Interrupt Status Register (NISR, Offset: 0x30) 478
Table 5-30. Error Interrupt Status Register (EISR, Offset: 0x32) 480
Table 5-31. Normal Interrupt Status Enable Register (NISER, Offset: 0x34) 482
Table 5-32. Error Interrupt Status Enable Register (EISER, Offset: 0x36) 483
Table 5-33. Normal Interrupt Signal Enable Register (NISEN, Offset: 0x38) 484
Table 5-34. Error Interrupt Signal Enable Register (EISEN, Offset: 0x3A) 484
Table 5-35. Auto CMD12 Error Status Register (AC12ES, Offset: 0x3C) 485
Table 5-36. Relationship between CRC Error and Timeout Error for Auto CMD 485
Table 5-37. Host Control 2 Register (HC2, Offset: 0x3E) 486
Table 5-38. Capabilities 0/1 Register (CAP, Offset: 0x40) 488
Table 5-39. Maximum Current Capabilities Register (MCC, Offset: 0x48) 492
Table 5-40. Definitions of Maximum Current Value 492
Table 5-41. Force Event Register for Auto CMD Error Status (FACERR, Offset: 0x50) 493
Table 5-42. Force Event Register for Error Interrupt Status (FERR, Offset: 0x52) 493
Table 5-43. ADMA Error Status Register (AESR, Offset: 0x54) 494
Table 5-44. ADMA System Address Register (ALSAR, Offset: 0x58) 495
Table 5-45. Preset Value Registers (PVn, Offset: 0x60 ~ 0x6F, n = 0 ~ 3) 495
Table 5-46. Speed Mode Selection 495
Table 5-47. Preset Value Register Based on Speed Mode 496
Table 5-48. Host Controller Version Register (HCVR, Offset: 0XFE) 496
Table 5-49. Vendor-defined Register 0 (VR0, Offset: 0X100) 497
Table 5-50. Vendor-defined Register 1 (VR1, Offset: 0X104) 498
Table 5-51. Vendor-defined Register 2 (VR2, Offset: 0X108) 499
Table 5-52. Vendor-defined Register 3 (VR3, Offset: 0X10C) 499
Table 5-53. Vendor-defined Register 4 (VR4, Offset: 0X110) 500
Table 5-54. Vendor-defined Register 5 (VR5, Offset: 0X114) 501
Table 5-55. Vendor-defined Register 6 (VR6, Offset: 0X118) 501
Table 5-56. Vendor-defined Register 7 (VR7, Offset: 0X11C) 501
Table 5-57. Vendor-defined Register 8 (VR8, Offset: 0X120) 502
Table 5-58. Vendor-defined Register 9 (VR9, Offset: 0X124) 502
Table 5-59. DMA Handshake Enable Register (DHER, Offset: 0X128) 503
Table 5-60. Hardware Attributes Register (HWA, Offset: 0X178) 504
Table 5-61. IP Revision Register (IPRR, Offset: 0X17C) 504
Table 5-62. Cipher Mode Control Register (CMCR, Offset: 0x180) 504
Table 5-63. Cipher Mode Status Register (CMSR, Offset: 0x184) 505
Table 5-64. Cipher Mode Signal Enable Register (CMSE, Offset: 0x188) 506
Table 5-65. Low Word of Input Data Register (LWID, Offset: 0x18C) 506
Table 5-66. High Word of Input Data Register (HWID, Offset: 0x190) 506
Table 5-67. Low Word of Input Key Register (LWIK, Offset: 0x194) 506
Table 5-68. High Word of Input Key Register (HWIK, Offset: 0x198) 506
Table 5-69. Low Word of Output Data Register (LWOD, Offset: 0x19C) 506
Table 5-70. High Word of Output Data Register (HWOD, Offset: 0x1A0) 507
Table 5-71. Secret Constant Table Data Port (SCTDP, Offset: 0x1A4) 507
Table 6-1. LCD Function Enable Parameter (lcd_function_en, Offset 0x0000) 514
Table 6-2. LCD Panel Pixel Parameter (lcd_panel_pixel, Offset 0x0004) 517
Table 6-3. LCD Interrupt Enable Mask Parameter (lcd_intr_mask, Offset 0x0008) 519
Table 6-4. LCD Interrupt Status Clear Parameter(lcd_intr_clr, Offset 0x000C) 519
Table 6-5. LCD Interrupt Status Parameter(lcd_intr_status, Offset 0x0010) 520
Table 6-6. Frame Buffer Parameter (lcd_frame_buffer, Offset 0x0014) 520
Table 6-7. Image0 Frame0 Base Address (lcd_im0frm0base, Offset 0x0018) 521
Table 6-8. Image1 Frame0 Base Address(lcd_im1frm0base, Offset 0x0024) 522
Table 6-9. Image2 Frame0 Base Address(lcd_im2frm0base, Offset 0x0030) 522
Table 6-10. Image3 Frame0 Base Address(lcd_im3frm0base, Offset 0x003C) 522
Table 6-11. PatGen Pattern Bar Distance Parameter (lcd_patbarcolor Offset 0x0048) 523
Table 6-12. FIFO Threshold Control Parameter (lcd_fifoth, Offset 0x004C) 523
Table 6-13. Bus Bandwidth Control Parameter (lcd_bandwithCtrl, Offset 0x0050) 524
Vango Technologies

Table 6-14. Revision Register(lcd_revision, Offset: 0x58) 525
Table 6-15. VBI Base Address (lcd_vbibase, Offset 0x005C) 525
Table 6-16. FIFO Threshold Control Parameter (lcd_fifo4th, Offset 0x0060) 526
Table 6-17. LCD Horizontal Timing Control Parameter (lcd_hortiming, Offset 0x0100) 526
Table 6-18. LCD Vertical Timing Control Parameter (lcd_vertiming0, Offset 0x0104) 527
Table 6-19. LCD Vertical Back Porch Parameter (lcd_vertiming1,Offset 0x0108) 528
Table 6-20. LCD Polarity Control Parameter(lcd_polarity, Offset 0x010C) 528
Table 6-21. PiP Blending Parameter(lcd_pipblend, Offset 0x0300) 529
Table 6-22. PiP Sub-Picture1 Position Parameter(lcd_pip1pos, Offset 0x0304) 530
Table 6-23. PiP Sub-Picture1 Dimension Parameter (lcd_pip1dim, Offset 0x0308) 530
Table 6-24. PiP Sub-Picture2 Position Parameter(lcd_pip2pos, Offset 0x030C) 531
Table 6-25. PiP Sub-Picture2 Dimension Parameter(lcd_pip2dim, Offset 0x0310) 531
Table 6-26. PiP Image Priority Parameter(lcd_pipriority, Offset 0x0314) 532
Table 6-27. PiP/PoP Image Format 1 Parameter(lcd_param1,Offset 0x0318) 533
Table 6-28. PiP/PoP Image Format 2 Parameter(lcd_param2, Offset 0x031C) 535
Table 6-29. PiP Color Key1(lcd_colorkey1, Offset 0x0320) 537
Table 6-30. PiP Color Key2(lcd_colorkey2, Offset 0x0324) 537
Table 6-31. PiP Color Key3 (lcd_colorkey3, Offset 0x0328) 537
Table 6-32. PiP Sub-Picture3 Position(lcd_pip3pos, Offset 0x032C) 537
Table 6-33. PiP Sub-Picture3 Dimension Parameter (lcd_pip3dim, Offset 0x0330) 538
Table 6-34. Palette RAM(PaletteWritePort, Offset 0x0A00 ~ 0x0BFC) 538
Table 6-35. LCD Color Management Parament0 Parameter(lcd_colorman0, Offset 0x0400) 539
Table 6-36. LCD Color Management Parament1 Parameter(lcd_colorman1, Offset 0x0404) 539
Table 6-37. LCD Color Management Parament2 Parameter(lcd_colorman2, Offset 0x0408) 540
Table 6-38. LCD Color Management Parament3 Parameter(lcd_colorman3, Offset 0x040C) 540
Table 6-39. Lookup Table of LCD Gamma Red(lcd_GaRdLUT, Offset 0x0600 ~ 0x06FC) 540
Table 6-40. Lookup Table of LCD Gamma Green(lcd_GaGnLUT, Offset 0x0700 ~ 0x07FC) 541
Table 6-41. Lookup Table of LCD Gamma Blue (lcd_GaBuLUT, Offset 0x0800 ~ 0x08FC) 542
Table 6-42. Horizontal Resolution Register of Scaler Input(scal_hor_no_in, Offset: 0x1100) 542
Table 6-43. Vertical Resolution Register of Scaler Input(scal_ver_no_in, Offset 0x1104) 542
Table 6-44. Horizontal Resolution Register of Scaler Output(scal_hor_no_out, Offset 0x 1108) 543
Table 6-45. Vertical Resolution Register of Scaler Output(scal_ver_no_out, Offset 0x110C) 543
Table 6-46. Scaler Control Register (scal_misc, Offset 0x1110) 543
Table 6-47. Horizontal High Threshold Register(scal_hor_high_th, Offset 0x1114) 544
Table 6-48. Horizontal Low Threshold Register (scal_hor_low_th, Offset 0x1118) 545
Table 6-49. Vertical High Threshold Register (scal_ver_high_th, Offset 0x111C) 545
Table 6-50. Vertical Low Threshold Register(scal_ver_low_th, Offset 0x1120) 545
Table 6-51. Scaler Resolution Parameter (scal_hor_ver_num, Offset 0x112C) 546
Table 6-52. Cursor Position Control Register (cursor_position, Offset 0x1200) 546
Table 6-53. Cursor Color Control Register (cur_color1 ~ cur_color15, Offset 0x1204 ~0x123C) 546
Table 6-54. Cursor Font (cur_font, Offset 0x1300 ~ 0x13FC) 547
Table 6-55. Cursor Font(cur_font, Offset 0x1600 ~ 0x1DFC) 549
Table 6-56. Virtual Screen Control Register (vs_control, Offset 0x1500) 556
Table 6-57. Virtual Screen1 Control Register(vs1_control, Offset 0x1504) 557
Table 6-58. Virtual Screen2 Control Register(vs2_control, Offset 0x1508) 557
Table 6-59. Virtual Screen3 Control Register (vs3_control, Offset 0x150C) 557
Table 6-60. OSD Windows On/Off Switch Register (osd_windowOn, Offset 0x2000) 558
Table 6-61. OSD Font Control Register 0 (osd_fontctrl0, Offset 0x2004) 558
Table 6-62. OSD Font Control Register 1 (osd_fontctrl1, Offset 0x2008) 560
Table 6-63. OSD Windows 1 to 4 Control Register 0 561
Table 6-64. OSD Windows 1 to 4 Control Register 1 561
Table 6-65. OSD Windows 1 to 4 Control Register 2 562
Table 6-66. OSD Multi-Color Font Base Address 562
Table 6-67. OSD Palette Color Registers 0 ~ 15 562
Table 6-68. 2×18 Font Matrix for Font '1' 563
Table 6-69. Partial Content of Font-RAM Format 564
Table 6-70. 12×18 Multi-color Font Matrix for Font '1' 564
Vantechn

Table 6-71. Partial Content of Font-RAM Format 565
Table 6-72. OSD Display-RAM Data 567
Table 6-73. OSD Dimension Control 567
Table 6-74. OSD Position Control 568
Table 6-75. OSD Foreground Color Control 568
Table 6-76. OSD Background Color Control 569
Table 6-77. OSD Font Database Write Accessing Port 569
Table 6-78. OSD Window Attribute Write Accessing Port 570
Table 8-1. HC Capability Register (HCCAP, Address = 0x000) 578
Table 8-2. HC Structural Parameters (HCSPARAMS, Address = 0x004) 579
Table 8-3. HC Capability Parameters (HCCPARAMS, Address = 0x008) 579
Table 8-4. HC USB Command Register (USBCMD, Address = 0x010) 580
Table 8-5. HC USB Status Register (USBSTS, Address = 0x014) 581
Table 8-6. HC USB Interrupt Enable Register (USBINTR, Address = 0x018) 582
Table 8-7. HC Frame Index Register (FRINDEX, Address = 0x01C) 583
Table 8-8. HC Periodic Frame List Base Address Register (PERIODICLISTBASE, Address = 0x024) 584
Table 8-9. HC Current Asynchronous List Address Register (ASYNCLISTADDR, Address = 0x028) 584
Table 8-10. HC Port Status and Control Register (PORTSC, Address = 0x030) 584
Table 8-11. HC Miscellaneous Register (HCMISC, Address = 0x040) 587
Table 8-12. HC Full-Speed (FS) EOF1 Timing Point Register (FS_EOF, Address = 0x044) 589
Table 8-13. HC High-Speed (HS) EOF1 Timing Point Register (HS_EOF, Address = 0x048) 591
Table 8-14. OTG Control Status Register (OTG_CSR, Address = 0x080) 592
Table 8-15. OTG Interrupt Status Register (OTG_ISR, Address = 0x084) 595
Table 8-16. OTG Interrupt Enable Register (OTG_IER, Address = 0x088) 596
Table 8-17. Global HC/OTG/DEV Interrupt Status Register (GLB_ISR, Address = 0x0C0) 596
Table 8-18. Global Mask of HC/OTG/DEV Interrupt Register (GLB_INT, Address = 0x0C4) 597
Table 8-19. Revision Information Register (REVISION, Address = 0x0E0) 597
Table 8-20. Configured Information Register (FEATURE, Address = 0x0E4) 598
Table 8-21. Device Main Control Register (DEV_CTL, Address = 0x100) 598
Table 8-22. Device Address Register (DEV_ADR, Address = 0x104) 600
Table 8-23. Device Test Register (DEV_TST, Address = 0x108) 601
Table 8-24. Device SOF Frame Number Register (DEV_SFN, Address = 0x10C) 601
Table 8-25. Device SOF Mask Timer Register (DEV_SMT, Address = 0x110) 602
Table 8-26. PHY Test Mode Selector Register (PHY_TST, Address = 0x114) 602
Table 8-27. Device Vendor-Specific I/O Control Register (DEV_VCTL, Address = 0x118) 603
Table 8-28. Device CX Configuration Status Register (DEV_CXCFG, Address = 0x11C) 603
Table 8-29. Device CX Configuration and FIFO Empty Status Register (DEV_CXCFE, Address = 0x120) 603
Table 8-30. Device Idle Counter Register (DEV_ICR, Address = 0x124) 604
Table 8-31. Device Mask of Interrupt Group Register (DEV_MIGR, Address = 0x130) 605
Table 8-32. Device Mask of Interrupt Source Group 0 Register (DEV_MISG0, Address = 0x134) 605
Table 8-33. Device Mask of Interrupt Source Group 1 Register (DEV_MISG1, Address = 0x138) 606
Table 8-34. Device Mask of Interrupt Source Group 2 Register (DEV_MISG2, Address = 0x13C) 607
Table 8-35. Device Interrupt Group Register (DEV_IGR, Address = 0x140) 609
Table 8-36. Device Interrupt Source Group 0 Register (DEV_ISG0, Address = 0x144) 609
Table 8-37. Device Interrupt Source Group 1 Register (DEV_ISG1, Address = 0x148) 610
Table 8-38. Device Interrupt Source Group 2 Register (DEV_ISG2, Address = 0x14C) 612
Table 8-39. Device Receive Zero-Length Data Packet Register (DEV_RXZ, Address = 0x150) 614
Table 8-40. Device Transfer Zero-length Data Packet Register (DEV_TXZ, Address = 0x154) 615
Table 8-41. Device Isochronous Sequential Error/Abort Register (DEV_ISE, Address = 0x158) 615
Table 8-42. Device IN Endpoint n MaxPacketSize Register (One per Endpoint, n = 1 ~ 8) (DEV_INMPS, Address = 0x160 + 4(n -1)) 616
Table 8-43. Device OUT Endpoint n MaxPacketSize Register (One per Endpoint, n = 1 ~ 8) (DEV_OUTMPS, Address = 0x180 + 4(n -1)) 617
Table 8-44. Device Endpoint 1 ~ 4 Map Register (DEV_EPMAP0, Address = 0x1A0) 618

Table 8-45. Device Endpoint 5 ~ 8 Map Register (DEV_EPMAP1, Address = 0x1A4) 619
Table 8-46. Device FIFO Map Register (DEV_FMAP, Address = 0x1A8) 620
Table 8-47. Device FIFO Configuration Register (DEV_FCFG, Address = 0x1AC) 621
Table 8-48. Device FIFO n Instruction and Byte Count Register (One per FIFO, n = 0 ~ 3) (DEV_FIBC, Address = 0x1B0 + 4n) 624
Table 8-49. Device DMA Target FIFO Number Register (DMA_TFN, Address = 0x1C0) 624
Table 8-50. Device DMA Controller Parameter Setting 0 Register (DMA_CPS0, Address = 0x1C4) 625
Table 8-51. Device DMA Controller Parameter Setting 1 Register (DMA_CPS1, Address = 0x1C8) 626
Table 8-52. Device DMA Controller Parameter Setting 2 Register (DMA_CPS2, Address = 0x1CC) 628
Table 8-53. Device DMA Controller Parameter Setting 3 Register (DMA_CPS3, Address = 0x1D0) 628
Table 8-54. Device DMA Controller Parameter Setting 4 Register (DMA_CPS4, Address = 0x1D4) 629
Table 8-55. Device Virtual DMA CXF Parameter Setting 1 Register (VDMA_CXFPS1, Address = 0x300) 629
Table 8-56. Device Virtual DMA CXF Parameter Setting 2 Register (VDMA_CXFPS2, Address = 0x304) 630
Table 8-57. Device Virtual DMA FIFO0 Parameter Setting 1 Register (VDMA_F0PS1, Address = 0x308) 630
Table 8-58. Device Virtual DMA FIFO0 Parameter Setting 2 Register (VDMA_F0PS2, Address = 0x30C) 631
Table 8-59. Device Virtual DMA FIFO1 Parameter Setting 1 Register (VDMA_F1PS1, Address = 0x310) 632
Table 8-60. Device Virtual DMA FIFO1 Parameter Setting 2 Register (VDMA_F1PS2, Address = 0x314) 633
Table 8-61. Device Virtual DMA FIFO2 Parameter Setting 1 Register (VDMA_F2PS1, Address = 0x318) 633
Table 8-62. Device Virtual DMA FIFO2 Parameter Setting 2 Register (VDMA_F2PS2, Address = 0x31C) 634
Table 8-63. Device Virtual DMA FIFO3 Parameter Setting 1 Register (VDMA_F3PS1, Address = 0x320) 634
Table 8-64. Device Virtual DMA FIFO3 Parameter Setting 2 Register (VDMA_F3PS2, Address = 0x324) 635
Table 8-65. Device Interrupt Source Group 3 Register (DEV_ISG3, Address = 0x328) 636
Table 8-66. Device Mask of Interrupt Source Group 3 Register (DEV_MISG3, Address = 0x32C) 638
Table 8-67. Device Virtual DMA Control Register (VDMA_CTRL, Address = 0x330) 639
Table 8-68. Device LPM Capability Register (LPM_CAP, Address = 0x334) 640
Table 8-69. Device Interrupt Source Group 4 Register (DEV_ISG4, Address = 0x338) 640
Table 8-70. Device Mask of Interrupt Source Group 4 Register (DEV_MISG4, Address = 0x33C) 640
Table 8-71. SSP Control Register 0 (SSPCR0, Offset = 0x00) 648
Table 8-72. SSP Control Register 1 (SSPCR1, Offset = 0x04) 652
Table 8-73. SSP Control Register 2 (SSPCR2, Offset = 0x08) 653
Table 8-74. SSP Status Register (SSPStatus, Offset = 0x0C) 655
Table 8-75. Interrupt Control Register (IntrCR, Offset = 0x10) 655
Table 8-76. Interrupt Status Register (IntrStatus, Offset = 0x14) 657
Table 8-77. SSP Control Register 3 (SSPCR3, Offset = 0x1C) 658
Table 8-78. SSP Revision Register (SSPRevision, Offset = 0x60) 658
Table 8-79. SSP Feature Register (SSPFeature, Offset = 0x64) 659
Table 8-80. the command slave port limitation 661
Table 8-81. The XIP slave port limitation 661
Table 8-82. Command Queue First Word (Cmd_w0, Offset = 0x0000) 670
Table 8-83. Command Queue Second Word (Cmd_w1, Offset = 0x0004) 671
Table 8-84. Command Queue Third Word (Cmd_w2, Offset = 0x0008) 672
Table 8-85. Command Queue Fourth Word (Cmd_w3, Offset = 0x000C) 672
Table 8-86. Control Register (CR, Offset = 0x0010) 674
Vangotech

Table 8-87. AC Timing Register (ACTR, Offset = 0x0014) 676
Table 8-88. Status Register (SR, Offset = 0x0018) 676
Table 8-89. Interrupt Control Register (ICR, Offset = 0x0020) 676
Table 8-90. Interrupt Status Register (ISR, Offset = 0x0024) 677
Table 8-91. SPI Read Status Register (SPISR, Offset = 0x0028) 677
Table 8-92. SPI Flash Size Register (SPIFSR, Offset = 0x002C) 677
Table 8-93. XIP Command Word (XIPCMD, Offset = 0x0030) 677
Table 8-94. XIP Prefetch Function Option (XIPPFO, Offset = 0x0034) 679
Table 8-95. Revision Register (REVISION, Offset = 0x0050) 679
Table 8-96. Feature Register (FEATURE, Offset = 0x0054) 679
Table 8-97. Data Port Register (DR, Offset = 0x0100) 680
Table 8-98. SPI Bus Specification 681
Table 8-99. I2C Control Register (Offset: 0x00) 691
Table 8-100. I2C Status Register(SR,Offset:0x04) 693
Table 8-101. I2C Clock Division Register (CDR, Offset: 0x08) 695
Table 8-102. I2C Data Register (Offset: 0x0C) 696
Table 8-103. I2C Address Register (AR, Offset: 0x10) 696
Table 8-104. I2C Set/Hold Time and Glitch Suppression Setting Register (TGSR, Offset: 0x14) 697
Table 8-105. I2C Bus Monitor Register (BMR, Offset: 0x18) 698
Table 8-106. I2C Burst Mode Register (BSTMR, Offset: 0x1C) 698
Table 8-107. I2C Revision Register (REVISION, Offset: 0x30) 698
Table 8-108 Register Summary Table 713
Table 8-109 Receiver Buffer Register (RBR, Offset: 0x00 for Read) 715
Table 8-110 Transmitter Holding Register (THR, Offset: 0x00 for Write) 716
Table 8-111 Interrupt Enable Register (IER, Offset: 0x04) 716
Table 8-112 Interrupt Identification Register (IIR , Offset: 0x08) 717
Table 8-113 Interrupt Control Table 718
Table 8-114 FIFO Control Register (FCR, Offset: 0x08 for Write) 720
Table 8-115 FIFO Trigger Level of Receiver 721
Table 8-116 FIFO Trigger Level of Transmitter 721
Table 8-117 Line Control Register (LCR, Offset: 0x0C) 721
Table 8-118 Parity Setting Table 723
Table 8-119 Word Length and Stop Bit Setting 723
Table 8-120 Modem Control Register (MCR, Offset: 0x10) 723
Table 8-121 Line Status Register (LSR, Offset: 0x14 for Read) 725
Table 8-122 Testing Register (TST, Offset: 0x14 for Write) 727
Table 8-123 Modem Status Register (MSR, Offset: 0x18) 727
Table 8-124 Scratch Pad Register (SPR, Offset: 0x1C) 728
Table 8-125 Baud Rate Divisor Latch LSB (DLL, Offset: 0x00 when DLAB = 1) 728
Table 8-126 Baud Rate Divisor Latch MSB (DLM, Offset: 0x04 when DLAB = 1) 728
Table 8-127 Prescaler Register (PSR, Offset: 0x08 when DLAB = 1) 729
Table 8-128 Mode Definition Register (MDR, Offset: 0x20) 729
Table 8-129 Auxiliary Control Register (ACR, Offset: 0x24) 730
Table 8-130 Status FIFO Trigger Level 731
Table 8-131 Transmit Frame-Length Register Low (TXLENL, Offset: 0x28) 731
Table 8-132 Transmit Frame-Length Register High (TXLENH, Offset: 0x2C) 731
Table 8-133 Maximum Receiver Frame-Length Register Low (MRXLENL, Offset: 0x30) 731
Table 8-134 Maximum Receiver Frame-Length Register High (MRXLENH, Offset: 0x34) 732
Table 8-135 Preamble Length Register 732
Table 8-136 Number of FIR Preambles 732
Table 8-137 Interrupt Identification Register in FIR Mode (FMIIR_PIO, Offset: 0x3C, PIO Mode) 732
Table 8-138 Interrupt Identification Register in FIR Mode (FMIIR_DMA, Offset: 0x3C, DMA Mode) 733
Table 8-139 Interrupt Enable Register in FIR Mode (FMIIER_PIO, Offset: 0x40, PIO Mode) 734
Table 8-140 Interrupt Enable Register in FIR Mode (FMIIER_DMA, Offset: 0x40, DMA Mode) 734
Table 8-141 Status FIFO Line Status Register (STFF_STS, Offset: 0x44) 735
Table 8-142 Status FIFO Received Frame-Length Register Low (STFF_RXLENL, Offset: 0x48) 735
Vangotech

Table 8-143 Status FIFO Received Frame-Length Register High (STFF_RXLENH, Offset: 0x4C) ... 735
Table 8-144 Link Status Register in FIR Mode (FMLSR, Offset: 0x50) ... 736
Table 8-145 Link Status Interrupt Enable Register in FIR Mode (FMLSIER, Offset: 0x54) ... 736
Table 8-146. RX FIFO Count Register (RXFF_CNTR, Offset: 0x5C) ... 737
Table 8-147. Last Frame-Length Register Low (LSTFMLENL, Offset: 0x60) ... 737
Table 8-148 Last Frame-Length Register High (LSTFMLENH, Offset: 0x64) ... 737
Table 8-149 Frame Number Decoding Table (LSTFMLENH, Offset: 0x64) ... 738
Table 8-150 Feature Register (Feature, Offset: 0x68) ... 738
Table 8-151. Revision Register Digit 1 (REVD1, Offset: 0x6C) ... 739
Table 8-152. Revision Register Digit 2 (REVD2, Offset: 0x70) ... 739
Table 8-153. Revision Register Digit 3 (REVD3, Offset: 0x74) ... 739
Table 8-154. Wake-up Frame Format ... 749
Table 8-155. FTGMAC030 Frame Structure and Frame Formats ... 754
Table 8-156. PHY Bit Stream Format for 802.3 clause 22. ... 755
Table 8-157. Register Summary Table ... 756
Table 8-158. Interrupt Status Register, (ISR, Offset: 0x0) ... 759
Table 8-159. Interrupt Enable Register (IME, Offset: 0x4) ... 760
Table 8-160. Most Significant Address Register (MAC_MADR, Offset: 0x8) ... 760
Table 8-161. Least Significant Address Register (MAC_LADR, Offset: 0xC) ... 761
Table 8-162. Multicast Address Hash Table 0 Register (MAHT0, Offset: 0x10) ... 761
Table 8-163. Multicast Address Hash Table 1 Register (MAHT1, Offset: 0x14) ... 761
Table 8-164. Normal Priority Transmit Poll Demand Register (NPTXPD, Offset: 0x18) ... 761
Table 8-165. Receive Poll Demand Register (RXPD, Offset: 0x1C) ... 761
Table 8-166. Normal Priority Transmit Ring Base Address Register (NPTXR_BADR, Offset: 0x20) ... 761
Table 8-167. Receive Ring Base Address Register (RXR_BADR, Offset: 0x24) ... 762
Table 8-168. High Priority Transmit Poll Demand Register (HPTXPD, Offset: 0x28) ... 762
Table 8-169. High Priority Transmit Ring Base Address Register (HPTXR_BADR, Offset: 0x2C) ... 762
Table 8-170. TX Interrupt Timer Control Register (TXITC, Offset: 0x30) ... 762
Table 8-171. RX Interrupt Timer Control Register (RXITC, Offset: 0x34) ... 764
Table 8-172. Automatic Polling Timer Control Register (APTC, Offset: 0x38) ... 766
Table 8-173. DMA Burst Length and Arbitration Control Register (DBLAC, Offset: 0x3C) ... 767
Table 8-174. DMA/FIFO State Register (DMAFIFOS, Offset: 0x40) ... 769
Table 8-175. Transmit Priority Arbitration and FIFO Control Register (TPAFCR, Offset: 0x48) ... 770
Table 8-176. Receive Buffer Size Register (RBSR, Offset: 0x4C) ... 771
Table 8-177. MAC Control Register (MACCR, Offset: 0x50) ... 771
Table 8-178. MAC Status Register (MACSR, Offset: 0x54) ... 773
Table 8-179. Test Mode Register (TM, Offset: 0x58) ... 773
Table 8-180. PHY Control Register (PHYCR, Offset: 0x60) ... 774
Table 8-181. PHY Data Register (PHYDATA, Offset: 0x64) ... 775
Table 8-182. Flow Control Register (FCR, Offset: 0x68) ... 775
Table 8-183. Back Pressure Register (BPR, Offset: 0x6C) ... 776
Table 8-184. Wake-On-LAN Control Register (WOLCR, Offset: 0x70) ... 776
Table 8-185. Wake-On-LAN Status Register (WOLSR, Offset: 0x74) ... 777
Table 8-186. Wake-up Frame CRC Register (WFCRC, Offset: 0x78) ... 778
Table 8-187. Wake-up Frame Byte Mask 1st Double-word Register (WFBM1, Offset: 0x80) ... 778
Table 8-188. Wake-up Frame Byte Mask 2nd Double-word Register (WFBM2, Offset: 0x84) ... 779
Table 8-189. Wake-up Frame Byte Mask 3rd Double-word Register (WFBM3, Offset: 0x88) ... 779
Table 8-190. Wake-up Frame Byte Mask 4th Double-word Register (WFBM4, Offset: 0x8C) ... 780
Table 8-191. Normal Priority Transmit Ring Pointer Register (NPTXR_PTR, Offset: 0x90) ... 780
Table 8-192. High Priority Transmit Ring Pointer Register (HPTXR_PTR, Offset: 0x94) ... 781
Table 8-193. Receive Ring Pointer Register (RXR_PTR, Offset: 0x98) ... 781
Table 8-194. TPKT_CNT Counter Register (TX_CNT1, Offset: 0xA0) ... 781
Table 8-195. TXMCOL_CNT and TXSCOL_CNT Counter Register (TX_CNT2, Offset: 0xA4) ... 781
Table 8-196. TXECOL_CNT and TXFAIL_CNT Counter Register (TX_CNT3, Offset: 0xA8) ... 782
Table 8-197. TXLCOL_CNT and TXUNDERUN_CNT Counter Register (TX_CNT4, Offset: 0xAC) ... 782
Table 8-198. RPKT_CNT Counter Register (RX_CNT1, Offset: 0xB0) ... 782
Vangotech

Table 8-199. BROPKT_CNT Counter Register (RX_CNT2, Offset: 0xB4) 782
Table 8-200. MULPKT_CNT Counter Register (RX_CNT3, Offset: 0xB8) 782
Table 8-201. RPF_CNT and AEP_CNT Counter Register (RX_CNT4, Offset: 0xBC) 782
Table 8-202. RUNT_CNT Counter Register 783
Table 8-203. CRCER_CNT and FTL_CNT Counter Register (RX_CNT6, Offset: 0xC4h) 783
Table 8-204. RCOL_CNT and RLOST_CNT Counter Register (RX_CNT7, Offset: 0xC8) 783
Table 8-205. BIST Pattern Control Register (BISTR, Offset: 0xCC) 783
Table 8-206. Broadcast and Multicast Receiving Control Register (BMRCR, Offset: 0xD0) 784
Table 8-207. Error Response Control Register (ERCR, Offset: 0xE0) 785
Table 8-208. AXI Additional Control Information Register (ACIR, Offset: 0xE4) 785
Table 8-209. MAC Interface Selection Register (GISR, Offset: 0xE8) 786
Table 8-210. SW Reset Cycle Count Register (SCCR, Offset: 0xEC) 786
Table 8-211. EEE Control Register (ECT, Offset: 0xF0) 786
Table 8-212. Revision Register (REVR, Offset: 0xF4) 786
Table 8-213. Feature Register (FEAR, Offset: 0xF8) 787
Table 8-214. PTP RX Unicast IP Destination Address Register (PTP_RX_UnDA, Offset: 0x100) 788
Table 8-215. PTP TX Unicast IP Destination Address Register (PTP_TX_UnDA, Offset: 0x104) 788
Table 8-216. PTP TX Event Frame Register (PTP_TX, Offset: 0x110) 788
Table 8-217. PTP TX Event Frame Register (PTP_TX Offset: 0x114) 788
Table 8-218. PTP RX Event Frame Register, PTP_RX (Offset: 0x118) 789
Table 8-219. PTP RX Event Frame Register, PTP_RX (Offset: 0x11C) 789
Table 8-220. PTP TX Peer Frame Register (PTP_TX_P, Offset: 0x120) 789
Table 8-221. PTP TX Peer Frame Register (PTP_TX_P, Offset: 0x124) 789
Table 8-222. PTP RX Peer Frame Register (PTP_RX_P, Offset: 0x128) 789
Table 8-223. PTP RX Peer Frame Register (PTP_RX_P, Offset: 0x12C) 790
Table 8-224. PTP Timer Register (PTP_TMR, Offset: 0x130) 790
Table 8-225. PTP Timer Register (PTP_TMR, Offset: 0x134) 790
Table 8-226. PTP Timer Register (PTP_TMR, Offset: 0x138) 790
Table 8-227. PTP Period Increment 0 Register (PTP_PER0, Offset: 0x13C) 790
Table 8-228. PTP Period Increment 1 Register (PTP_PER1, Offset: 0x140) 791
Table 8-229. PTP Period Offset Increment Register (PTP_OFF, Offset: 0x144) 791
Table 8-230. PTP Timer Adjustment Register (PTP_ADJ, Offset: 0x148) 791
Table 8-231 Register Summary Table 794
Table 8-232 GPIO Data-out Register (GpioDataOut, Offset = 0x00) 796
Table 8-233 GPIO Data-in Register (GpioDataIn, Offset = 0x04) 796
Table 8-234 Pin Direction Register (PinDir, Offset = 0x08) 796
Table 8-235 Pin Bypass Mode Register (PinBypass, Offset = 0x0C) 797
Table 8-236 GPIO Data Bit Set Register (GpioDataSet, Offset = 0x10) 797
Table 8-237 GPIO Data Bit Clear Register (GpioDataClear, Offset = 0x14) 797
Table 8-238 Pin Pull Enable Register (PinPullEnable, Offset = 0x18) 797
Table 8-239 Pin Pull Type Register (PinPullType, Offset = 0x1C) 797
Table 8-240 Pull Truth Table 798
Table 8-241 Interrupt Enable Register (IntrEnable, Offset = 0x20) 798
Table 8-242 Interrupt Raw State Register (IntrRawState, Offset = 0x24) 798
Table 8-243 Interrupt Masked State Register (IntrMaskedState, Offset = 0x28) 798
Table 8-244 Interrupt Mask Register (IntrMask, Offset = 0x2C) 798
Table 8-245 Interrupt Clear Register (IntrClear, Offset = 0x30) 799
Table 8-246 IntrTrigger Register (IntrTrigger, Offset = 0x34) 799
Table 8-247 Interrupt Both Edge Trigger Register (IntrBoth, Offset = 0x38) 799
Table 8-248 Interrupt Rise or Neg Edge Trigger Register (IntrRiseNeg, Offset = 0x3C) 799
Table 8-249 Bounce Enable Register (BounceEnable, Offset = 0x40) 800
Table 8-250 Bounce Clock Pre-scale Register (BouncePreScale, Offset = 0x44) 800
Table 8-251. Register Summary Table 804
Table 8-252. Command Request Register (CR, Offset: 0x0000) 810
Table 8-253. Interrupt Request Enable Register (IRE, Offset: 0x0001) 811
Table 8-254. Control Enable 0 Register (CE0, Offset: 0x0002) 812
Table 8-255. Control Enable 1 Register (CE1, Offset: 0x0003) 812
Vangotech

Table 8-256. Bus Status Register (BS, Offset: 0x0004) 813
Table 8-257. Transmit and Receive Buffer Status Register (TRBS, Offset: 0x0005) 814
Table 8-258. Transmit Status Register (TS, Offset: 0x0006) 814
Table 8-259. Reserved (Offset: 0x0007) 815
Table 8-260. Transmit Flexible Data-rate Register 0 (TFD0, Offset: 0x0008) 815
Table 8-261. Transmit Remote and Extended Identifier Enable Register 0 (TREIE0, Offset: 0x0009) 816
Table 8-262. Transmit Extended Identifier Low Register 0 (TEIL0, Offset: 0x000A) 816
Table 8-263. Transmit Extended Identifier Middle Register 0 (TEIM0, Offset: 0x000B) 816
Table 8-264. Transmit Extended Identifier High Register 0 (TEIH0, Offset: 0x000C) 816
Table 8-265. Transmit Base Identifier Low and Data Length Register 0 (TBILDL0, Offset: 0x000D) 817
Table 8-266. Transmit Base Identifier High Register 0 (TBIH0, Offset: 0x000E) 817
Table 8-267. Reserved (Offset: 0x000F) 817
Table 8-268. Transmit flexible data-rate Register 1 (TFD1, Offset: 0x0010) 817
Table 8-269. Transmit Remote and Extended Identifier Enable Register 1 (TREIE1, Offset: 0x0011) 818
Table 8-270. Transmit Extended Identifier Low Register 1 (TEIL1, Offset: 0x0012) 818
Table 8-271. Transmit Extended Identifier Middle Register 1 (TEIM1, Offset: 0x0013) 818
Table 8-272. Transmit Extended Identifier High Register 1 (TEIH1, Offset: 0x0014) 819
Table 8-273. Transmit Base Identifier Low and Data Length Register 1 (TBILDL1, Offset: 0x0015) 819
Table 8-274. Transmit Base Identifier High Register 1 (TBIH1, Offset: 0x0016) 819
Table 8-275. Reserved (Offset: 0x0017) 820
Table 8-276. Transmit Flexible Data-rate Register 2 (TFD2, Offset: 0x0018) 820
Table 8-277. Transmit Remote and Extended Identifier Enable Register 2 (TREIE2, Offset: 0x0019) 820
Table 8-278. Transmit Extended Identifier Low Register 2 (TEIL2, Offset: 0x001A) 821
Table 8-279. Transmit Extended Identifier Middle Register 2 (TEIM2, Offset: 0x001B) 821
Table 8-280. Transmit Extended Identifier High Register 2 (TEIH2, Offset: 0x001C) 821
Table 8-281. Transmit Base Identifier Low and Data Length Register 2 (TBILDL2, Offset: 0x001D) 821
Table 8-282. Transmit Base Identifier High Register 2 (TBIH2, Offset: 0x001E) 822
Table 8-283. Reserved (Offset: 0x001F) 822
Table 8-284. Acceptance Filter for Extended Identifier Low Bit Register 0 (AFEIL0, Offset: 0x0020) 822
Table 8-285. Acceptance Filter for Extended Identifier Middle Bit Register 0 (AFEIM0, Offset: 0x0021) 822
Table 8-286. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 0 (AFBILEIH0, Offset: 0x0022) 823
Table 8-287. Acceptance Filter for Base Identifier High Bit Register 0 (AFBIH0, Offset: 0x0023) 823
Table 8-288. Acceptance Filter for Extended Identifier Low Bit Register 1 (AFEIL1, Offset: 0x0024) 823
Table 8-289. Acceptance Filter for Extended Identifier Middle Bit Register 1 (AFEIM1, Offset: 0x0025) 823
Table 8-290. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 1 (AFBILEIH1, Offset: 0x0026) 823
Table 8-291. Acceptance Filter for Base Identifier High Bit Register 1 (AFBIH1, Offset: 0x0027) 823
Table 8-292. Acceptance Filter for Extended Identifier Low Bit Register 2 (AFEIL2, Offset: 0x0028) 823
Table 8-293. Acceptance Filter for Extended Identifier Middle Bit Register 2 (AFEIM2, Offset: 0x0029) 824
Table 8-294. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 2 (AFBILEIH2, Offset: 0x002A) 824
Table 8-295. Acceptance Filter for Base Identifier High Bit Register 2 (AFBIH2, Offset: 0x002B) 824
Table 8-296. Acceptance Filter for Extended Identifier Low Bit Register 3 (AFEIL3, Offset: 0x002C) 824
Vangotech

Table 8-297. Acceptance Filter for Extended Identifier Middle Bit Register 3 (AFEIM3, Offset: 0x002D) 824
Table 8-298. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 3 (AFBILEIH3, Offset: 0x002E) 824
Table 8-299. Acceptance Filter for Base Identifier High Bit Register 3 (AFBIH3, Offset: 0x002F) 825
Table 8-300. Acceptance Filter for Extended Identifier Low Bit Register 4 (AFEIL4, Offset: 0x0030) 825
Table 8-301. Acceptance Filter for Extended Identifier Middle Bit Register 4 (AFEIM4, Offset: 0x0031) 825
Table 8-302. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 4 (AFBILEIH4, Offset: 0x0032) 825
Table 8-303. Acceptance Filter for Base Identifier High Bit Register 4 (AFBIH4, Offset: 0x0033) 825
Table 8-304. Acceptance Filter for Extended Identifier Low Bit Register 5 (AFEIL5, Offset: 0x0034) 825
Table 8-305. Acceptance Filter for Extended Identifier Middle Bit Register 5 (AFEIM5, Offset: 0x0035) 826
Table 8-306. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 5 (AFBILEIH5, Offset: 0x0036) 826
Table 8-307. Acceptance Filter for Base Identifier High Bit Register 5 (AFBIH5, Offset: 0x0037) 826
Table 8-308. Acceptance Filter Data Byte0 Register 0 (AFDB00, Offset: 0x0038) 826
Table 8-309. Acceptance Filter Data Byte1 Register 0 (AFDB10, Offset: 0x0039) 826
Table 8-310. Acceptance Filter Data Byte0 Register 1 (AFDB01, Offset: 0x003A) 826
Table 8-311. Acceptance Filter Data Byte1 Register 1 (AFDB11, Offset: 0x003B) 826
Table 8-312. Acceptance Filter Control Register 0 (AFC0, Offset: 0x003C) 826
Table 8-313. Acceptance Filter Control Register 1 (AFC1, Offset: 0x003D) 827
Table 8-314. Reserved (Offset: 0x003E) 828
Table 8-315. Reserved (Offset: 0x003F) 828
Table 8-316. Filter Mask Extended Identifier Low Register 0 (FMEIL0, Offset: 0x0040) 828
Table 8-317. Filter Mask Extended Identifier Middle Register 0 (FMEIM0, Offset: 0x0041) 828
Table 8-318. Filter Mask Base Identifier Low and Extended Identifier High 0 (FMBILEIH0, Offset: 0x0042) 828
Table 8-319. Filter Mask Base Identifier High Register 0 (FMBIH0, Offset: 0x0043) 829
Table 8-320. Filter Mask Extended Identifier Low Register 1 (FMEIL1, Offset: 0x0044) 829
Table 8-321. Filter Mask Extended Identifier Middle Register 1 (FMEIM1, Offset: 0x0045) 829
Table 8-322. Filter Mask Base Identifier Low and Extended Identifier High 1 (FMBILEIH1, Offset: 0x0046) 829
Table 8-323. Filter Mask Base Identifier High Register 1 (FMBIH1, Offset: 0x0047) 829
Table 8-324. Filter Mask Data Byte0 Register 0 (FMDB00, Offset: 0x0048) 829
Table 8-325. Filter Mask Data Byte1 Register 0 (FMDB10, Offset: 0x0049) 829
Table 8-326. Filter Mask Data Byte0 Register 1 (FMDB01, Offset: 0x004A) 829
Table 8-327. Filter Mask Data Byte1 Register 1 (FMDB11, Offset: 0x004B) 830
Table 8-328. Filter Mask Control Register (FMC, Offset: 0x004C) 830
Table 8-329. Reserved (Offset: 0x004D) 830
Table 8-330. Reserved (Offset: 0x004E) 831
Table 8-331. Reserved (Offset: 0x004F) 831
Table 8-332. Received Flexible Data-Rate Register 0 (RFD0, Offset: 0x0050) 831
Table 8-333. Received Remote and Extended Identifier Enable Register 0 (RREIE0, Offset: 0x0051) 831
Table 8-334. Received Extended Identifier Low Register 0 (REIL0, Offset: 0x0052) 831
Table 8-335. Received Extended Identifier Middle Register 0 (REIM0, Offset: 0x0053) 832
Table 8-336. Received Extended Identifier High Register 0 (REIH0, Offset: 0x0054) 832
Table 8-337. Received Base Identifier Low and Data Length Register 0 (RBILDL0, Offset: 0x0055) 832
Table 8-338. Received Base Identifier High Register 0 (RBIH0, Offset: 0x0056) 833
Table 8-339. Reserved (Offset: 0x0057) 833
Table 8-340. Received Flexible Data-Rate Register 1 (RFD1, Offset: 0x0058) 833
Table 8-341. Received Remote and Extended Identifier Enable Register 1 (RREIE1, Offset: 0x0059) 833

Vangotech

Table 8-342. Received Extended Identifier Low Register 1 (REIL1, Offset: 0x005A) 833
Table 8-343. Received Extended Identifier Middle Register 1 (REIM1, Offset: 0x005B) 834
Table 8-344. Received Extended Identifier High Register 1 (REIH1, Offset: 0x005C) 834
Table 8-345. Received Base Identifier Low and Data Length Register 1 (RBILDL1, Offset: 0x005D) 834
Table 8-346. Received Base Identifier High Register 1 (RBIH1, Offset: 0x005E) 834
Table 8-347. Reserved (Offset: 0x005F) 835
Table 8-348. Nominal Bit Timing Configuration 0 Register (NBTC0, Offset: 0x0060) 835
Table 8-349. Nominal Bit Timing Configuration 1 Register (NBTC1, Offset: 0x0061) 835
Table 8-350. Nominal Bit Timing Configuration 2 Register (NBTC2, Offset: 0x0062) 835
Table 8-351. Nominal Bit Timing Configuration 3 Register (NBTC3, Offset: 0x0063) 835
Table 8-352. Nominal Bit Timing Configuration 4 Register (NBTC4, Offset: 0x0064) 835
Table 8-353. Reserved (Offset: 0x0065) 836
Table 8-354. Reserved (Offset: 0x0066) 836
Table 8-355. Reserved (Offset: 0x0067) 836
Table 8-356. Data Bit Timing Configuration 0 Register (DBTC0, Offset: 0x0068) 836
Table 8-357. Data Bit Timing Configuration 1 Register (DBTC1, Offset: 0x0069) 836
Table 8-358. Data Bit Timing Configuration 2 Register (DBTC2, Offset: 0x006A) 837
Table 8-359. Data Bit Timing Configuration 3 Register (DBTC3, Offset: 0x006B) 837
Table 8-360. Receiver Error Counter Low Register (RECL, Offset: 0x006C) 837
Table 8-361. Receiver Error Counter High Register (RECH, Offset: 0x006D) 837
Table 8-362. Transmitter Error Counter Low Register (TECL, Offset: 0x006E) 837
Table 8-363. Transmitter Error Counter High Register (TECH, Offset: 0x006F) 837
Table 8-364. Error Type Register (ET, Offset: 0x0070) 837
Table 8-365. Transmitter and Receive Error Detected Register (TED, Offset: 0x0071) 838
Table 8-366. Reserved (Offset: 0x0072) 838
Table 8-367. Reserved (Offset: 0x0073) 838
Table 8-368. Receive Buffer Time Stamp Low 0 (RBTSL0, Offset: 0x0074) 839
Table 8-369. Receive Buffer Time Stamp High 0 (RBTSH0, Offset: 0x0075) 839
Table 8-370. Receive Buffer Time Stamp Low 1 (RBTSL1, Offset: 0x0076) 839
Table 8-371. Receive Buffer Time Stamp High 1(RBTSH1, Offset: 0x0077) 839
Table 8-372. Interrupt Request Register (IR, Offset: 0x0078) 839
Table 8-373. Transmit Buffer0 Data Register (TB0D, Offset: 0x007C ~ 0x00BB) 840
Table 8-374. Transmit Buffer1 Data Register (TB1D, Offset: 0x00BC ~ 0x00FB) 840
Table 8-375. Transmit Buffer2 Data Register (TB2D, Offset: 0x00FC ~ 0x013B) 841
Table 8-376. Receive Buffer0 Data Register (RB0D, Offset: 0x0013C ~ 0x017B) 841
Table 8-377. Receive Buffer1 Data Register (RB1D, Offset: 0x0017C ~ 0x01B8) 841
Table 8-378. Control Register (CTRL, Offset: 0x00) 845
Table 8-379. Sampling Rate Division register(SRDR, Offset: 0x04) 846
Table 8-380. Request to Send Counter Register (RSCR, Offset 0x08) 846
Table 8-381. Status Register (STS, Offset: 0x0C) 846
Table 8-382. Interrupt Status Register (ISR, Offset: 0x10) 846
Table 8-383. Keyboard Receive Register (KRR, Offset 0x14) 847
Table 8-384. Keyboard Transmit Register (KTR, Offset: 0x18) 847
Table 8-385. Interrupt Mask Register (IMR, Offset 0x1C) 847
Table 8-386. Keypad Code X Register (KCXR, Offset 0x30) 847
Table 8-387. Keypad Code Y Register (KCYR, Offset 0x34) 847
Table 8-388. Auto-Scan Period Register (ASPR, Offset 0x38) 848
Table 8-389. Revision Register (REV, Offset 0x50) 848
Table 8-390. Feature Register (FEATURE, Offset 0x54) 848
Table 9-1. Absolute Maximum Ratings 849
Table 9-2. Recommended Operating Conditions 850
Table 9-3. DC Electrical Characteristics 851
Table 9-4. DC Input Logic Level 852
Table 9-5. Output DC Current Drive 852
Table 9-6. 24MHz Crystal Characteristics 852
Table 9-7. 32.768kHz Crystal Characteristics 853
Vangotech

Table 9-8. LCD HV_IF Interface Timing Constants 854
Table 9-9. SPI Timing Constants 855
Table 9-10. UART Timing Constants 856
Table 9-11. AC97 Timing Constants 858
Table 9-12. SCR Timing Constants 860
Table 10-1. V8130L Thermal Resistance Characteristics 862
Table 12-1. Matrix Tray Carrier Information 865
Table 12-2. Packing Quantity Information 865
Table 12-3. MSL Summary 867
Table 12-4. Bagged Storage Conditions 867
Table 12-5. Out-of-bag Duration 868
Table 13-1. Lead-free Reflow Profile Conditions 869
Table 14-1. V8130L Marking Definitions 871

Vangotech

About This Documentation

- Purpose
- Intended Audience
- Symbol Conventions
- Acronyms and Abbreviations

Vangotech

1.1. Purpose

This document provides an overall description of the VANGO's V8130L application processor, which describes the overview, features, logical structures, functions and register listings of each module. The document also describes the interface timings and related parameters in diagrams. In addition, the document describes the pins, pin usages, performance parameters, and package dimension of V8130L in detail.

1.2. Intended Audience

The document is intended for:

- Design and maintenance personnel for electronics
- Sales personnel for electronic parts and components

1.3. Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
WARNING	A warning means that injury or death is possible if the instructions are not obeyed.
CAUTION	A caution means that damage to equipment is possible.
NOTE	Provides additional information to emphasize or supplement important points of the main text.

1.4. Acronyms and Abbreviations

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

AES	Advanced Encryption Standard
ADC	Analog-to-Digital Converter
AHB	AMBA High-speed Bus
APB	Advanced Peripheral Bus
ARM	Advanced RISC Machine
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DES	Data Encryption Standard
DLL	Delay-Locked Loop
DMA	Direct Memory Access
ECC	Error Correction Code

Vangotech

eFuse	Electrical Fuse, A one-time programmable memory
EHCI	Enhanced Host Controller Interface
eMMC	Embedded Multi-Media Card
ESD	Electrostatic Discharge
FBGA	Fine Ball Grid Array
FIFO	First In First Out
GIC	Generic Interrupt Controller
GMII	Gigabit Media Independent Interface
GPIO	General Purpose Input Output
GPU	Graphics Processing Unit
I2C	Inter Integrated Circuit
I2S	Inter IC Sound
JEDEC	Joint Electron Device Engineering Council
JTAG	Joint Test Action Group
LCD	Liquid-Crystal Display
LSB	Least Significant Bit
MAC	Media Access Control
MII	Media Independent Interface
MMC	Multimedia Card
MSB	Most Significant Bit
N/A	Not Application
NMI	Non Maskable Interrupt
PAL	Phase Alternating Line
PCM	Pulse Code Modulation
PHY	Physical Layer Controller
PID	Packet Identifier
PLL	Phase-Locked Loop
PWM	Pulse Width Modulation
R	Read only/non-Write
RGB	Read Green Blue
RMII	Reduced Media Independent Interface
ROM	Read Only Memory
RTC	Real Time Clock
RW	Read/Write
RWAC	Read/Write-Automatic-Clear,clear the bit automatically when the operation of complete.Write 0 has no effect
RWC	Read/Write-Clear
RW1C	Read/Write 1 to Clear, Writing 0 has non-effect
RW1S	Read/Write 1 to Set, Writing 0 has non-effect
Rsvd	Reserved: Reserved for future RO implementations. Software shall ignore the value read from these bits.
SDIO	Secure Digital Input Output
SDRAM	Sychronous Dynamic Random Access Memory
SOC	System On Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver Transmitter
UDF	Undefined

Vangotech

USB	Universal Serial Bus
UTMI	USB2.0 Transceiver Macrocell Interface

Vangotech

Chapter 2. Overview

This part describes the overview for V8130L processor.

- Processor Overview
- Features
- Block Diagram

Vangotech

2.1. Processor Overview

The V8130L processor represents VANGO's latest achievement in intelligent industrial control processors. The processor is ideal for applications that require advanced data processing, rich user interfaces, lower power consumption and higher system integration.

The V8130L processor has some very exciting features:

• **CPU:** V8130L is based on quad-core Cortex™-A7 CPU architecture, the most power efficient CPU core ARM's ever developed.

• **Display:** Content can be displayed on RGB panel.

• **Audio:** Supports I2S interface for connecting to an external audio codec.

• **Memory:** Supports external memory interfaces to NAND Flash, eMMC, NOR Flash and SDRAM port. SDRAM port can be configured to support LPDDR2, DDR2, DDR3, DDR3L, MDDR.

• **Peripherals:** To reduce total system cost, V8130L has a broad range of hardware peripherals to meet the flexible peripheral configuration requirements such as UART, SPI, USB2.0 OTG, CAN, I2C, MAC, eMMC, etc.

Vangotech

2.2. Features

2.2.1. CPU Architecture

- Quad-core ARM Cortex™-A7 Processor
  - ARMv7 ISA standard ARM instruction set
  - Thumb-2 Technology
  - TrustZone
  - Jazeller RCT
  - NEON Advanced SIMD
  - VFPv4 floating point
  - Large Physical Address Extensions(LPAE)
  - 32KB L1 Instruction cache and 32KB L1 Data cache for per CPU
  - 512KB L2 cache shared

2.2.2. Memory

2.2.2.1. Boot ROM

- On-chip 32KB ROM boot loader
- Supports fast boot from SPI NAND Flash, eMMC, SPI NOR Flash, or those images downloaded via UART.
- Supports system code download through USB OTG, UART, and Ethernet

2.2.2.2. Embedded SRAM

- 128KB SRAM for system
- 2 instances of 32KB SRAM for one port RGMII GMAC controller memory
- 2 instances of 4KB SRAM for one port RMII MAC controller memory
- 2 instances of 4KB SRAM for one port MII MAC controller memory
- 32KB SRAM for CA7

2.2.2.3. SDRAM

- Compatible with JEDEC standard DDR2/DDR3/DDR3L/LPDDR2/MDDR SDRAM
- Up to 4GB address space
- 8/16/32-bit data bus width
- 8-bit for ECC that can do one-bit error correction, and two-bit error detection
- Supports clock frequency up to 1600MHz(DDR3)

2.2.2.4. eMMC

- Up to two ports, the first one is bootable
- JEDEC eMMC 5.1 mandatory part
- HS200, not support high speed DDR and HS400
- Up to 32 GB in device density

2.2.3. System Peripherals

2.2.3.1. Timer

- 8 independent 32-bit timers
- Four watchdog to generate reset signal or interrupt

Vangotech

• Internal or external clock source for each timer

2.2.3.2. GIC

• Supports 16 SGIs(Software Generated Interrupt), 16 PPIs(Private Peripheral Interrupt) and 101 SPIs(Shared Peripheral Interrupts)

• Supports ARM architecture security extensions

• Supports ARM architecture virtualization extensions

2.2.3.3. DMA

• 3 instances ,8 channels for each instance. Two instances in AXI, Channel arbiter with 2-level round-robin mechanism. One instance in AHB, Channel arbiter with 4-level round-robin mechanism.

• Interrupt generated for each DMA channel

• Transfers data width of 8/16/32/64-bit

• Supports linear and IO address modes

• Programs the DMA burst size

• Flexible data source and destination address generation

• Supports data transfer types with memory-to-memory, memory-to-peripheral, peripheral-to-memory

2.2.3.4. SCU

• 7 PLLs. There are total 7 PLL instances to create AXI, APB, and peripheral clocks.

• Two external oscillators. One is 25MHz oscillator, and another is 32.768kHz oscillator.

• Two internal ring oscillators for CPU fast booting purpose and RTC. One is 120MHz, and +/- 5% accuracy for CPU operation clock, and another is 32.768KHz for RTC awake module purpose.

• Supports clock configuration and clock generated for corresponding modules.

• Supports software-controlled clock gating and software-controlled reset for corresponding modules.

2.2.3.5. RTC

• Timer, Calendar, Alarm

• Supports full clock features: second/minute/hour/day/month/year(with leap year)

2.2.3.6. PWM

• 16 PWM channels outputs(4 PWM instances)

• Supports three kinds of output waveforms: continuous waveform, pulse waveform and complementarity pair

• Programmable deadzone generator and controllable dead-time

• 0% to 100% adjustable duty cycle

• Up to 15/100MHz output frequency

• Minimum resolution is 1/(65536*65536)

• Supports interrupt for PWM output

2.2.3.7. Thermal Sensor

• Temperature Accuracy : ±3℃ from 0℃ to +100℃, ±5℃ from -20℃ to +125℃

• Supports over-temperature protection interrupt and over-temperature alarm interrupt

• Averaging filter for thermal sensor reading

2.2.3.8. Crypto Engine

AES

• NIST FIPS 197

Vangotech

• Key size: 128-bit, 192-bit, and 256-bit

• Modes based on NIST SP 800-38: ECB, CTR, CBC, CFB, OFB, CBC-CS2, CMAC, CCM, GCM, and XTS

HASH

• FIPS 180-3: MD-5, SHA-1, SHA-256, SHA-224, SHA-384, and SHA-512

• FIPS 198: HMAC is supported for all algorithms except SM3

• 66 cycles per 512-bits blocks for MD5, SHA-224, and SHA-256

• 82 cycles per 512/1024-bits blocks for SHA-1, SHA-384, and SHA-512

TRNG

• FIPS 140-2 certification

• Pass NIST-800-22, NIST-800-90B, and AIS31

2.2.3.9. eFuse

• One on-chip 4Kbit eFuse

2.2.4. Display Subsystem

2.2.4.1. Video Output

• Programmable resolution of up to 2048x2048

• RGB serial output: 8-bit

• Pixel clock rate of up to 150MHz

2.2.5. External Peripherals

2.2.5.1. USB

• Up to three USB OTG interfaces

• USB 2.0 OTG, with integrated one USB 2.0 analog PHY

• Compatible with USB2.0 Specification

• Support High-Speed(HS,480Mbit/s),Full-Speed(FS,12Mbit/s),and Low-Speed(LS,1.5Mbit/s) in host mode

• Supports High-Speed (HS, 480Mbit/s), Full-Speed (FS, 12Mbit/s) in Device mode

• Up to 8 user-configurable endpoints for Bulk, Isochronous, Control and Interrupt transfer

2.2.5.2. MAC

• Up to three MAC interfaces

• Compliant with the IEEE 802.3-2002 standard

• Programmable frame length to support Standard or Jumbo Ethernet frames with size up to 9KB

• Supports 10/100/1000Mbps data transfer rates

• One RGMII port, one RMII port, and one MII port

• Supports a variety of flexible address filtering modes

• Supports full and half duplex operations

2.2.5.3. SPI

• Up to 9 independent SPI controllers

- One port for SPI flash with XIP, and prefetch
- One port for ESAM
- 7 ports for others, and can be master or slave

Vangotech

- Internally generated or externally supplied serial clock, and can be up to 100MHz
- Programmable frame/sync polarity
- Programmable serial clock polarity, phase, and frequency
- Programmable serial bit sequences (MSB or LSB first)
- Serial data length ranges from 4 bits to 128 bits
- Texas Instrument Synchronous Serial Port, Motorola Serial Peripheral Interface, National Semiconductor MICROWIRE, and LCD driver serial interface

2.2.5.4. UART

- Up to 10 UART controllers
  - Five ports without RTC/CTS
  - Three ports with RTC/CTS
  - Two ports with IrDA, and without RTC/CTS
- Compatible with industry-standard 16C550A UARTs
- Supports for word length from 5 to 8 bits,an optional parity bit,and 1,1.5 or 2 stop bits
- Programmable parity(even,odd and no parity)
- UART baud rate from 1200bps to 3686400bps
- IrDA 1.3 SIR with up to 115200bps
- IrDA 1.3 FIR 4000000bps

2.2.5.5. I2C

- Five I2C ports
- Mode: Standard, Fast, Fast+ (F/S), High-Speed (Hs)
- master-TX, master-RX, slave-TX, and slave-RX
- multi-master mode
- 7-bit, 10-bit

2.2.5.6. I2S

- Two I2S ports
- Can be reconfigured as SPI
- Can be master or slave
- Internally generated or externally supplied serial clock
- Programmable frame/sync polarity
- Programmable serial clock polarity, phase, and frequency
- Programmable serial bit sequences (MSB or LSB first)
- Serial data length ranges from 4 bits to 128 bits
- Philips I2S, Intel AC-link, SPDIF frame
- Programmable I2S format (Includes data padding bits and justification)

2.2.5.7. CAN

- Two CAN ports
- Bosch CAN 2.0 A/B and FD 1.0
- Arbitration bit rate up to 1Mbps
- Data bit rate up to 8 Mbps

Vangotech

2.2.5.8. Keypad

- One keypad matrix interface up to 8 rows and 16 columns
- Interrupt for key press or key release
- Internal debouncing filter to prevent switching noises

2.2.6. Package

- TFBGA 510 balls, 0.65mm ball pitch, 16mm x 16mm

Vangotech

2.3. Block Diagram

The follow figure shows the block diagram of the V8130L processor.

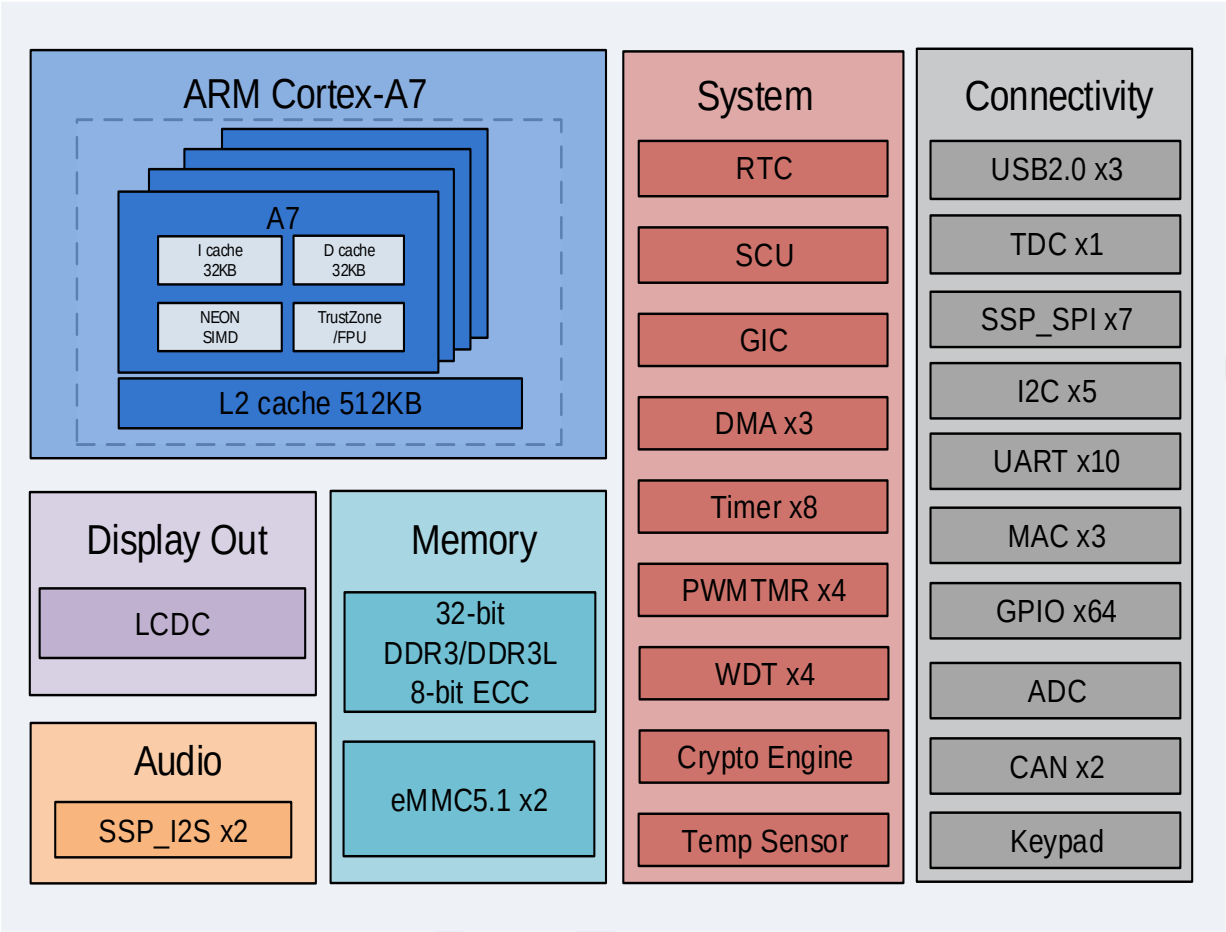


Figure 2-1. V8130L Block Diagram

Chapter 3. Pin Description

This part details the V8130L pin description from the following aspects:

- Pin Characteristics
- GPIO Multiplex Function
- PWM timer output signals
- Signal Descriptions

Vangotech

3.1. Pin Characteristics

Table 3-1 lists the characteristics of V8130L pins.

Table 3-1. Pin Characteristics

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	GND	X_DDR_RDRVUP	X_DDR_DQ_1[6]	X_DDR_DQ_1[4]		X_DDR_DM_1	X_DDR_DQ_1[7]		X_DDR_DQ_3[4]	X_DDR_DQS_3		X_DDR_DQ_3[1]	X_DDR_DQ_3[5]		X_DDR_DM_4	X_DDR_DQ_4[4]		X_MII_COL	X_MII_MDIO		X_MII_RXD[0]	X_MII_RXD[1]	X_RMII_TXD[1]	GND
B	X_DDR_RASN	X_DDR_RDRVDN	X_DDR_DQ_1[2]	X_DDR_DQ_1[0]	X_DDR_DQS_1	X_DDR_DQ_1[1]	X_DDR_DQ_1[5]	X_DDR_DQ_3[2]	X_DDR_DQ_3[6]	X_DDR_DQSB_3	X_DDR_DQ_3[0]	X_DDR_DM_3	X_DDR_DQ_3[3]	X_DDR_DQ_4[5]	X_DDR_DQ_4[1]	X_DDR_DQ_4[7]	X_DDR_DQ_4[6]	GND	X_MII_MDC	X_MII_RX_DV	X_MII_RX_CK	X_MII_RXD[2]	X_RMII_RXD[1]	X_RMII_TXD[0]
C	X_DDR_CKE[0]	X_DDR_CASN	X_DDR_VREF_D0	X_DDR_VREF_D1	X_DDR_DQSB_1	X_DDR_DQSB_0	X_DDR_DQ_1[3]	GND	X_DDR_VREF_D2	X_DDR_VREF_D3	GND	X_DDR_DQSB_2	X_DDR_DQ_3[7]	X_DDR_DQ_4[3]	GND	X_DDR_DQ_4[0]	X_DDR_DQ_4[2]	GND	X_MII_TXD[0]	X_MII_RX_ER	X_MII_TXD[3]	X_RMII_RX_ER	X_RMII_RXD[0]	X_RMII_TX_EN
D	X_DDR_WEN	X_DDR_ADDR[10]	GND	X_DDR_DQ_0[7]		GND	X_DDR_DQS_0		GND	X_DDR_DQ_2[7]		GND	X_DDR_DQS_2		X_DDR_DQS_4	X_DDR_VREF_D4		X_MII_CRS	X_MII_TX_EN		X_UART0_RX	X_RMII_REF_CLK	X_RMII_MDIO	X_RMII_RX_CRSDV
E		X_DDR_CSN[0]	X_DDR_DQ_0[5]	X_DDR_DQ_0[0]		X_DDR_DQ_0[4]	X_DDR_DQ_0[6]		X_DDR_DQ_2[5]	X_DDR_DQ_2[0]		X_DDR_DQ_2[4]	X_DDR_DQ_2[6]		X_DDR_DQSB_4	GND		X_MII_TX_ER	X_MII_TXD[1]		X_I2C2_CLK	X_UART0_TX	X_RMII_MDC	

X_RGMII_MDIO	X_RGMII_RXD[1]		X_RGMII_TXD[3]	X_HSPI_CLK	
X_RGMII_GTX_CK	X_RGMII_RXD[2]	X_RGMII_TXD[1]	X_RGMII_TXD[2]	X_HSPI_DI	X_SPI1_CLK
X_RGMII_RXCTL	X_RGMII_MDC	X_RGMII_TXD[0]	X_RGMII_TXCTL	X_SPI5_DO	X_SPI5_CS
X_I2C2_DATA	X_RGMII_RX_CK		X_RGMII_RXD[3]	GND	
GND	X_RGMII_RXD[0]		X_SPI1_CS	X_SPI6_CS	
X_MII_TXD[2]	X_HSPI_DO	X_HSPI_CS	X_SPI1_DO	X_SPI6_DO	X_SPI4_DO
X_MII_TX_CK	VCC3IO	VCC3IO	VCCK	VCCK	VCC3IO
GND	VCC15O_DDR	VCC15O_DDR	GND	GND	GND
X_MII_RXD[3]	VCC15O_DDR	VCC15O_DDR	GND	GND	GND
GND	VCC15O_DDR	VCC15O_DDR	GND	VCCK	GND
GND	VCC15O_DDR	GND	GND	VCCK	VCCK
X_DDR_DQ_2[2]	VCC15O_DDR	VCC15O_DDR	GND	GND	GND
X_DDR_DQ_2[1]	VCC15O_DDR	VCC15O_DDR	GND	VCCK	VCCK
GND	VCC15O_DDR	VCC15O_DDR	GND	GND	GND
X_DDR_DQ_2[3]	VCC15O_DDR	GND	GND	VCCK	VCCK
X_DDR_DM_2	VCC15O_DDR	VCC15O_DDR	GND	GND	GND
GND	VCC15O_DDR	VCC15O_DDR	GND	VCCK	VCCK
X_DDR_DQ_0[2]	VCC15O_DDR	VCC15O_DDR	GND	GND	VCC15O_DDRCK
X_DDR_DQ_0[1]	GND	X_DDR_ODT[1]	X_DDR_ODT[0]	X_DDR_ADDR[7]	X_DDR_BA[0]
GND	X_DDR_CKE[1]		X_DDR_ADDR[9]	X_DDR_RESETN	
X_DDR_DQ_0[3]	X_DDR_ADDR[2]		GND	X_DDR_ADDR[5]	
X_DDR_DM_0	X_DDR_ADDR[0]	X_DDR_BA[1]	X_DDR_ADDR[13]	X_DDR_BA[2]	X_DDR_ADDR[8]
X_DDR_CK	X_DDR_CSN[1]	X_DDR_ADDR[3]	X_DDR_ADDR[1]	X_DDR_ADDR[14]	X_DDR_ADDR[6]
X_DDR_CKB	X_DDR_ADDR[15]		X_DDR_ADDR[11]	X_DDR_ADDR[4]	
F	G	H	J	K	L

X_SPI1_DI	X_SPI6_CLK		X_SPI5_DI	X_SPI4_DI
X_SPI6_DI	X_SPI5_CLK	X_SPI4_CLK	X_SPI0_DI	X_JTAG_TDI
X_I2S_MCLK	X_SPI0_CLK	X_SPI0_DO	X_SPI0_CS	X_UART3_RX
X_I2S0_RXD	GND		X_UART4_TX	X_I2C0_CLK
X_SPI4_CS	X_JTAG_TDO		X_I2C3_CLK	X_UART3_TX
X_JTAG_SWDITMS	X_SPI3_DO	X_SPI3_CS	X_UART3_NRTS	X_SPI_HOLD_N
VCC3IO	VCCK	VCCK	VCC3IO	VCC3IO
GND	GND	GND	GND	VCC3IO
GND	GND	GND	GND	VCC3IO
GND	GND	GND	GND	VCCK
GND	GND	GND	GND	VCCK
GND	GND	GND	GND	VCC3IO
GND	GND	GND	GND	VCC3IO
GND	GND	GND	GND	VCCK
GND	GND	GND	GND	VCCK
GND	GND	GND	VCCK	VCCK
VCC11A_PLL_DDR	GND11A_PLL04	GND33A_PLL_USB2	VCC33A_PLL_USB2	GND33A_PLL_USB1
GND11A_PLL_DDR	VCC11A_PLL04	GND33A_HSRT_USB2	VCC33A_HSRT_USB2	GND33A_HSRT_USB1
X_DDR_ADDR[12]	GND	VCC11K_RTC	X_UART5_TX	X_PSW_0
GND	VCC3IO_RTC		X_UART6_RX	X_UART8_RX_H
VCC3IO_RTCOSC	X_SPI_DCX1		X_PSW_DDRCK	X_UART8_RX
GND	X_OM	X_UART9_TX	X_UART9_RX	X_UART5_RX
GND	X_RESET_N	X_SPI_DCX2	X_UART6_TX	X_UART7_RX
X_RTC_XTAL_IN	X_RTC_XTAL_IO		X_UART5_RX_H	X_UART7_TX
M	N	P	R	T

	X_JTAG_TRST_N	X_I2C0_DATA		X_I2S0_FS	X_I2S0_SCLK
X_SPI3_CLK	X_SPI3_DI	X_UART3_NCTS	X_SPI_WP_N	X_I2S0_TXD	GND
X_JTAG_SWCLKTCK	X_UART2_NRTS	X_UART2_RX	X_SPI_DO	X_SPI_CS_N	X_USB0_DRVBUS
	GND	X_UART1_RX	X_SPI7_DO	X_SPI7_CS	GND
	X_UART2_NCTS	X_SPI_DI			X_EMMC0_RSTN
X_SPI_CLK	X_USB0_VBUS	GND	X_EMMC1_CMD	X_EMMC1_D3	VCC3IO_EMMC0
GND11A_DLL_1	GND11A_DLL_0	X_EMMC1_CLK	X_EMMC1_D1	X_EMMC1_D5	X_EMMC0_CLK
VCC11A_DLL_1	VCC11A_DLL_0	X_EMMC1_D0			X_EMMC0_D1
GND	GND	X_EMMC1_D2	X_EMMC1_D4	X_EMMC1_D6	X_EMMC1_RSTN
VCC33A_PLL_USB0	VCC33A_HSRT_USB0	GND	GND	VCC3IO_EMMC1	GND33A_HSRT_USB0
GND33A_HSRT_USB0	GND33A_PLL_USB0	GND			X_USB0_DM
VCC33IO_ADC	GND33IO_ADC	VCC33A_TDC	GND33A_TDC	VCC3IO_EFUSE	
GND33A_ADC	VCC33A_ADC	GND33D_ADC	VCC33D_ADC	GND33A_ADC	X_ADC_XAIN[6]
GND11A_PLL35	VCC11A_PLL35	GND			X_ADC_XVRT
GND	GND	X_USB1_DRVBUS	GND	GND33A_ADC	X_ADC_XAIN[2]
GND11A_PLL126	VCC11A_PLL126	GND			GND33A_ADC
VCC33A_PLL_USB1	GND	X_USB2_DRVBUS	X_LC_HS	X_LC_VS	X_LC_DE
VCC33A_HSRT_USB1	VCC3IO_OSC	X_USB1_VBUS	X_LC_PCLK	X_LC_DATA[23]	X_LC_DATA[16]
GND	X_I2C1_CLK	X_USB2_VBUS	X_LC_DATA[19]	X_LC_DATA[20]	X_LC_DATA[22]
	X_I2C1_DATA	X_LC_DATA[18]			X_LC_DATA[12]
	GND	X_SPI2_CS	X_LC_DATA[2]	X_LC_DATA[3]	X_LC_DATA[9]
X_UART8_TX	GND	GND	X_SPI2_DO	X_LC_DATA[0]	X_LC_DATA[4]
GND	X_USB2_DM	X_USB1_DM	GND	X_SPI2_DI	X_LC_DATA[5]
	X_USB2_DP	X_USB1_DP		X_SPI2_CLK	X_LC_DATA[1]
U	V	W	Y	AA	AB

X_I2C3_DATA	GND
X_UART4_RX	X_UART2_TX
X_UART1_TX	X_SPI7_CLK
GND	X_SPI7_DI
X_EMMC0_D2	
X_EMMC0_D6	X_EMMC0_D0
X_EMMC1_D7	X_EMMC0_D4
X_EMMC0_D7	
X_EMMC0_D5	X_EMMC0_D3
GND33A_HSRT_USB0	X_EMMC0_CMD
X_USB0_DP	
	X_ADC_XAIN[7]
X_ADC_XAIN[4]	X_ADC_XAIN[5]
X_ADC_XAIN[3]	
X_ADC_XAIN[1]	X_ADC_XVRB
GND33A_ADC	X_ADC_XAIN[0]
GND	
X_LC_DATA[21]	X_LC_DATA[17]
X_LC_DATA[14]	X_LC_DATA[15]
X_LC_DATA[13]	
X_LC_DATA[11]	X_LC_DATA[10]
X_LC_DATA[7]	X_LC_DATA[8]
X_SYS_XTAL_IO	X_LC_DATA[6]
X_SYS_XTAL_IN	GND
A C	**A D**

Vangotech

3.2. GPIO Multiplex Function

The following table provides a description of the V8130L GPIO multiplex function.

Table 3-2. GPIO Multiplex Function

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_SPI_CS_N	SPI_CS_N								
X_SPI_CLK	SPI_CLK								
X_SPI_DO	SPI_DO								
X_SPI_DI	SPI_DI								
X_SPI_WP_N	SPI_WP_N								
X_SPI_HOLD_N	SPI_HOLD_N								
X_EMMC0_D7	SD_DAT7								
X_EMMC0_D6	SD_DAT6								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_EMMC0_D5	SD_DAT5								
X_EMMC0_D4	SD_DAT4								
X_EMMC0_D3	SD_DAT3								
X_EMMC0_D2	SD_DAT2								
X_EMMC0_D1	SD_DAT1								
X_EMMC0_D0	SD_DAT0								
X_EMMC0_RSTN	SD_RSTn								
X_EMMC0_CMD	SD_CMD								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_EMMC0_CLK	SD_CLK								
GNDK	GNDK								
VCC11K	VCC11K								
X_OSC_IN	X_OSC_IN								
X_OSC_IO	X_OSC_IO								
GNDIO_RTC	GNDIO_RTC								
VCC3IO_RTC	VCC3IO_RTC								
X_OSCRTC_IN	X_OSCRTC_IN								
X_OSCRTC_IO	X_OSCRTC_IO								
X_RESET_N	X_RESET_N								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_PSW_0	X_PSW_0								
X_PSW_DDRCK	X_PSW_DDRCK								
X_OM	X_OM								
X_USB0_DRVBUS	USB0_DRVBUS								
X_USB0_VBUS	USB0_VBUS								
X_USB1_DRVBUS	USB1_DRVBUS								
X_USB1_VBUS	USB1_VBUS								
X_USB2_DRVBUS	USB2_DRVBUS								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_USB2_VBUS	USB2_VBUS								
X_JTAG_TRST_N	JTAG_TRST_N								
X_JTAG_TDI	JTAG_TDI								
X_JTAG_SWDITMS	JTAG_SWDITMS								
X_JTAG_SWCLKTCK	JTAG_SWCLKTCK								
X_JTAG_TDO	JTAG_TDO								
X_I2C0_CLK	I2C0_CLK								
X_I2C0_DATA	I2C0_DATA								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_I2C1_CLK	I2C1_CLK								
X_I2C1_DATA	I2C1_DATA								
X_I2C2_CLK	I2C2_CLK								
X_I2C2_DATA	I2C2_DATA								
X_I2C3_CLK	I2C3_CLK								
X_I2C3_DATA	I2C3_DATA								
X_HSPI_CLK	HSPI_CLK								
X_HSPI_CS	HSPI_CS								
X_HSPI_DI	HSPI_DI								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_HSPI_DO	HSPI_DO								
X_SPI0_CLK	SPI0_CLK								
X_SPI0_CS	SPI0_CS								
X_SPI0_DI	SPI0_DI	SPI0_DO							
X_SPI0_DO	SPI0_DO	SPI0_DI							
X_SPI1_CLK	SPI1_CLK								
X_SPI1_CS	SPI1_CS								
X_SPI1_DI	SPI1_DI	SPI1_DO							
X_SPI1_DO	SPI1_DO	SPI1_DI							
X_SPI2_CLK	SPI2_CLK								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_SPI2_CS	SPI2_CS								
X_SPI2_DI	SPI2_DI	SPI2_DO							
X_SPI2_DO	SPI2_DO	SPI2_DI							
X_SPI3_CLK	SPI3_CLK								
X_SPI3_CS	SPI3_CS								
X_SPI3_DI	SPI3_DI	SPI3_DO							
X_SPI3_DO	SPI3_DO	SPI3_DI							
X_SPI4_CLK	SPI4_CLK								
X_SPI4_CS	SPI4_CS								
X_SPI4_DI	SPI4_DI	SPI4_DO							

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_SPI4_DO	SPI4_DO	SPI4_DI							
X_SPI5_CLK	SPI5_CLK								
X_SPI5_CS	SPI5_CS								
X_SPI5_DI	SPI5_DI	SPI5_DO							
X_SPI5_DO	SPI5_DO	SPI5_DI							
X_SPI6_CLK	SPI6_CLK								
X_SPI6_CS	SPI6_CS								
X_SPI6_DI	SPI6_DI	SPI6_DO							
X_SPI6_DO	SPI6_DO	SPI6_DI							
X_SPI7_CLK	SPI7_CLK								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_SPI7_CS	SPI7_CS								
X_SPI7_DI	SPI7_DI	SPI7_DO							
X_SPI7_DO	SPI7_DO	SPI7_DI							
X_UART0_TX	UART0_SOUT								
X_UART0_RX	UART0_SIN								
X_UART1_TX	UART1_SOUT								
X_UART1_RX	UART1_SIN								
X_UART2_TX	UART2_SOUT								
X_UART2_RX	UART2_SIN								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_UART2_NRTS	UART2_NRTS								GPIO0_19
X_UART2_NCTS	UART2_NCTS								GPIO0_20
X_UART3_TX	UART3_SOUT								
X_UART3_RX	UART3_SIN								
X_UART3_NRTS	UART3_NRTS								
X_UART3_NCTS	UART3_NCTS								
X_UART4_TX	UART4_SOUT								
X_UART4_RX	UART4_SIN								

Vangotech

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_I2S0_FS	I2S0_FS	I2S0_FS	GPIO1_11	I2S0_FS	I2S0_FS	PWM_1	X_CODEX_0	I2S0_FS	GPIO0_25
X_I2S0_RXD	I2S0_RXD	I2S0_RXD	GPIO1_12	I2S0_RXD	I2S0_RXD	PWM_2	X_CODEX_1	I2S0_RXD	GPIO0_22
X_I2S0_SCLK	I2S0_SCLK	I2S0_SCLK	GPIO1_13	I2S0_SCLK	I2S0_SCLK	PWM_3	X_CODEX_2	I2S0_SCLK	GPIO0_23
X_I2S0_TXD	I2S0_TXD	I2S0_TXD	GPIO1_14	I2S0_TXD	I2S0_TXD	PWM_4	X_CODEX_3	I2S0_TXD	GPIO0_18
X_UART5_TX	UART5_irda_TX	UART5_irda_TX	CAN0_TX	GPIO0_0	UART5_irda_TX	UART5_irda_TX	X_CODEY_0	UART5_irda_TX	GPIO0_2
X_UART5_RX	UART5_irda_RX_l	UART5_irda_RX_l	CAN0_RX	GPIO0_1	UART5_irda_RX_l	UART5_irda_RX_l	X_CODEY_1	UART5_irda_RX_l	GPIO0_3
X_UART5_RX_h	UART5_irda_RX_h	UART5_irda_RX_h	PWM_1	GPIO0_2	UART5_irda_RX_h	UART5_irda_RX_h	X_CODEY_2	UART5_irda_RX_h	
X_UART6_TX	UART6_irda_TX	UART6_irda_TX	UART6_irda_TX	GPIO0_3	UART6_irda_TX	UART6_irda_TX	X_CODEY_3	UART6_irda_TX	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_UART6_RX	UART6_irda_RX_l	UART6_irda_RX_l	UART6_irda_RX_l	GPIO0_4	UART6_irda_RX_l	UART6_irda_RX_l	X_CODEY_4	UART6_irda_RX_l	
X_UART9_TX	UART9_irda_TX	UART9_irda_TX	UART9_irda_TX	GPIO0_5	UART9_irda_TX	UART9_irda_TX	X_CODEY_5	UART9_irda_TX	
X_UART7_TX	UART7_irda_TX	UART7_irda_TX	UART7_irda_TX	GPIO0_6	UART7_irda_TX	UART7_irda_TX	X_CODEY_6	UART7_irda_TX	GPIO0_29
X_UART7_RX	UART7_irda_RX_l	UART7_irda_RX_l	UART7_irda_RX_l	GPIO0_7	UART7_irda_RX_l	UART7_irda_RX_l	X_CODEY_7	UART7_irda_RX_l	GPIO0_30
X_UART9_RX	UART9_irda_RX	UART9_irda_RX	UART9_irda_RX	GPIO0_8	UART9_irda_RX	UART9_irda_RX	GPIO1_11	UART9_irda_RX	
X_UART8_TX	UART8_irda_TX	UART8_irda_TX	CAN1_TX	GPIO0_9	UART8_irda_TX	UART8_irda_TX	GPIO1_12	UART8_irda_TX	UART8_TX
X_UART8_RX	UART8_irda_RX_l	UART8_irda_RX_l	CAN1_RX	GPIO0_10	UART8_irda_RX_l	UART8_irda_RX_l	GPIO1_13	UART8_irda_RX_l	UART8_RX

Preliminary

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_UART8_RX_h	UART8_irda_RX_h	UART8_irda_RX_h	PWM_2	GPIO0_11	UART8_irda_RX_h	UART8_irda_RX_h	GPIO1_14	UART8_irda_RX_h	
X_RGMII_RX_CK	RGMII0_RX_CK	X_CODEX_0	X_CODEX_0	X_CODEX_0	X_CODEX_0	GPIO1_15	RGMII0_RX_CK	GPIO1_30	SPI6_MISO
X_RGMII_RXD[0]	RGMII0_RXD[0]	X_CODEX_1	X_CODEX_1	X_CODEX_1	X_CODEX_1	GPIO1_16	RGMII0_RXD[0]	GPIO1_31	UART9_RX
X_RGMII_RXD[1]	RGMII0_RXD[1]	X_CODEX_2	X_CODEX_2	X_CODEX_2	X_CODEX_2	GPIO1_17	RGMII0_RXD[1]	GPIO1_17	UART9_TX
X_RGMII_RXD[2]	RGMII0_RXD[2]	X_CODEX_3	X_CODEX_3	X_CODEX_3	X_CODEX_3	GPIO1_18	RGMII0_RXD[2]	GPIO1_18	
X_RGMII_RXD[3]	RGMII0_RXD[3]	X_CODEX_4	X_CODEX_4	X_CODEX_4	X_CODEX_4	GPIO1_19	RGMII0_RXD[3]	GPIO1_19	
X_RGMII_RXCTL	RGMII0_RXCTL	X_CODEX_5	X_CODEX_5	X_CODEX_5	X_CODEX_5	GPIO1_20	RGMII0_RXCTL	GPIO1_20	UART6_TX

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_RGMII_GTX_CK	RGMII0_GTX_CK	X_CODEX_6	X_CODEX_6	X_CODEX_6	X_CODEX_6	GPIO1_21	RGMII0_GTX_CK	GPIO1_21	
X_RGMII_TXCTL	RGMII0_TXCTL	X_CODEX_7	X_CODEX_7	X_CODEX_7	X_CODEX_7	GPIO1_22	RGMII0_TXCTL	GPIO1_22	
X_RGMII_TXD[0]	RGMII0_TXD[0]	GPIO1_19	GPIO1_19	PWM_2	GPIO1_19	PWM_2	RGMII0_TXD[0]	GPIO1_19	SPI6_MOSI
X_RGMII_TXD[1]	RGMII0_TXD[1]	GPIO1_20	GPIO1_20	PWM_3	GPIO1_20	PWM_3	RGMII0_TXD[1]	PWM_3	SPI6_CLK
X_RGMII_TXD[2]	RGMII0_TXD[2]	GPIO1_21	GPIO1_21	PWM_4	GPIO1_21	PWM_4	RGMII0_TXD[2]	PWM_4	SPI6_CS
X_RGMII_TXD[3]	RGMII0_TXD[3]	GPIO1_22	GPIO1_22	PWM_5	GPIO1_22	PWM_5	RGMII0_TXD[3]	PWM_5	SPI6_CS2
X_RGMII_MDC	RGMII0_MDC	GPIO1_23	GPIO1_23	PWM_6	GPIO1_23	PWM_6	RGMII0_MDC	PWM_6	UART6_RX

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_RGMII_MDIO	RGMII0_MDIO	GPIO1_24	GPIO1_24	PWM_7	GPIO1_24	PWM_7	RGMII0_MDIO	PWM_7	
X_RMII_TXD[0]	RMII0_TXD[0]	RMII0_TXD[0]	X_CODEY_0	CAN0_TX	RMII0_TXD[0]	PWM_8	RMII0_TXD[0]	GPIO1_05	
X_RMII_TXD[1]	RMII0_TXD[1]	RMII0_TXD[1]	X_CODEY_1	CAN0_RX	RMII0_TXD[1]	PWM_9	RMII0_TXD[1]	GPIO1_06	
X_RMII_TX_EN	RMII0_TX_EN	RMII0_TX_EN	X_CODEY_2	CAN1_TX	RMII0_TX_EN	PWM_10	RMII0_TX_EN	GPIO1_07	
X_RMII_RX_CRSDV	RMII0_RX_CRS_DV	RMII0_RX_CRS_DV	X_CODEY_3	CAN1_RX	RMII0_RX_CRS_DV	PWM_11	RMII0_RX_CRS_DV	GPIO1_08	
X_RMII_RX_ER	RMII0_RX_ER	RMII0_RX_ER	X_CODEY_4	PWM_12	RMII0_RX_ER	PWM_12	RMII0_RX_ER	GPIO1_09	
X_RMII_RXD[0]	RMII0_RXD[0]	RMII0_RXD[0]	X_CODEY_5	PWM_13	RMII0_RXD[0]	PWM_13	RMII0_RXD[0]	GPIO1_10	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_RMII_RXD[1]	RMII0_RXD[1]	RMII0_RXD[1]	X_CODEY_6	GPIO1_11	RMII0_RXD[1]	GPIO1_11	RMII0_RXD[1]	GPIO1_11	GPIO0_26
X_RMII_REF_CLK	RMII0_REF_CLK	RMII0_REF_CLK	X_CODEY_7	GPIO1_12	RMII0_REF_CLK	GPIO1_12	RMII0_REF_CLK	GPIO1_12	
X_RMII_MDC	RMII0_MDC	RMII0_MDC	GPIO1_25	GPIO1_13	RMII0_MDC	GPIO1_13	RMII0_MDC	GPIO1_13	
X_RMII_MDIO	RMII0_MDIO	RMII0_MDIO	PWM_2	GPIO1_14	RMII0_MDIO	GPIO1_14	RMII0_MDIO	GPIO1_14	
X_MII_TXD[0]	MII_TXD[0]	X_CODEY_0	MII_TXD[0]	X_CODEY_0	X_CODEY_0	MII_TXD[0]	MII_TXD[0]	GPIO1_23	
X_MII_TXD[1]	MII_TXD[1]	X_CODEY_1	MII_TXD[1]	X_CODEY_1	X_CODEY_1	MII_TXD[1]	MII_TXD[1]	GPIO1_24	
X_MII_TX_EN	MII_TX_EN	X_CODEY_2	MII_TX_EN	X_CODEY_2	X_CODEY_2	MII_TX_EN	MII_TX_EN	GPIO1_25	GPIO0_24
X_MII_RX_DV	MII_RX_CRS_DV	X_CODEY_3	MII_RX_CRS_DV	X_CODEY_3	X_CODEY_3	MII_RX_CRS_DV	MII_RX_CRS_DV	GPIO1_26	

Preliminary

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_MII_RX_ER	MII_RX_ER	X_CODEY_4	MII_RX_ER	X_CODEY_4	X_CODEY_4	MII_RX_ER	MII_RX_ER	GPIO1_27	
X_MII_RXD[0]	MII_RXD[0]	X_CODEY_5	MII_RXD[0]	X_CODEY_5	X_CODEY_5	MII_RXD[0]	MII_RXD[0]	GPIO1_28	
X_MII_RXD[1]	MII_RXD[1]	X_CODEY_6	MII_RXD[1]	X_CODEY_6	X_CODEY_6	MII_RXD[1]	MII_RXD[1]	GPIO1_29	
X_MII_MDC	MII_MDC	GPIO1_25	MII_MDC	GPIO1_25	PWM_2	MII_MDC	MII_MDC	GPIO0_31	
X_MII_MDIO	MII_MDIO	GPIO1_26	MII_MDIO	GPIO1_26	PWM_3	MII_MDIO	MII_MDIO	GPIO1_0	
X_MII_RX_CK	MII_RX_CK	GPIO0_09	MII_RX_CK	GPIO0_09	GPIO0_09	MII_RX_CK	MII_RX_CK	GPIO1_1	
X_MII_RXD[2]	MII_RXD[2]	GPIO0_10	MII_RXD[2]	GPIO0_10	GPIO0_10	MII_RXD[2]	MII_RXD[2]	GPIO1_2	
X_MII_RXD[3]	MII_RXD[3]	GPIO0_11	MII_RXD[3]	GPIO0_11	GPIO0_11	MII_RXD[3]	MII_RXD[3]	GPIO1_3	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_MII_TX_CK	MII_TX_CK	GPIO0_12	MII_TX_CK	GPIO0_12	GPIO0_12	MII_TX_CK	MII_TX_CK	GPIO1_4	
X_MII_TX_ER	MII_TX_ER	GPIO0_13	MII_TX_ER	GPIO0_13	GPIO0_13	MII_TX_ER	MII_TX_ER	GPIO1_15	
X_MII_TXD[2]	MII_TXD[2]	GPIO0_14	MII_TXD[2]	GPIO0_14	GPIO0_14	MII_TXD[2]	MII_TXD[2]	GPIO1_16	
X_MII_TXD[3]	MII_TXD[3]	GPIO0_15	MII_TXD[3]	GPIO0_15	GPIO0_15	MII_TXD[3]	MII_TXD[3]	GPIO1_17	
X_MII_CRS	MII_CRS	GPIO0_16	MII_CRS	GPIO0_16	GPIO0_16	MII_CRS	MII_CRS	GPIO1_18	
X_MII_COL	MII_COL	GPIO0_17	MII_COL	GPIO0_17	GPIO0_17	MII_COL	MII_COL	GPIO1_19	
X_EMMC1_D7	SD1_DAT7	GPIO1_0	I2S1_FS	GPIO1_0	GPIO1_0	GPIO1_0	SD1_DAT7	GPIO1_0	
X_EMMC1_D6	SD1_DAT6	GPIO1_1	I2S1_RXD	GPIO1_1	GPIO1_1	GPIO1_1	SD1_DAT6	GPIO1_1	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_EMMC1_D5	SD1_DAT5	GPIO1_2	I2S1_SCLK	GPIO1_2	GPIO1_2	GPIO1_2	SD1_DAT5	GPIO1_2	
X_EMMC1_D4	SD1_DAT4	GPIO1_3	I2S1_TXD	GPIO1_3	GPIO1_3	GPIO1_3	SD1_DAT4	GPIO1_3	
X_EMMC1_D3	SD1_DAT3	GPIO1_4	I2C4_CLK	GPIO1_4	GPIO1_4	GPIO1_4	SD1_DAT3	GPIO1_4	
X_EMMC1_D2	SD1_DAT2	GPIO1_5	I2C4_DATA	GPIO1_5	GPIO1_5	GPIO1_5	SD1_DAT2	GPIO1_5	
X_EMMC1_D1	SD1_DAT1	GPIO1_6	UART4_NRTS	GPIO1_6	GPIO1_6	GPIO1_6	SD1_DAT1	GPIO1_6	
X_EMMC1_D0	SD1_DAT0	GPIO1_7	UART4_NCTS	GPIO1_7	GPIO1_7	GPIO1_7	SD1_DAT0	GPIO1_7	
X_EMMC1_RSTN	SD1_RSTn	GPIO1_8	SPI6_CS2	GPIO1_8	GPIO1_8	GPIO1_8	SD1_RSTn	GPIO1_8	
X_EMMC1_CMD	SD1_CMD	GPIO1_9	SPI6_CS3	GPIO1_9	GPIO1_9	GPIO1_9	SD1_CMD	GPIO1_9	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_EMMC1_CLK	SD1_CLK	GPIO1_10	PWM_3	GPIO1_10	GPIO1_10	GPIO1_10	SD1_CLK	GPIO1_10	
X_LC_PCLK	LC_PCLK	LC_PCLK	CAN0_TX	LC_PCLK	LC_PCLK	GPIO0_0	LC_PCLK	GPIO0_0	
X_LC_VS	LC_VS	LC_VS	CAN0_RX	LC_VS	LC_VS	GPIO0_1	LC_VS	GPIO0_1	
X_LC_HS	LC_HS	LC_HS	CAN1_TX	LC_HS	LC_HS	GPIO0_2	LC_HS	GPIO0_2	
X_LC_DE	LC_DE	LC_DE	CAN1_RX	LC_DE	LC_DE	GPIO0_3	LC_DE	GPIO0_3	
X_LC_DATA[0]	LC_DATA[0]	LC_DATA[0]	PWM_0	LC_DATA[0]	LC_DATA[0]	GPIO0_4	LC_DATA[0]	GPIO0_4	
X_LC_DATA[1]	LC_DATA[1]	LC_DATA[1]	PWM_1	LC_DATA[1]	LC_DATA[1]	GPIO0_5	LC_DATA[1]	GPIO0_5	
X_LC_DATA[2]	LC_DATA[2]	LC_DATA[2]	PWM_2	LC_DATA[2]	LC_DATA[2]	GPIO0_6	LC_DATA[2]	GPIO0_6	
X_LC_DATA[3]	LC_DATA[3]	LC_DATA[3]	PWM_3	LC_DATA[3]	LC_DATA[3]	GPIO0_7	LC_DATA[3]	GPIO0_7	

Vango Technologies

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_LC_DATA[4]	LC_DATA[4]	LC_DATA[4]	I2S1_FS	LC_DATA[4]	LC_DATA[4]	GPIO0_8	LC_DATA[4]	GPIO0_8	
X_LC_DATA[5]	LC_DATA[5]	LC_DATA[5]	I2S1_RXD	LC_DATA[5]	LC_DATA[5]	GPIO0_9	LC_DATA[5]	GPIO0_9	GPIO0_21
X_LC_DATA[6]	LC_DATA[6]	LC_DATA[6]	I2S1_SCLK	LC_DATA[6]	LC_DATA[6]	GPIO0_10	LC_DATA[6]	GPIO0_10	GPIO0_17
X_LC_DATA[7]	LC_DATA[7]	LC_DATA[7]	I2S1_TXD	LC_DATA[7]	LC_DATA[7]	GPIO0_11	LC_DATA[7]	GPIO0_11	
X_LC_DATA[8]	LC_DATA[8]	LC_DATA[8]	I2C4_CLK	LC_DATA[8]	LC_DATA[8]	GPIO0_12	LC_DATA[8]	GPIO0_12	
X_LC_DATA[9]	LC_DATA[9]	LC_DATA[9]	I2C4_DATA	LC_DATA[9]	LC_DATA[9]	GPIO0_13	LC_DATA[9]	GPIO0_13	
X_LC_DATA[10]	LC_DATA[10]	LC_DATA[10]	UART4_NRTS	LC_DATA[10]	LC_DATA[10]	GPIO0_14	LC_DATA[10]	GPIO0_14	
X_LC_DATA[11]	LC_DATA[11]	LC_DATA[11]	UART4_NCTS	LC_DATA[11]	LC_DATA[11]	GPIO0_15	LC_DATA[11]	GPIO0_15	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_LC_DATA[12]	LC_DATA[12]	LC_DATA[12]	SPI0_CS2	LC_DATA[12]	LC_DATA[12]	GPIO0_16	LC_DATA[12]	GPIO0_16	
X_LC_DATA[13]	LC_DATA[13]	LC_DATA[13]	SPI0_CS3	LC_DATA[13]	LC_DATA[13]	GPIO0_17	LC_DATA[13]	GPIO0_17	
X_LC_DATA[14]	LC_DATA[14]	LC_DATA[14]	SPI1_CS2	LC_DATA[14]	LC_DATA[14]	GPIO0_18	LC_DATA[14]	GPIO0_18	
X_LC_DATA[15]	LC_DATA[15]	LC_DATA[15]	SPI1_CS3	LC_DATA[15]	LC_DATA[15]	GPIO0_19	LC_DATA[15]	GPIO0_19	
X_LC_DATA[16]	LC_DATA[16]	LC_DATA[16]	SPI2_CS2	LC_DATA[16]	LC_DATA[16]	GPIO0_20	LC_DATA[16]	GPIO0_20	
X_LC_DATA[17]	LC_DATA[17]	LC_DATA[17]	SPI2_CS3	LC_DATA[17]	LC_DATA[17]	GPIO0_21	LC_DATA[17]	GPIO0_21	
X_LC_DATA[18]	LC_DATA[18]	LC_DATA[18]	SPI3_CS2	LC_DATA[18]	LC_DATA[18]	GPIO0_22	LC_DATA[18]	GPIO0_22	

Vango Confidential

PAD	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mode7	ModeX
X_LC_DATA[19]	LC_DATA[19]	LC_DATA[19]	SPI3_CS3	LC_DATA[19]	LC_DATA[19]	GPIO0_23	LC_DATA[19]	GPIO0_23	
X_LC_DATA[20]	LC_DATA[20]	LC_DATA[20]	SPI4_CS2	LC_DATA[20]	LC_DATA[20]	GPIO0_24	LC_DATA[20]	GPIO0_24	
X_LC_DATA[21]	LC_DATA[21]	LC_DATA[21]	SPI4_CS3	LC_DATA[21]	LC_DATA[21]	GPIO0_25	LC_DATA[21]	GPIO0_25	
X_LC_DATA[22]	LC_DATA[22]	LC_DATA[22]	SPI5_CS2	LC_DATA[22]	LC_DATA[22]	GPIO0_26	LC_DATA[22]	GPIO0_26	
X_LC_DATA[23]	LC_DATA[23]	LC_DATA[23]	SPI5_CS3	LC_DATA[23]	LC_DATA[23]	GPIO0_27	LC_DATA[23]	GPIO0_27	
X_I2S_MCLK	I2S_MCLK	I2S_MCLK	GPIO0_28	I2S_MCLK	I2S_MCLK	I2S_MCLK	GPIO0_28	GPIO0_28	
X_SPI_DCX1	GPIO0_29	SPI0_DCX	SPI2_DCX	SPI4_DCX	SPI6_DCX	GPIO0_29	GPIO0_29	GPIO0_29	
X_SPI_DCX2	GPIO0_30	SPI1_DCX	SPI3_DCX	SPI5_DCX	GPIO0_30	GPIO0_30	GPIO0_30	GPIO0_30	

The RGB output can be multiplexed to BT656 output, so the pin correspondence between LCD0 and BT656 is as follows.

Vangotech

Pin Name	LCD Pin	BT656 Pin
PD3	LCD0_D3	VD0
PD4	LCD0_D4	VD1
PD5	LCD0_D5	VD2
PD6	LCD0_D6	VD3
PD7	LCD0_D7	VD4
PD10	LCD0_D10	VD5
PD11	LCD0_D11	VD6
PD12	LCD0_D12	VD7

Note: Please never switch pin-mux mode for different pad to the same GPIO bit, or same functional pin.

3.3. PWM timer output signals

The PWM timer output signals can be mapped to Leo dedicated instance by the table list as below.

	FTTMR010 instance	Timer Output Signal
PWM_0	u_FTPWMTMR010	tmr1_out
PWM_1	u_FTPWMTMR010	tmr2_out
PWM_2	u_FTPWMTMR010	tmr3_out
PWM_3	u_FTPWMTMR010	tmr4_out
PWM_4	u_FTPWMTMR010_u1	tmr1_out
PWM_5	u_FTPWMTMR010_u1	tmr2_out
PWM_6	u_FTPWMTMR010_u1	tmr3_out
PWM_7	u_FTPWMTMR010_u1	tmr4_out
PWM_8	u_FTPWMTMR010_u2	tmr1_out
PWM_9	u_FTPWMTMR010_u2	tmr2_out
PWM_10	u_FTPWMTMR010_u2	tmr3_out
PWM_11	u_FTPWMTMR010_u2	tmr4_out
PWM_12	u_FTPWMTMR010_u3	tmr1_out
PWM_13	u_FTPWMTMR010_u3	tmr2_out

3.4. Signal Descriptions

V8130L contains many peripheral interfaces. Many of the interfaces can multiplex up to eight functions. Pin-multiplexing configuration can refer to Table 3-1 and Table 3-2. Table 3-3 shows the detailed function description of every signal based on the different interface.

Table 3-3. Signal Descriptions

Pad Name	Description
SPIF	

Vangotech

Pad Name	Description
X_SPI_CLK	SPI Clock Signal
X_SPI_CS_N	SPI Chip Select Signal (active low)
X_SPI_DI	SPI Master Data In, Slave Data Out
X_SPI_DO	SPI Master Data Out, Slave Data In
X_SPI_HOLD_N	SPI Data In/Out
X_SPI_WP_N	SPI Data In/Out
EMMC0	
X_EMMC0_CLK	EMMC Clock Signal
X_EMMC0_CMD	EMMC Command Line
X_EMMC0_RSTN	EMMC Reset signal
X_EMMC0_DATA[7:0]	EMMC Data
RTC	
X_RTC_XTAL_IN	RTC XTAL In
X_RTC_XTAL_IO	RTC XTAL OUT
AON	
X_SYS_XTAL_IN	System XTAL In
X_SYS_XTAL_IO	System XTAL Out
X_RESET_N	Reset Signal
X_PSW_0	Power Swith Signal
X_PSW_DDRCK	Power Swith For DDR
X_OM	Entering Test Mode Signal
USB	
X_USB0_DRVBUS	USB0 Drive Vbus
X_USB0_VBUS	USB0 Voltage Bus
X_USB1_DRVBUS	USB1 Drive Vbus
X_USB1_VBUS	USB1 Voltage Bus
X_USB2_DRVBUS	USB2 Drive Vbus
X_USB2_VBUS	USB2 Voltage Bus
JTAG	
X_JTAG_TRST_N	JTAG Reset Signal
X_JTAG_TDI	JTAG Data Input
X_JTAG_SWDITMS	JTAG Mode Select Input
X_JTAG_TDO	JTAG Data Output
X_JTAG_SWCLKTCK	JTAG Clock Input
I2C	
X_I2C0_SCL	Serial Clock Line
X_I2C0_SDA	Serial Data Line
X_I2C1_SCL	Serial Clock Line
X_I2C1_SDA	Serial Data Line
X_I2C2_SCL	Serial Clock Line

Vangotech

Pad Name	Description
X_I2C2_SDA	Serial Data Line
X_I2C3_SCL	Serial Clock Line
X_I2C3_SDA	Serial Data Line
SPI2AHB	
X_HSPI_CLK	HSPI Clock Signal
X_HSPI_CS	HSPI Chip Select Signal
X_HSPI_DI	HSPI Data In
X_HSPI_DO	HSPI Data Out
SSP(SPI)	
X_SPI0_CLK	SPI Clock Signal
X_SPI0_CS	SPI Chip Select Signal (active low)
X_SPI0_DI	SPI Master Data In, Slave Data Out
X_SPI0_DO	SPI Master Data Out, Slave Data In
X_SPI1_CLK	SPI Clock Signal
X_SPI1_CS	SPI Chip Select Signal (active low)
X_SPI1_DI	SPI Master Data In, Slave Data Out
X_SPI1_DO	SPI Master Data Out, Slave Data In
X_SPI2_CLK	SPI Clock Signal
X_SPI2_CS	SPI Chip Select Signal (active low)
X_SPI2_DI	SPI Master Data In, Slave Data Out
X_SPI2_DO	SPI Master Data Out, Slave Data In
X_SPI3_CLK	SPI Clock Signal
X_SPI3_CS	SPI Chip Select Signal (active low)
X_SPI3_DI	SPI Master Data In, Slave Data Out
X_SPI3_DO	SPI Master Data Out, Slave Data In
X_SPI4_CLK	SPI Clock Signal
X_SPI4_CS	SPI Chip Select Signal (active low)
X_SPI4_DI	SPI Master Data In, Slave Data Out
X_SPI4_DO	SPI Master Data Out, Slave Data In
X_SPI5_CLK	SPI Clock Signal
X_SPI5_CS	SPI Chip Select Signal (active low)
X_SPI5_DI	SPI Master Data In, Slave Data Out
X_SPI5_DO	SPI Master Data Out, Slave Data In
X_SPI6_CLK	SPI Clock Signal
X_SPI6_CS	SPI Chip Select Signal (active low)
X_SPI6_DI	SPI Master Data In, Slave Data Out
X_SPI6_DO	SPI Master Data Out, Slave Data In
SPI (ESAM)	
X_SPI7_CLK	SPI Clock Signal
X_SPI7_CS	SPI Chip Select Signal (active low)

Vangotech

Pad Name	Description
X_SPI7_DI	SPI Master Data In, Slave Data Out
X_SPI7_DO	SPI Master Data Out, Slave Data In
SSP (I2S)	
X_I2S0_FS	I2S Frame/Sync.
X_I2S0_RXD	I2S Data Input
X_I2S0_SCLK	I2S Bit Clock
X_I2S0_TXD	I2S Data Output
X_I2S1_FS	I2S Frame/Sync
X_I2S1_RXD	I2S Data Input
X_I2S1_SCLK	I2S Bit Clock
X_I2S1_TXD	I2S Data Output
UART	
X_UART0_TX	UART Data Transmit
X_UART0_RX	UART Data Receive
X_UART1_TX	UART Data Transmit
X_UART1_RX	UART Data Receive
X_UART2_TX	UART Data Transmit
X_UART2_RX	UART Data Receive
X_UART2_NRTS	UART Data Request to Send
X_UART2_NCTS	UART Data Clear to Send
X_UART3_TX	UART Data Transmit
X_UART3_RX	UART Data Receive
X_UART3_NRTS	UART Data Request to Send
X_UART3_NCTS	UART Data Clear to Send
X_UART4_TX	UART Data Transmit
X_UART4_RX	UART Data Receive
IrDA	
X_UART5_TX	UART Data Transmit
X_UART5_RX	UART Data Receive
X_UART5_RX_h	Infrared Receive For FIR
X_UART6_TX	UART Data Transmit
X_UART6_RX	UART Data Receive
X_UART7_TX	UART Data Transmit
X_UART7_RX	UART Data Receive
X_UART8_TX	UART Data Transmit
X_UART8_RX	UART Data Receive
X_UART8_RX_h	Infrared Receive For FIR
X_UART9_TX	UART Data Transmit
X_UART9_RX	UART Data Receive
RGMII	

Vangotech

Pad Name	Description
X_RGMII_RX_CK	RGMII receive clock
X_RGMII_RXD[3:0]	Receive data
X_RGMII_RXCTL	Control signal for transferring other RX signals from RGMII
X_RGMII_GTX_CK	RGMII transmit clock
X_RGMII_TXCTL	Control signal for transferring other TX signals from RGMII
X_RGMII_TXD[3:0]	RGMII Transmit data
X_RGMII_MDC	Clock of PHY management
X_RGMII_MDIO	Data input/output of PHY management
RMII	
X_RMII_TXD[1:0]	RMII Transmit data
X_RMII_TX_EN	Transmit enable
X_RMII_RX_CRSDV	Receive carrier sense/data valid
X_RMII_RX_ER	Receive error
X_RMII_RXD[1:0]	Receive data
X_RMII_REF_CLK	RMII 50-MHz REF_CLK
X_RMII_MDC	Clock of PHY management
X_RMII_MDIO	Data input/output of PHY management
MII	
X_MII_TXD[3:0]	MII Transmit data
X_MII_TX_EN	MII Transmit enable
X_MII_RX_DV	Receive data valid
X_MII_RX_ER	Receive error
X_MII_RXD[3:0]	MII Receive data
X_MII_MDC	Clock of PHY management
X_MII_MDIO	Data input/output of PHY management
X_MII_RX_CK	MII receive clock
X_MII_TX_CK	MII transmit clock
X_MII_TX_ER	Transmit error
X_MII_CRS	Carrier sense
X_MII_COL	Collision detect
EMMC1	
X_EMMC1_CLK	EMMC1 Clock Signal
X_EMMC1_CMD	EMMC1 Command Line
X_EMMC1_RSTN	EMMC1 Reset Signal
X_EMMC1_DATA[7:0]	EMMC1 Data
LCDC	
X_LC_PCLK	Pixel clock of panel
X_LC_VS	Vertical synchronization signal of panel
X_LC_HS	Horizontal synchronization signal of panel
X_LC_DE	Data enable signal of panel

Vangotech

Pad Name	Description
X_LC_DATA[23:0]	Pixel data of panel DATA
MISC	
X_I2S_MCLK	I2S Clock Signal
X_SPI_DCX1	SPI Data Command Flag
X_SPI_DCX2	SPI Data Command Flag
DDR	
X_DDR_RASn	Row Address Strobe
X_DDR_ADDR[15:0]	DDR Address
X_DDR_ODT	DDR On-Die Termination
X_DDR_BA[2:0]	DDR Bank Select Signal
X_DDR_CSn[1:0]	DDR Chip Select Signal
X_DDR_RESETN	DDR Reset Signal
X_DDR_CKE[1:0]	DDR Clock Select
X_DDR_CK	DDR Clock Output Signal
X_DDR_CKB	DDR Clock Output Signal
X_DDR_WEn	DDR Writer Enable
X_DDR_CASn	DDR Column Address Strobe
X_DDR_DM_0	DDR Data Mask 0
X_DDR_DQSB_0	DDR DQS 0 # Used In the Different DQS Mode
X_DDR_DQS_0	DDR DQS 0
X_DDR_DQ_0[7:0]	DDR DQ Data Bus 0
X_DDR_VREF_D0	DDR Reference Voltage For Receivers 0
X_DDR_DM_1	DDR Data Mask 1
X_DDR_DQSB_1	DDR DQS 1 # Used In the Different DQS Mode
X_DDR_DQS_1	DDR DQS 1
X_DDR_DQ_1[7:0]	DDR DQ Data Bus 1
X_DDR_VREF_D1	DDR Reference Voltage For Receivers 1
X_DDR_DM_2	DDR Data Mask 2
X_DDR_DQSB_2	DDR DQS 2 # Used In the Different DQS Mode
X_DDR_DQS_2	DDR DQS 2
X_DDR_DQ_2[7:0]	DDR DQ Data Bus 2
X_DDR_VREF_D2	DDR Reference Voltage For Receivers 2
X_DDR_DM_3	DDR Data Mask 3
X_DDR_DQSB_3	DDR DQS 3 # Used In the Different DQS Mode
X_DDR_DQS_3	DDR DQS 3
X_DDR_DQ_3[7:0]	DDR DQ Data Bus 3
X_DDR_VREF_D3	DDR Reference Voltage For Receivers 3
X_DDR_DM_4	DDR Data Mask 4
X_DDR_DQSB_4	DDR DQS 4 # Used In the Different DQS Mode
X_DDR_DQS_4	DDR DQS 4

Vangotech

Pad Name	Description
X_DDR_DQ_4[7:0]	DDR DQ Data Bus 4
X_DDR_VREF_D4	DDR Reference Voltage For Receivers 4
X_DDR_RDRVUP	Pad To The Precise External Resistor For Compensating The Pull-up Driver
X_DDR_RDRVDN	Pad To The Precise Internal Resistor For Compensating The Pull-down Driver
OTG	
X_OTG_0_DP	USB D+ Signal
X_OTG_0_DM	USB D- Signal
X_OTG_1_DP	USB D+ Signal
X_OTG_1_DM	USB D- Signal
X_OTG_2_DP	USB D+ Signal
X_OTG_2_DM	USB D- Signal
ADC	
X_ADC_XAIN0	Analog Input 0
X_ADC_XAIN1	Analog Input 1
X_ADC_XAIN2	Analog Input 2
X_ADC_XAIN3	Analog Input 3
X_ADC_XAIN4	Analog Input 4
X_ADC_XAIN5	Analog Input 5
X_ADC_XAIN6	Analog Input 6
X_ADC_XAIN7	Analog Input 7
X_ADC_XVRT	Analog Higher/Positive Reference Voltage For ADC
X_ADC_XVRB	Analog Lower/Negative Reference Voltage For ADC

Vangotech

Chapter 4. System

This part details the V8130L system construction from following aspects.

- Memory Mapping
- Clock
- Reset
- GPV
- DDR
- SCU
- WDT
- System Boot
- PWMTMR
- GIC
- DMAC for AXI
- DMAC for AHB
- ADC
- TDC
- eFUSE
- Port Controller
- Crypto Engine
- Secure design

Vangotech

4.1. Memory Mapping

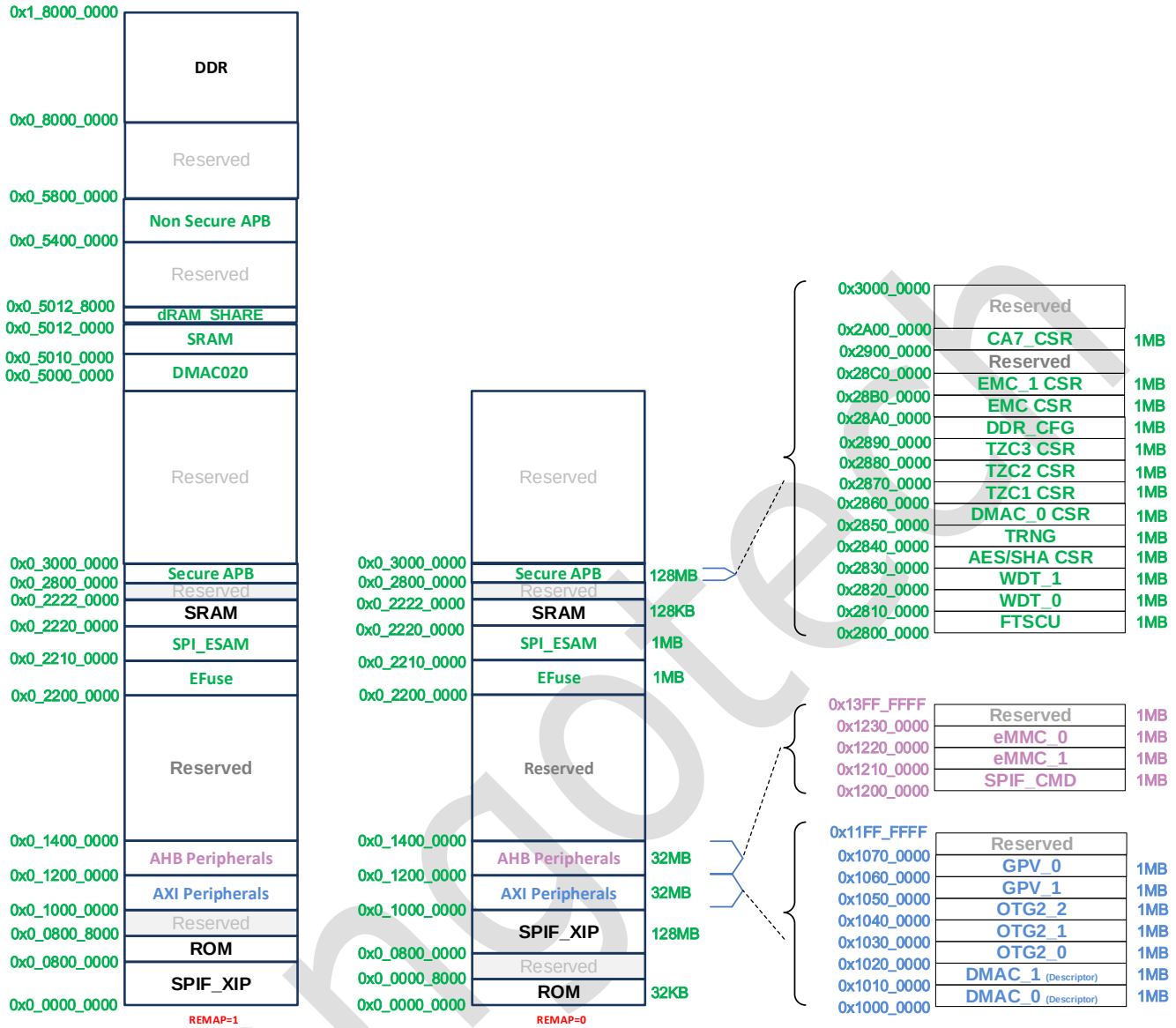


Figure 4-1. Memory Space Block Diagram of CA7

Vangotech

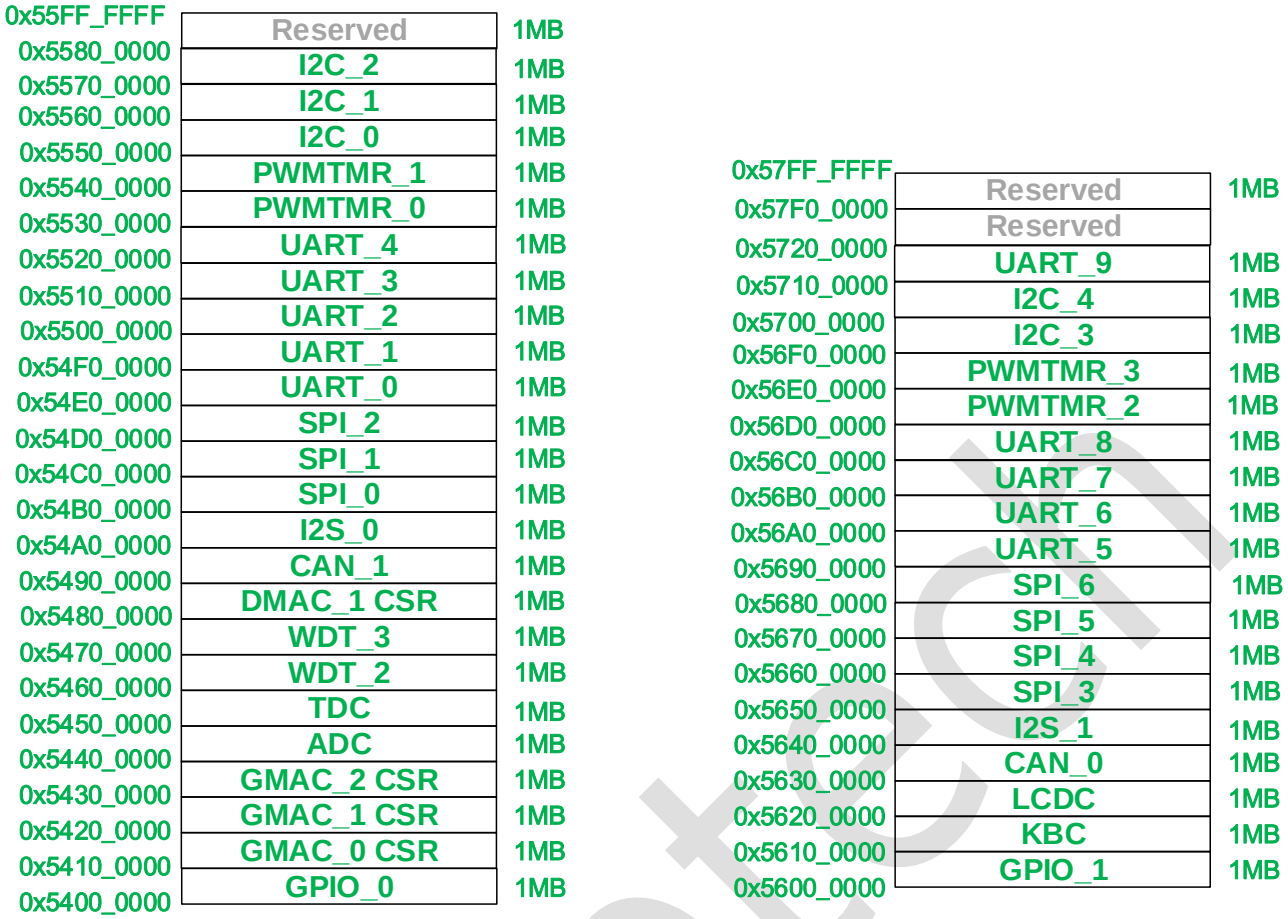


Figure 4-2. Memory Space Block Diagram of CA7 Non-Secure APB

Vangotech

4.2. Clock

4.2.1. System Clocking

The clocking mechanism for V8130L is divided into multiple groups. These groups are summarized in Table 4-3 according to which PLL they are attached to.

4.2.2. External Clock Sources

V8130L will have the external clock sources connected to the device. These clock sources are shown in Table 4-1.

Table 4-1. External Clock Sources (XTAL)

Clock Frequency	Clock Name	Details
25 MHz	T_osc_in	Used to drive PLL from which the CPU and bus clocks are derived
32.768 KHz	T_rtclk_in	Used to RTC timer clock (Calendar)

4.2.3. Internal Ring OSC

Table 4-2. Internal Clock Sources

Clock Frequency	Clock Name	Details
120 MHz ± 6.5%	OSC_HI	Used to provide system fast-boot application
32.768 KHz ± 4%	OSC_LO	Used to SCU and WDT timer clock source

4.2.4. PLL Programming Values

Table 4-3 defines the PLLs used in this device as well as the default output frequencies and corresponding default programming values.

Table 4-3. PLL Default Frequencies and Programming Values

PLL Instance	Default Output Clock (MHz)	Programming NS[5:0]	Programming MS[3:0]	Details
PLL0	1000 MHz	32	1	The output of this PLL is divided by programmer divider to 800 MHz and drives the Cortex-A7 CPU and the AMBA bus clock for high performance purpose that keeps the ratio between CPU and bus is 1:2 mode. AXI bus Clk (AClk) = 400 MHz Note: 1. The programming values of NS and MS are controlled by SCU. 2. When fast boot, the clock and bus of

				Cotex-A7 come from the internal ring O.S.C of 120 MHz.
PLL1	1000 MHz	40	1	For GMAC controller and partial peripheral clock generation Note: 1. PLL1 output frequency= PLL1 input frequency * NS / MS.
PLL2	600 MHz	24	1	For peripheral bus clock generation Note: 1. PLL2 output frequency= PLL2 input frequency * NS / MS.
PLL3	159.75 MHz	213	2	For audio clock generation Note: 1. PLL3 output frequency= PLL3 input frequency * NS / MS.
PLL4	400 MHz	32	-	For DDR controller clock generation Note: 1. PLL4 output frequency= PLL4 input frequency * NS.
PLL5	297 MHz	198	2	For LCDC display clock Note: 1. PLL5 output frequency= PLL5 input frequency * NS / MS.
PLL6	600 MHz	24	1	For sub-system clock generation Note: 1. PLL6 output frequency= PLL6 input frequency * NS / MS.
PLL7	400 MHz	32	1	For eMMC clock generation Note: 1. PLL7 output frequency= PLL7 input frequency * NS / MS.

Notes:

1. PLL0 and PLL4 are controlled by SCU unit, and configured by base address 0x2800_0000.
2. PLL0/PLL4 enable is controlled by the same bit (0x2800_0030[0]).
3. PLL0, PLL1, PLL2, PLL4 and PLL6 are enabled and disabled by software on booting up. The clock switching is designed for glitch free, and the signal needs to be synchronized by the source clocks. Users need to keep clock exist when switching is on-going, and close the clock after switching is completed.

4.2.5. Summary Clock information

This section summarizes the clock names, frequencies, clock periods, and synthesis recommendations for V8130L. Further descriptions of each clock domain can be found in the following sections.

Vangotech

Table 4-4. Clock Summary

Block	Clock Name	Clock Freq.(MHz)	Clock Derived from:	Description/Note
CORTEXA7	ca7_clk	800	PLL0	Cortex-A7 CPU clock frequency
CORTEXA7	ca7_aclk	400	PLL0	Cortex-A7 bus clock frequency
DDR	ddr_ch0_aclk ddr_ch1_aclk ddr_ch2_aclk ddr_ch3_aclk	400 300 300 200	PLL0 PLL2 PLL2 PLL1	DDR bus clock
DDR	ddr_pclk	100	PLL1	APB clock
DDR	ddr_ctlclk	400	PLL4	FTDDR3030 controller mclk
EMC	aclk	300	PLL2	AHB2ROM AHB bus interface clock
EMC	aclk	300	PLL2	AHB2ERAM AHB bus interface clock
MAC (u0 ~ u2)	aclk	200	PLL1	AXI bus interface clock of GMAC controller
MAC_0	Clk125_i	125	PLL1	125 MHz reference clock
MAC_0	Clk25_i	25	PLL1	25 MHz reference clock
MAC_0	Clk2pt5_i	2.5	PLL1	2.5 MHz reference clock
MAC_0	PTPClk	100	PLL1	PTP clock
MAC_1	RMII_REF_CLK	50	External	External reference clock
MAC_1	PTPClk	100	PLL1	PTP clock
MAC_2	Clk25_i	25	PLL1	25 MHz reference clock
MAC_2	Clk2pt5_i	2.5	PLL1	2.5 MHz reference clock
MAC_2	PTPClk	100	PLL1	PTP clock
OTG(0 ~ 2)	sclk	200	PLL1	FOTG210 controller AXI bus clock
OTG(0 ~ 2)	coreclkin	12	PLL2	FOTG210 controller core clock input
SPI2AHB	sck	25	External	SPI clock
SPI2AHB	sys_clk	200	PLL1	AHB bus clock
SDC(0 ~ 1)	sys_hclk	200	PLL1	AHB bus clock
SDC(0 ~ 1)	sdclk1x	200	PLL7	Used for memory control clock
SDC(0 ~ 1)	sdclk2x	400	PLL7	Twice as the sdclk1x clock
AES/SHA	BusClk	200	PLL1	All clocks of BP450 IP are the same with bus clock
SPIF	sys_clk	200	PLL1	AHB bus clock

Vangotech

SPIF	spi_clk	100	PLL1	SPI clock
DMAC030 (0 ~ 1)	aclk	300	PLL2	AXI bus interface clock of DMA controller
SPI(0 ~ 7)	PCLK	100	PLL6	SSP/SPI low-speed APB bus interface clock
SPI(0 ~ 7)	SSPCLK	100	PLL1	SPI serial clock input This clock drives the serial clock generator.
SPI(0 ~ 7)	sclk_out_r	-	SSPCLK	The output serial clock (sclk_out_r) is generated according to the following equation: Fsclk_out_r = Fsspclk / 2 * (SCLKDIV + 1)
PWMTMR(0 ~ 3)	ext_clk0	15	PLL2	Additional external clock source that can be chosen as the source clock of the timer1 counter
PWMTMR(0 ~ 3)	ext_clk1	15	PLL2	Additional external clock source that can be chosen as the source clock of the timer2 counter
PWMTMR(0 ~ 3)	ext_clk2	15	PLL2	Additional external clock source that can be chosen as the source clock of the timer3 counter
PWMTMR(0 ~ 3)	ext_clk3	15	PLL2	Additional external clock source that can be chosen as the source clock of the timer4 counter
PWMTMR(0 ~ 3)	PCLK	100	PLL6	APB clock
SCU	pclk	100	PLL1	APB bus interface clock of system control unit
SCU	scu_clk	32.768 KHz	Internal oscillator	Clock output of system control unit
SCU	rtc_clk	32.768 KHz	Internal oscillator	RTC clock output of system control unit
IIC(0 ~ 4)	PCLK	100	PLL6	APB bus interface clock of I2C interface controller
UART(0 ~ 9)	PCLK	100	PLL6	APB bus interface clock of UART
UART(0 ~ 9)	io_urda_uclk	120	PLL2	UART controller clock
GPIO(0 ~ 1)	PCLK	100	PLL6	APB bus interface clock of GPIO controller

Vangotech

WDT(0 ~ 3)	PCLK	100	PLL6	APB bus interface clock of WDT
WDT(0 ~ 3)	EXTCLK	32.768 KHz	Internal oscillator	Additional external clock source that can be chosen as the source clock
EFUSE	PCLK	100	PLL1	APB bus interface clock of eFuse controller
EFUSE	eFuse_clk	25	XTAL	System clock of eFuse controller
TRNG	PCLK	100	PLL1	APB bus clock of TRNG configuration
CAN(0 ~ 1)	sys_clk	100	PLL6	CAN bus controller system clock
CAN(0 ~ 1)	pclk	100	PLL6	APB bus clock for configuration feature
KBC	pclk	100	PLL6	APB bus clock for configuration feature
LCDC	Aclk	200	PLL1	AXI bus clock
LCDC	LC_CLK	148.5	PLL5	LCD controller clock
LCDC	LC_SCALER_ CLK	148.5	PLL5	Clock for scaler
LCDC	TV_CLK	148.5	PLL5	Clock for the TV domain
DMAC	hclk	150	PLL6	AHB bus clock

4.2.6. Top Level Clocking Diagram

The crystal and PLL are integrated in SoC_top. SoC_top provides the stable 25 MHz clock from crystal and 120 MHz internal oscillator for CPU fast boot purpose. The detailed clock generation diagram are listed in Figure 4-3~Figure 4-8

Figure 4-8.

Note:

There are two clock sources for V8130L clock system, one is internal 120 MHz OSC and the other is PLL source. The clock diagram lists all generated clock frequency on leaf marked by “red” and “green”.

“Red”: From PLL clock source

“Green”: From internal 120 MHz OSC

Vangotech

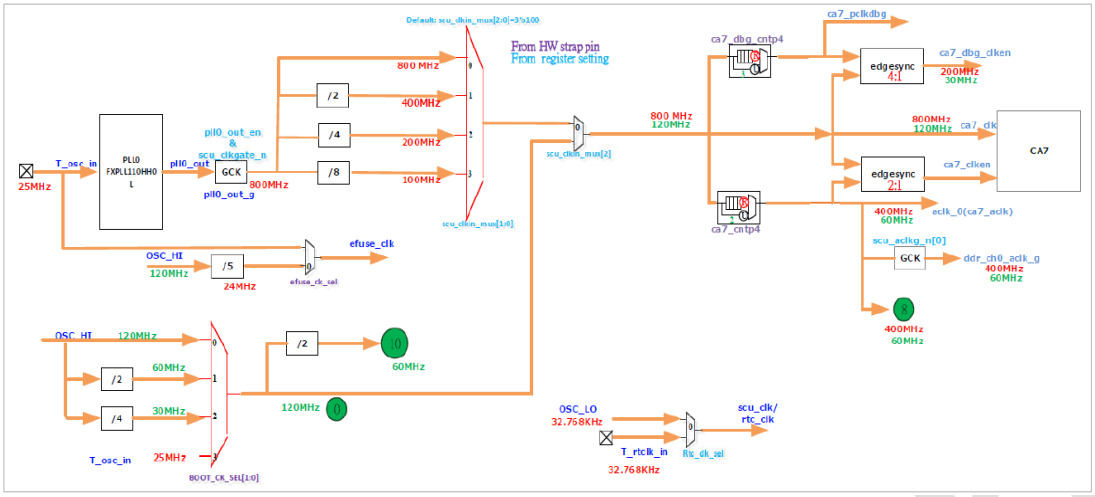
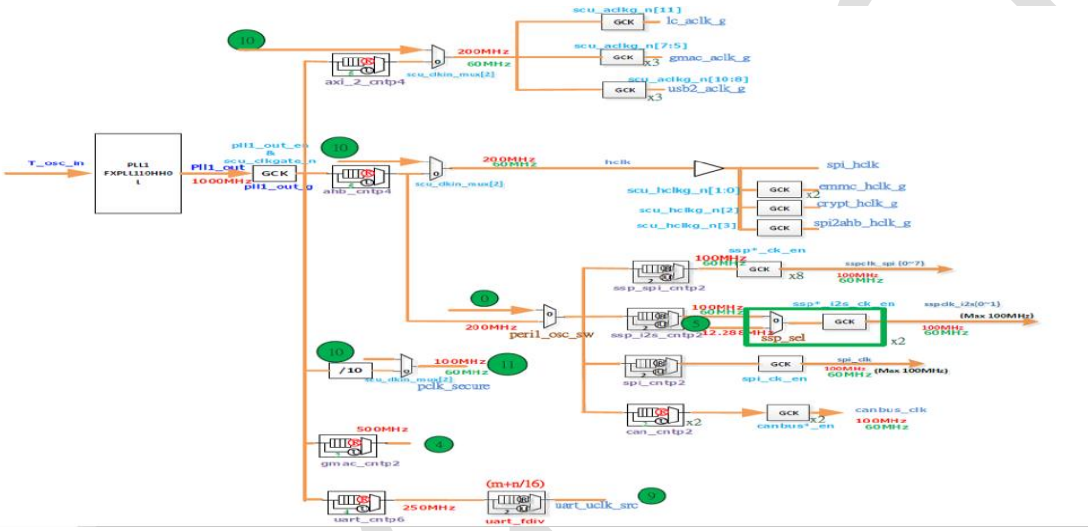


Figure 4-3. Clock Distribution Diagram (For CA7)



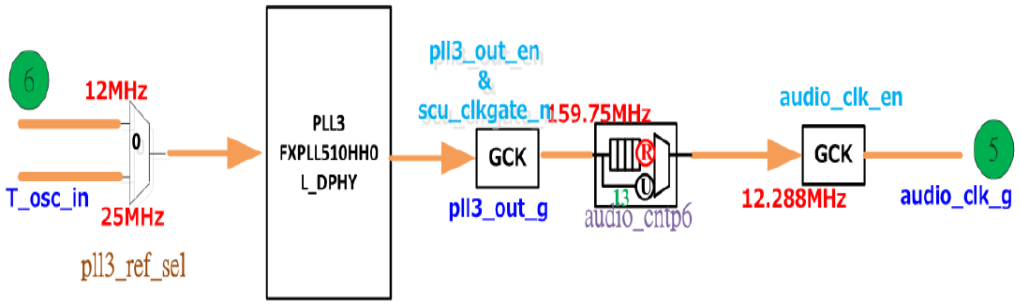


Figure 4-4. Clock Distribution Diagram (For Peripheral Part 1)

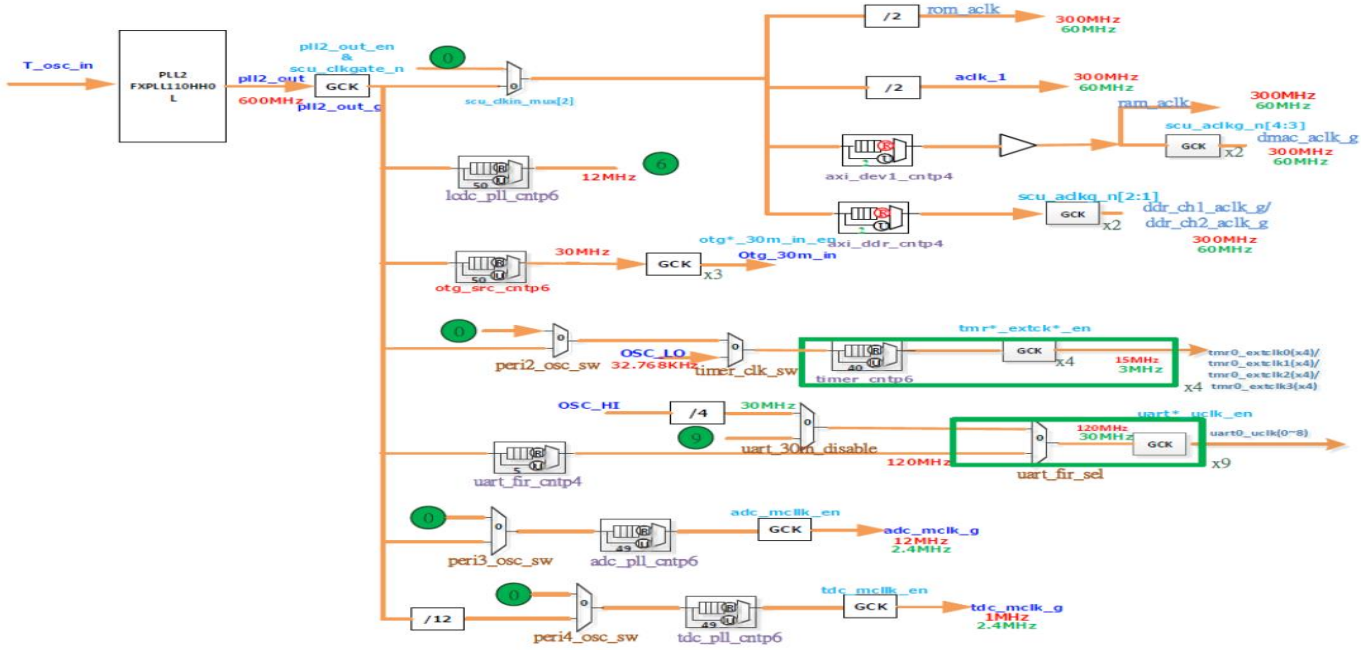


Figure 4-5. Clock Distribution Diagram (For Peripheral Part 2)

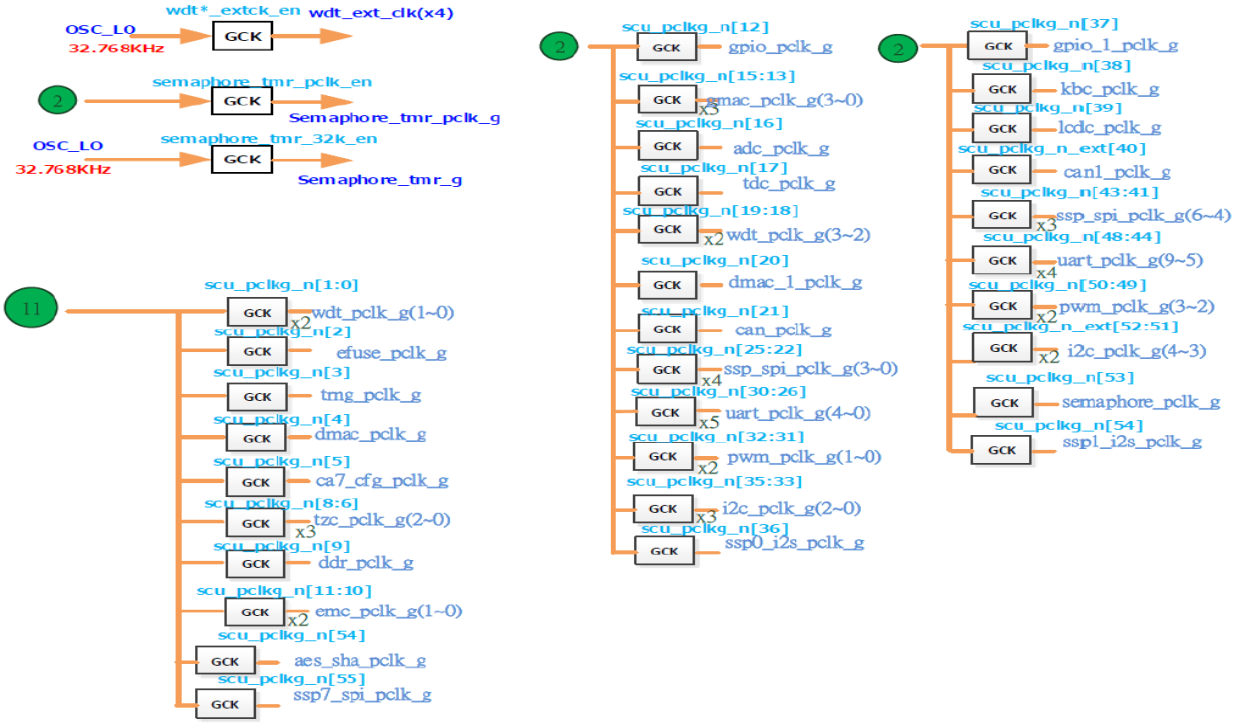


Figure 4-6. Clock Distribution Diagram (For Peripheral Part 3)

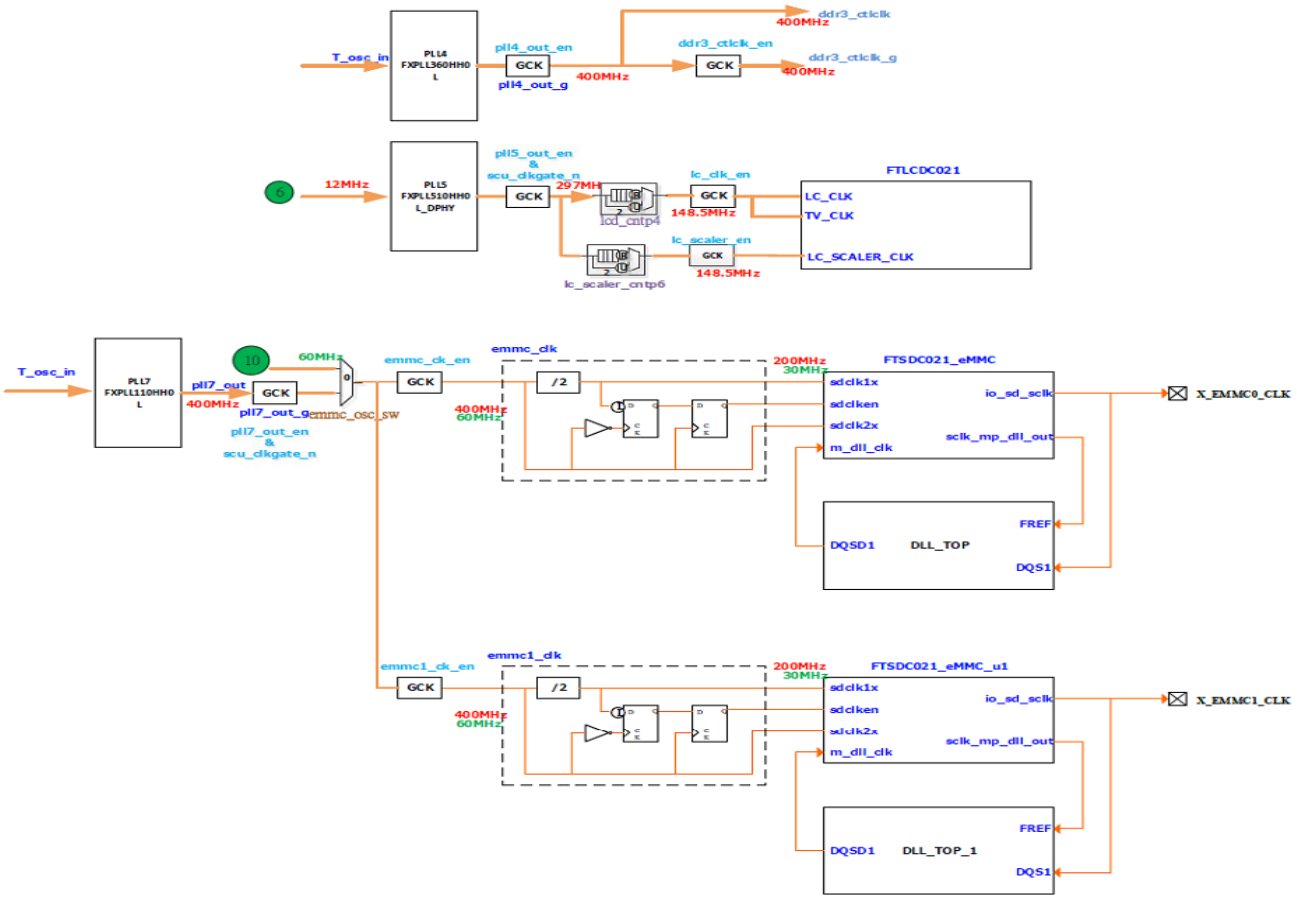


Figure 4-7. Clock Distribution Diagram (eMMC/LCDC Controller Clock Generation)

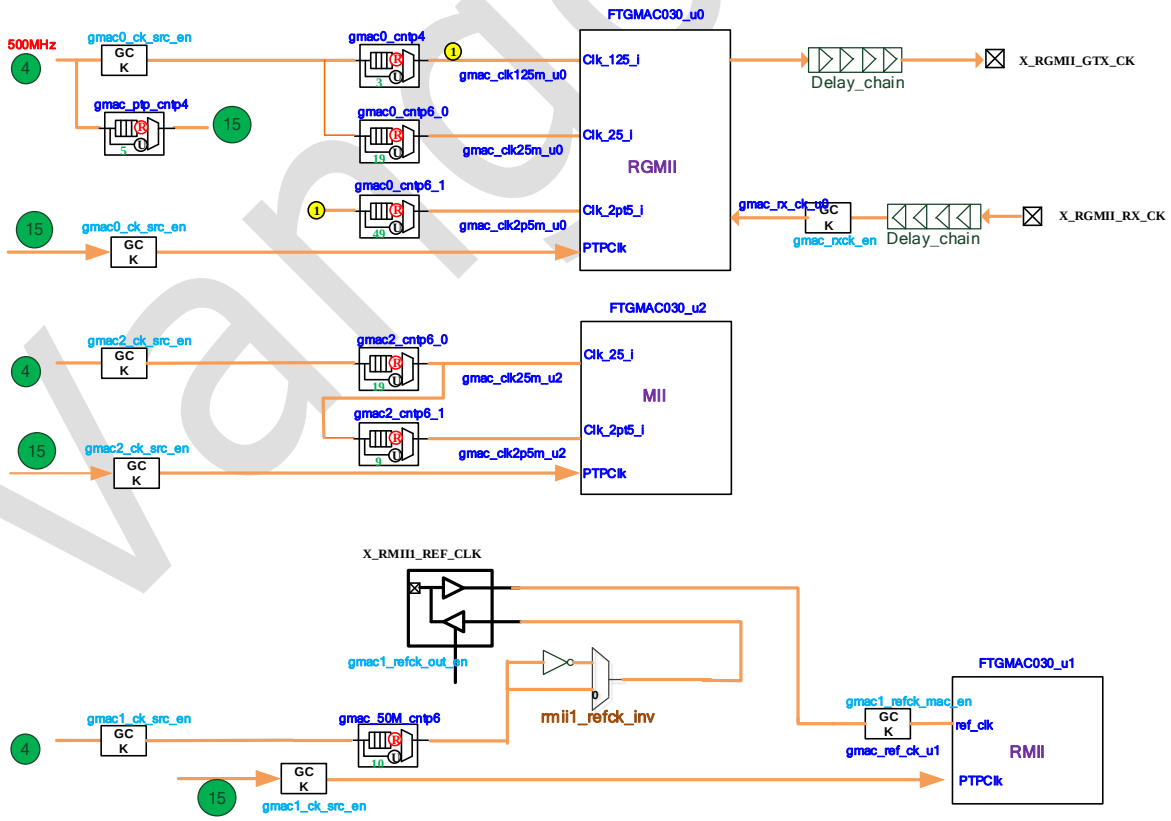


Figure 4-8. Clock Distribution Diagram (MAC Controller Clock)

4.2.7. Special Clock Specification

For customer system application and requirement, there are some specifications for peripheral device and

clock generation configuration in this chapter.

4.2.7.1. UART Baud Rate Specification and Configuration of UART

For V8130L system requirement, the UART baud rate can support 1200bps, and up to 3686400bps by the configuration extension register. In some applications, the ideal baud rate is not equivalent to the practical baud rate.

Below is the baud rate configuration list and deviation for reference.

Target BR (bps)	CLK Source (MHz)	m of FDIV	n of FDIV	clko** (io_irda_uclk)	PSR	DL	Pratical BR*** (bps)	Deviation
3686400	250	4	4	58.82352941	1	1	3676470.588	-0.269
1843200	250	4	4	58.82352941	1	2	1838235.294	-0.269
921600	250	4	4	58.82352941	1	4	919117.6471	-0.269
460800	250	4	4	58.82352941	1	8	459558.8235	-0.269
230400	250	4	4	58.82352941	1	16	229779.4118	-0.269
115200	250	4	4	58.82352941	1	32	114889.7059	-0.269352533
57600	250	4	4	58.82352941	1	64	57444.85294	-0.269352533
38400	250	3	3	78.43137255	1	128	38296.56863	-0.269352533
19200	250	3	3	78.43137255	1	256	19148.28431	-0.269352533
9600	250	3	3	78.43137255	1	512	9574.142157	-0.269352533
4800	250	3	3	78.43137255	1	1024	4787.071078	-0.269352533
2400	250	3	3	78.43137255	1	2048	2393.535539	-0.269352533
1200	250	3	3	78.43137255	1	4096	1196.76777	-0.269352533

Notes:

1. m/n is the fraction divider setting in the extension register.
2. PSR/DL are the UART configuration registers by the UART specification.
3. By the combination of these register settings, it can meet the system baud rate requirement.

4.2.7.2. IrDA Data Rate Specification and Clock Configuration of UART

For data transformation by IrDA protocol, it supports SIR mode and FIR mode. The data rate can be 9.6 KB/s and 115.2 KB/s in SIR mode and 4 MB/s in FIR mode. The clock source of UART mode and IrDA mode can be switched by the mux of uart_fir and PSR and DL as listed in the below table.

	Target DataRate	clko** (io_irda_uclk)	PSR	DL	Pratical BR*** (bps)	Deviation
SIR mode	115200	120	2	33	113636.3636	-1.357323232
SIR mode	9600	120	24	33	9469.69697	-1.357323232
FIR mode	4000000	120	15	1		

Note: For FIR mode, PB16XCLK must be dedicated to 8 MHz!

4.2.8. V8130L Core and IO Voltages

V8130L has a core operating voltage of 1.1 V. The V8130L I/O is designed for the 3.3-V operation.

4.2.9. Clock Gating for Dynamic Power Reduction

To reduce the dynamic power consumption, clock gating is implemented by using the clock control registers within the System Control unit. Synthesis level clock gating will also be implemented in V8130L.

4.2.9.1. Power Rail Scheme

Table 4-5. Power Rail Scheme

Power Rail Parameter	Minimum Voltage	Nominal Voltage	Maximum Allowable Voltage
Core voltage	1.045 V	1.1 V	1.155 V
I/O voltage	3.135 V	3.3 V	3.465 V

4.2.9.2. Power Domains

There is always-on domain (AON) for RTC and others (Default domain) can be power-off for application. The AON and Default domain are all 1.1 V power source.

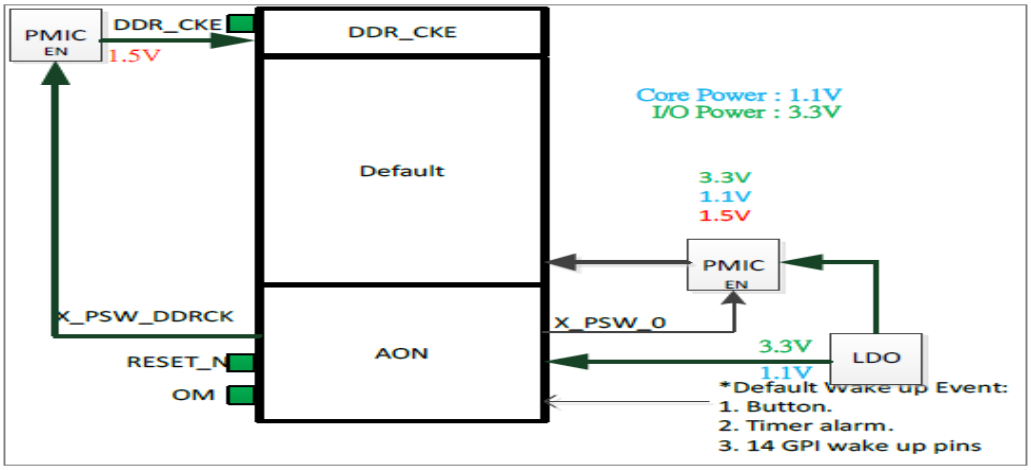


Figure 4-9. Power Domain Diagram of V8130L

V8130L has a single internal power domain, the core power domain (Core power is 1.1 V).

4.3. Reset

4.3.1. Power-On Reset (Cold Reset) and Power Enable Operation

The power-on reset is a cold reset that can be asserted by the POR hardcore in SoC. The POR analog IP is usually integrated to the same power domain with the Real Time Clock Unit. When the RTC power is applied, the power-on reset will be asserted to initialize all functions of the Real Time Clock Unit and clocked by build-in RC OSC. The system will start to execute the power enable sequence. It will drive the scu_pwren port to enable the power of the power control unit. When the power control unit is ready, it will issue a scu_por_n to the reset related logical and IPs that are integrated into the PCU power domain. A hardware reset input port is in the power control unit, which is connected to a RC circuit with a reset button in PCB to automatically trigger a reset event and then, release it after the interval of the RC charging time when power is attached. In the power enable sequence, PCU will enable the power of the clock control unit and start the boot-up sequence after inactivating the reset.

The clock control unit is implemented in the core power domain, and the most blocks and peripherals are integrated into this power domain. Consequently, it is possible to cause the in-rush current when the power is turned on. SCU provides 2time-slot power start-up sequences in the power control unit, and they can be programed. After finishing the power-up sequence, the core power will be checked stable from the ccu_vccok signal, and the isolation control signal, scu_isopwr_n, will be de-asserted to build up the communication between the backup, power control, and core power domains. To stabilize the internal clocks, PLL and DLL must be stable and are set to a specified frequency before the clock gating signals are de-asserted. When the boot-up sequence is complete, all clocks are unblocked and CPU starts to fetch the program code.

The reset time is shown in Figure 4-10 and Figure 4-11

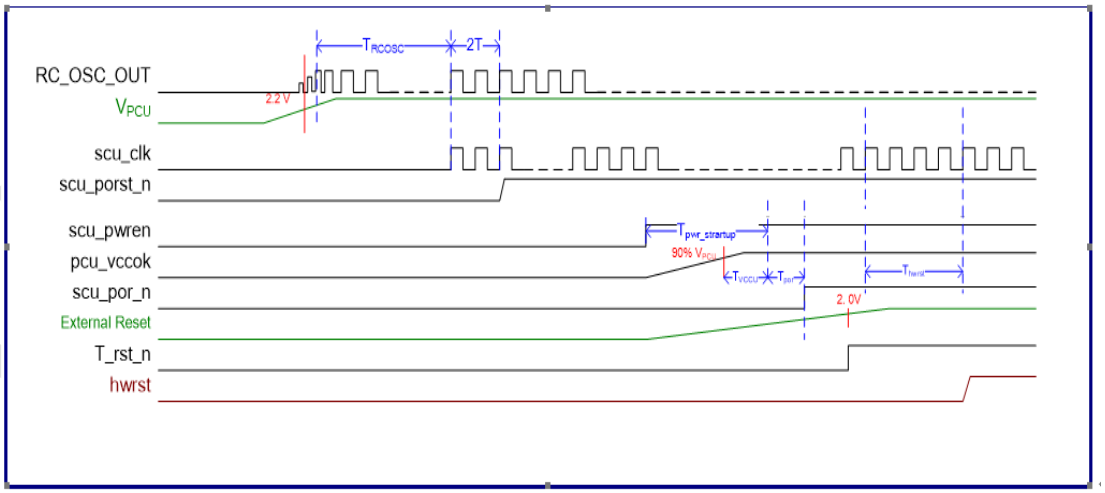


Figure 4-10. Power-on Reset Timing when scu_clk is from Built-in RC Ring Oscillator

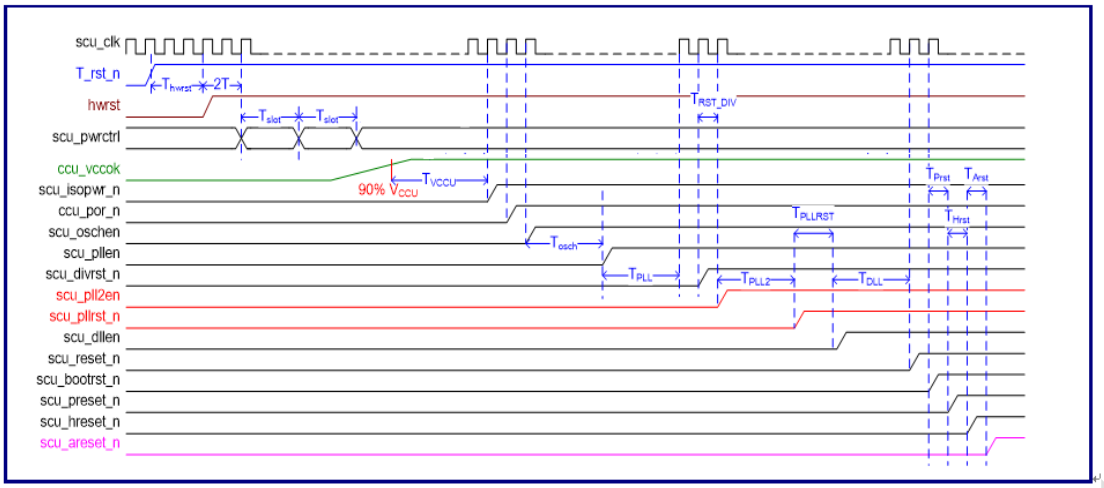


Figure 4-11. Power-on Reset Timing

About reset mechanism still included "Warm reset" and "Watch dog reset", and some system operation by SCU list as below.

1. Frequency Change Sequence (FCS)
2. PLL update sequence
3. Sleeping sequence
4. Soft-off sequence

For V8130L reset schematic, all controlled by SCU and configured by customer's requirement. V8130L with 2 power control pin to control PMIC. One is default domain and the other is DDR_CKE power switch. The detail reset sequence can be refer by Fig 6-3 Reset control diagram. The reset sequence can be referenced by Fig 6-6 Reset control diagram. All reset released by SCU except otg_rst_n, Dphy_resetn (0x2800_80A4) these three reset signals need released by software.

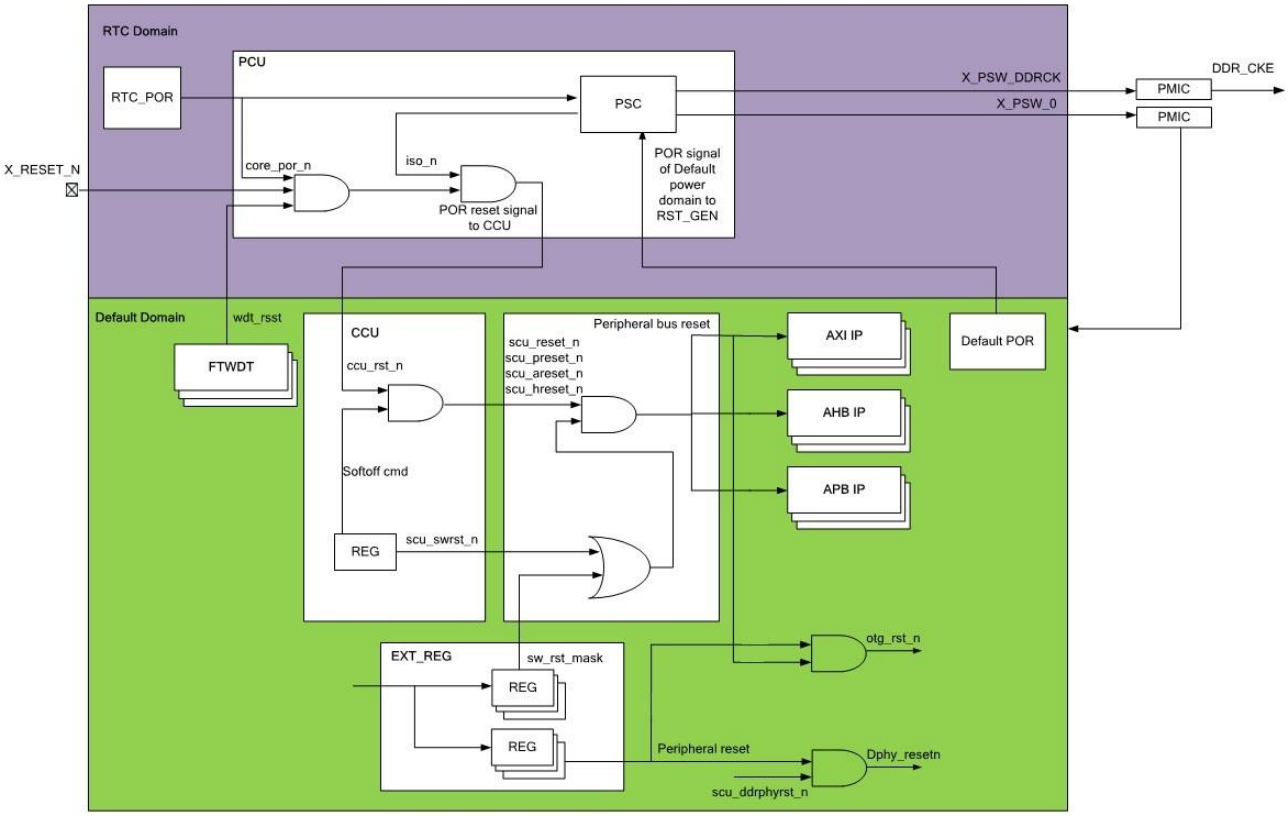


Figure 4-12. Reset control diagram

4.4. Global Programming View (GPV)

The GPV implemented on NIC bus 0/1 that are configuration register region for bus feature programed by users. Partialslaves of NIC 0/1 are "Boot Secure" attribute that means secure access when reset is released. Users can modify the attribution by programming GPV on the corresponding offset by slave "Node".

Table 4-6. NIC Bus 1 Security Attribution Register (Base Address = 0x1050_0000)

Offset	Default Value	Description
0x0000_000C	0x1	Slave S0: DDR CH1
0x0000_0010	0x1	Slave S1: DDR CH2
0x0000_0014	0x0	Slave S2: ROM
0x0000_0018	0x0	Slave S3: SRAM
0x0000_001C	0x0	Slave S4: DMAC_0
0x0000_0020	0x0	Slave S5: DMAC_1
0x0000_0024	0x0	Slave S6: OTG_0
0x0000_0028	0x0	Slave S7: OTG_1
0x0000_004C	0x0	Slave S8: ESAM
0x0000_002C	0x0	Slave S9: SPI_XIP
0x0000_0030	0x0	Slave S10: XPI_CMD
0x0000_0034	0x0	Slave S11: SDC_0
0x0000_0038	0x0	Slave S12: SDC_1
0x0000_0048	0x0	Slave S13: X2P_0
0x0000_0008	0x0	Slave S14: H2P_1
0x0000_003C	0x0	Slave S15: H2P_2
0x0000_0040	0x0	Slave S16: EFUSE
0x0000_0044	0x0	Slave S17: OTG_2

Table 4-7. NIC Bus 0 Security Attribution Register (Base Address = 0x1060_0000)

Offset	Default Value	Description
0x0000_0008	0x1	DDR_CH0(S0) Security setting This register consists of 1 bit for the non-virtual slaves, and up to 16 bits for the virtual or APB master interfaces. Users can configure the register bits as follows: 0: Secure 1: Non-secure
0x0000_000C	0x0	NIC_0_to_1(S1) security setting

Table 4-8. NIC Bus 1 GPV Offset (Base Address = 0x1050_0000)

Vangotech

Offset	Default Value	Description
Slave Device	-	-
0x0000_3000	-	Slave S0: DDR CH1
0x0000_4000	-	Slave S1: DDR CH2
0x0000_5000	-	Slave S2: ROM
0x0000_6000	-	Slave S3: SRAM
0x0000_7000	-	Slave S4: DMAC_0
0x0000_8000	-	Slave S5: DMAC_1
0x0000_9000	-	Slave S6: OTG_0
0x0000_A000	-	Slave S7: OTG_1
0x0001_3000	-	Slave S8: ESAM
0x0000_B000	-	Slave S9: SPI_XIP
0x0000_C000	-	Slave S10: SPI_CMD
0x0000_D000	-	Slave S11: SDC_0
0x0000_E000	-	Slave S12: SDC_1
0x0001_2000	-	Slave S13: X2P_0
0x0000_2000	-	Slave S14: H2P_1
0x0000_F000	-	Slave S15: H2P_2
0x0001_0000	-	Slave S16: H2P_2
0x0001_1000	-	Slave S17: OTG_2
Master Device	-	-
0x0004_2000	-	Master M0: DMA_0
0x0004_4000	-	Master M1: DMA_1
0x0004_3000	-	Master M2: OTG_0
0x0004_F000	-	Master M3: OTG_1
0x0004_5000		Master M4: OTG_2
0x0004_6000	-	Master M5: GMAC_0
0x0004_7000	-	Master M6: GMAC_1
0x0004_8000	-	Master M7: GMAC_2
0x0004_C000	-	Master M8: NIC_0_to_1
0x0004_A000	-	Master M9: SDC_0
0x0004_B000	-	Master M10: SDC_1
0x0004_D000	-	Master M11: AES/SHA
0x0004_9000	-	Master M12: SPI2AHB

Table 4-9. NIC Bus 0 GPV Offset (Base Address = 0x1060_0000)

Offset	Default Value	Description
Slave Device	-	-
0x0000_2000	-	Slave S0: DDR CH0
0x0000_3000	-	Slave S1: NIC_0_to_1

vangotech

Master Device	-	-
0x0004_2000	-	Master M0: CA7

Vangotech

4.5. DDR

4.5.1. Overview

The DDRx SDRAM memory controller is an AHB/AXI compliant DDRx SDRAM controller that supports various DDR1/ DDR2/ DDR3/ DDR3L/ LPDDR1/ LPDDR2 SDRAMs. This controller uses a burst length of four for the DDR1/ LPDDR1/ DDR2 (1:1 mode) memories and a burst length of eight for the DDR3/DDR3L/LPDDR2/DDR2 (1:2 mode) memories to accelerate the read and write speeds. The memory controller prefetches the sequential read data for the burst read commands to enhance the data transfer rate.

If the clock ratio between MClk and the DRAM clock is 1:2, two commands will be required at each MClk cycle for DDRx PHY. For example, DFI_CS_N_P0 and DFI_CS_N_P1 will be used in DDRx PHY. FTDDR3030 uses the clock ratio of 1:2 in the DDR3/ DDR3L/ LPDDR2/ DDR2 (1:2 mode) applications.

If the clock ratio between MClk and the DRAM clock is 1:1, only DFI_CS_N_P0 will be used in DDRx PHY. FTDDR3030 uses the clock ratio of 1:1 in the DDR1/LPDDR1/DDR2 (1:1 mode) applications.

Note: FTDDR3030 treats DDR3L as the same as DDR3. In the following sections, DDR3 denotes DDR3 and DDR3L as well.

4.5.2. Features

• Compliant with AMBA AXI interface protocols

• Supports maximum of 8G bytes (DDR3 type), 2G bytes (DDR2/LPDDR2 type), and 1G bytes (DDR1/LPDDR1 type) for each rank

• Uses a burst length of four for DDR1/LPDDR1/DDR2 (1:1 mode) memories and a burst length of eight for DDR3/LPDDR2/DDR2 (1:2 mode) memories to speed up read and write operations

• Supports four AXI memory slaves (Slave 0 ~ Slave 3)

• Supports one DIMM or two RANKs (Chip selection) of on-board DRAM (Two ranks should have the same trace length to ASIC chip.)

• Enters DDRx SDRAM self-refresh mode by using hardware handshake signal or software configuration

• Prefetches sequential read data for AXI burst read commands

• Interacts with three DDRx PHY hard cores, CMDADDR block, Data block and compensation block

• Supports DFI2.1 control/write/read interfaces between controller and PHY

• Supports hardware configurable data FIFO depths of 8, 16, and 32

• Supports MA tables corresponding to different sizes and types

Vangotech

- Supports configurable 32-bit or 64-bit AXI bus, 128-bit AXI bus (Only supports 1:2 mode)
- Supports 33-bit AXI address bus
- Supports 8-bit, 16-bit, 32-bit, and 64-bit memory device widths
- Provides two arbitration strategies for channel arbitrations:

  Two-level round-robin arbitration with channel grant counts

  Read-write grouping arbitration with R/W commands grant counts
- Supports synchronous/asynchronous modes for AHB/AXI slave interface
- Automatically enters SDRAM power-down mode when controller idles exceed user-defined clock cycles
- Supports x8 and x16 DDR1/DDR2/DDR3 SDRAM chips
- Supports x16 and x32 LPDDR1/LPDDR2 SDRAM chips
- Supports exclusive accesses with burst length of 0 (SINGLE) for AXI slave interface
- Supports ECC RMW (Read-Modify-Write) operation, which can do one-bit error correction and two-bit error detection

4.5.3. Block Diagram

Figure 4-13 depicts the functional block diagram of the DDRx memory controller.

Vangotech

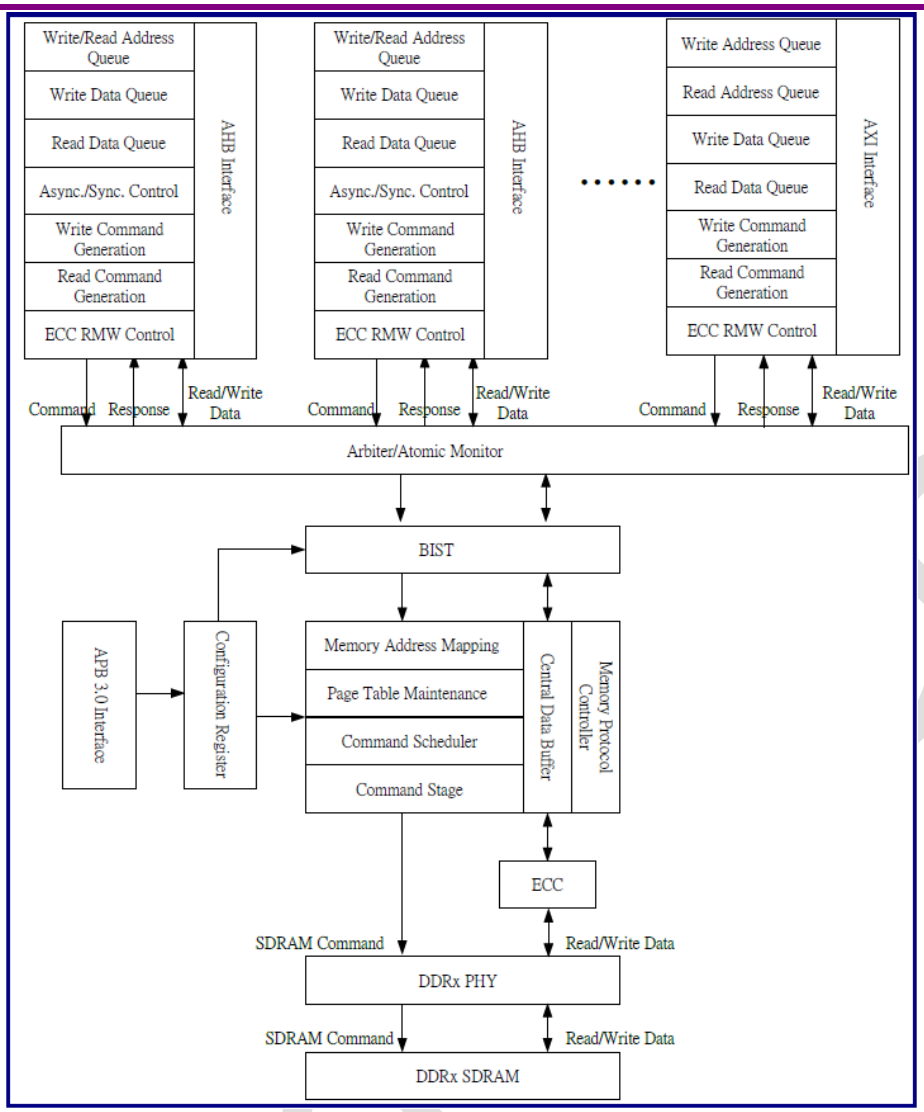


Figure 4-13 Functional Block Diagram

The DDRx memory controller separates a design into several blocks, including the AHB interface, AXI interface, arbiter, and memory protocol controller. These blocks are described below:

AHB Interface

The AHB interface comprises an AHB slave interface, which performs the following functions:

Asynchronous FIFO depth control. The supported FIFO depths include:

Command FIFO: It has a depth of 4 bits or 8 bits.

Read data FIFO: It has a depth of 8 bits or 16 bits (Hardware configurable).

Write data FIFO: It has a depth of 8 bits or 16 bits (Hardware configurable).

Read command generation to prefetch data for the AHB read bursts

A programmable register is provided to define the prefetch length for the AHB unspecified-length (INCR) read bursts.

AHB SPLIT response generation. The following two conditions can trigger the SPLIT responses:

Condition 1: FTDDR3030 receives an AHB read transaction and the desired data cannot be supplied at the next cycle (Data phase).

Condition 2: FTDDR3030 receives any valid AHB transaction and one of the command/read/write FIFO is full.

Provides the read-only mode and the write-only mode to reduce the gate count of the data FIFO

Provides the command flushing mechanism for handling the cross-channel data coherence problem

AXI Interface

The AXI interface comprises an AXI slave interface, which performs the following functions:

Write command generation

Read command generation

AXI write response, read response, and exclusive access response generations. AXI write response will be issued after the command going into the arbiter.

Merge write/read data for transfer efficiency

ECC RMW Block

There are ECC RMW (Read-Modify-Write) blocks inside each AHB/AXI channel.

RMW operation for non-full memory width write access

Additional write data queue for write data check

(AXI interface queue size = 4 x 8 x memory width, AHB interface = 2 x 8 x memory width)

Arbiter

The two-level round-robin arbiter is used to arbitrate the requests from the AHB/AXI command queue.

The commands from the highest priority channel will be issued to the memory protocol controller. An advanced arbitration mechanism of the read-write grouping is used to reduce the "Write-to-Read" wait cycles of DDRx SDRAMs.

Vangotech

Memory Protocol Controller

The memory protocol controller comprises the memory address mapping, page table maintenances, command reorder and scheduler, and inter-command DDRx timing parameter check:

Memory address can map to different DDRx SDRAM types and memory interface widths.

Command reorder can choose the proper DDR command to reduce the bank-conflict or R/W turnaround performance impact.

The controller adopts the open-page policy, which means that the DDR bank is activated until the bank-conflict (same bank with different row) condition is met, or other exceptional DDR commands such as auto-refresh are necessary. When the bank-conflict condition happens, the single bank is precharged and then activated with the same bank with different rows.

Command scheduler can check the inter-command timing parameters to meet the DDRx SDRAM specifications and perform a command scheduling, which can choose the proper commands to maximize the DDRx bus efficiency.

Command stage is the actual DDRx SDRAM command stage, which contains the pre_command register output of the command scheduler.

Elastic FIFO can separate the clock trees of MClk and POSTCTLCK domains to reduce the clock tree length of POSTCTLCLK clock domain.

ECC generation and error correction blocks can generate ECC code with write data and correct/ detect the read data with read ECC code.

Cooperate with DDRx PHY to perform write-leveling and read-leveling calibrations for DDR3 SDRAM.

FTDDR3030 accesses DDRx SDRAM per byte basis, and users can do bit swaps within the same byte for the PCB trace routing flexibilities (DDR2 and DDR3). The bit swaps within the same bytes are not allowed in LPDDR2 system, because the read data of MRR command is placed in DQ[7:0].

Vangotech

4.5.4. Connection between DDRx Controller and DDRx PHY

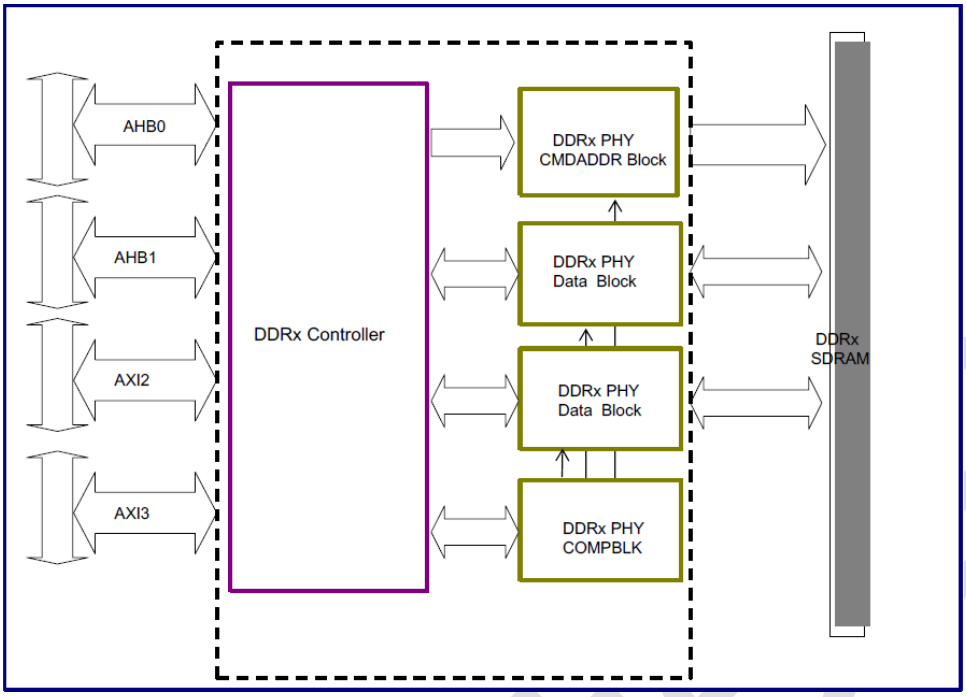


Figure 4-14 Logic Symbol between DDRx SDRAM Controller and DDRx PHY

Figure 4-14 illustrates the logic symbol between the DDRx SDRAM controller and DDRx PHY. There are two types of interface signals: AHB/AXI interface signals and DDRx PHY hard core signals. Please refer to Section 1.6 for the detailed signal definitions. Please note that the DDRx SDRAM controller only connects to DDR PHY; there is no direct connection between the DDRx SDRAM controller and DDRx SDRAM. As a result, the DDRx SDRAM controller must be connected to DDRx PHY. For the signals of DDRx PHY, please refer to the DDRx PHY specifications.

4.5.5. Register List

Base Offset	Name	Description
00	MCCR	Memory controller configuration
04	MCSR	Memory controller state control
0C	MRSVR1	Mode register set values for EMR2 and EMR3
28	COMPBLKCR	COMPBLK control
2C	AODCR	Automatic power-down/self-refresh control
30	CHARBRA	Channel arbitration setup register A
34	CHGNTRA	Channel arbiter grant count register A
38	CHGNTRB	Channel arbiter grant count register B
3C	PHYWRTMR	DDRx PHY write/read data access timing control
40	FLUSHCR	Command flush control
44	FLUSHSR	Command flush status
48	SPLITCR	AHB split control
4C	UPDCR	DDRx PHY and DDR3 update control

50	REVR	Revision register
54	FEATR1	Feature register
58	FEATR2	Feature register 2
5C	UDEFR	User-defined register
60	WLEVELCR	Write-leveling control register
64	WLEVELBHR	Byte 7 ~ byte 4 of the write-leveling control register
68	WLEVELBLR	Byte 3 ~ byte 0 of the write-leveling control register
6C	PHYMISCR1	DDR3 MISC register
70	RLEVELCR	Read-leveling control register
74	MSDLYCR	msdly byte control register
78	WRDLLCR	wdllsel control register
7C	TRAFMR	Traffic monitor clock cycle register
80	CMDCNTR0	Command count for channel 0
84	CMDCNTR1	Command count for channel 1
88	CMDCNTR2	Command count for channel 2
8C	CMDCNTR3	Command count for channel 3
90	CMDCNTR4	Command count for channel 4
94	CMDCNTR5	Command count for channel 5
98	CMDCNTR6	Command count for channel 6
9C	CMDCNTR7	Command count for channel 7
A0	AHBRPRER1	AHB INCR read prefetch length 1
A4	AHBRPRER2	AHB INCR read prefetch length 2
A8	INITWCR1	Initialization waiting cycle count 1
AC	INITWCR2	Initialization waiting cycle count 2
B0	QOSCR	QoS control register
B4	QOSCNTRA	QoS command count register A
B8	QOSCNTRB	QoS command count register B
BC	QOSCNTRC	QoS command count register C
C0	QOSCNTRD	QoS command count register D
C4	CHARBRB	Channel arbitration setup register B
C8	CHGNTRC	Channel arbiter grant count register C
CC	CHGNTRD	Channel arbiter grant count register D
D0	DBGADR	Debug address
D4	DBGADMR	Debug address mask
D8	DBGWDR	Debug write data
DC	DBGWDMR	Debug write data mask
E0	DBGMSTR	Debug master control
E4	DBGACCR	Debug access control
E8	DBGPCR	Debug policy control
EC	DBGENR	Debug control
F0	DBGADSR	Debug address status

Vangotech

F4	DBGWDSR	Debug write data status
F8	DBGMSTSR	Debug master status
FC	DBGACCSR	Debug access status
100	LP2MRA	LPDDR2 Mode Register MR1/MR2/MR3
104	LP2MRB	LPDDR2 Mode Register MR9/MR10
108	LP2MRC	LPDDR2 Mode Register MR16/MR17
10C	LP2MRD	Reserved
110	LP2MRE	LPDDR2 Mode Register MR63
114	LP2MRCR	LPDDR2 MRW and MRR Commands
118	LP2MRVR	LPDDR2 MRR Read Value
11C	LP2ADLR	LPDDR2 Added Latency
120	LP2WCR1	LPDDR2 1us waiting cycle
124	LP2WCR2	LPDDR2 10us waiting cycle
128	-	Reserved
12C	REARBDISR	Re-arbitration disable count register
130	PHYRDTFR	DLL delay tuning of the DDR PHY read path (Falling)
134	PHYMISCR2	DDR PHY MISC2 register
138	EFIFOCR	EFIFO control register
13C ~ 14C	-	Reserved and Opaque
150 ~ 15C	-	Reserved
160	RB0DSKW	Byte 0 read de-skew register
164	RB1DSKW	Byte 1 read de-skew register
168	RB2DSKW	Byte 2 read de-skew register
16C	RB3DSKW	Byte 3 read de-skew register
170	RB4DSKW	Byte 4 read de-skew register
174	RB5DSKW	Byte 5 read de-skew register
178	RB6DSKW	Byte 6 read de-skew register
17C	RB7DSKW	Byte 7 read de-skew register
180	WB0DSKW	Byte 0 write de-skew register
184	WB1DSKW	Byte 1 write de-skew register
188	WB2DSKW	Byte 2 write de-skew register
18C	WB3DSKW	Byte 3 write de-skew register
190	WB4DSKW	Byte 4 write de-skew register
194	WB5DSKW	Byte 5 write de-skew register
198	WB6DSKW	Byte 6 write de-skew register
19C	WB7DSKW	Byte 7 write de-skew register
1A0	WDMDSKW	Write data mask de-skew register
1A4	RB8DSKW	Byte 8 read de-skew register
1A8	WB8DSKW	Byte 8 write de-skew register
1AC	B8_PHYCR	Byte 8 PHY control register
1B0	BIST_CTRL	BIST control register

Vangotech

1B4	BIST_START_ADDR	BIST start address
1B8	BIST_SIZE	BIST size
1BC	-	Reserved
1C0	BIST_RESULT	BIST result
1C4	BIST_FAIL_ADDR	BIST failed address
1C8	BIST_FAIL_DATA_LOW	BIST failed data low
1CC	BIST_FAIL_DATA_HIGH	BIST failed data high
1D0	BIST_EXP_DATA_LOW	BIST expected data low
1D4	BIST_EXP_DATA_HIGH	BIST expected data high
1D8	-	Reserved
1DC	-	Reserved
1E0	ECC_CTRLR	ECC control register
1E4	ECC_INTCR	ECC interrupt control register
1E8	ECC_CHR	ECC error channel register
1EC	ECC error addr	ECC error address
1F0	ECC_EBPLR	ECC error bit position low bytes
1F4	ECC_EBPHR	ECC error bit position high bytes
1F8	ECC_ERR1_COUNT	ECC one-bit error counts register
1FC	ECC_ERR2_COUNT	ECC two-bit error counts register

4.5.6. Register Description

4.5.6.1. Memory Controller Configuration Register

Table 4-10 Memory Controller Configuration Register (MCCR, Offset = 0x00)

Bit	Name	Type	Default Value	Description
31	share_slv	RW	0	When share_en is set to '1', share_slv will be validated. 0 = Controller plays master role in the share mode. 1 = Controller plays slave role in the share mode. this bit is fixed to 0.
30	share_en	RW	0	Share function means that controller can share command/address PHY with another controller. 0 = Disable 1 = Enable this bit is fixed to 0.
29	2TCmd_en	RW	0	2T command function makes command period last for two DRAM cycles and cs low moves to successive cycle. 0 = Disable

				1 = Enable If **FTDDR3030_SHARE_WRAPPER_NO** and **FTDDR3030_2TCMD_WRAPPER_NO** are configured, this bit is fixed to 0.
28	manual refresh enable	RW	0	FTDDR3030 checks the hardware input signal, manual_refresh, to issue one auto-refresh command and then the successive auto-refresh command will be issued based on the tREFI setting. 0 = Manually disables refresh despite of the hardware manual_refresh signal. 1 = FTDDR3030 will issue one auto-refresh command after receiving the manual_refresh trigger signal.
27	DisAPBErrResp	RW	1	This bit disables the APB error response register. The APB register interface will respond to PSIVERR under one of the following conditions: Write/Read to the non-existed registers Write to the read-only registers 0: Enable the APB error response 1: Disable the APB error response The default value is '1'.
[26:23]	WaitCycle	RW	0	When one command is pushed into the reorder engine, it starts counting the cycles until the command is popped out to the scheduler. If the cycle count of the command exceeds the wait cycle, this command will have the highest priority for popping out to the scheduler. 0: Wait cycle equals to 64 cycles. 1: Wait cycle equals to 2 * 64 cycles. 2: Wait cycle equals to 3 * 64 cycles. … 15: Wait cycle equals to 16 * 64 cycles.
[22:21]	Cont_Cmd_Limit	RW	0	Continuous command limitation in the scheduler If there is a pending read command in the scheduler, there will be at most n write commands that can be issued. If there is a pending write command in the scheduler, there will be at most n read commands that can be issued. If Cont_Cmd_Limit = 0, n = 32 If Cont_Cmd_Limit = 1, n = 64

Vangotech

				If Cont_Cmd_Limit = 2, n = 128 If Cont_Cmd_Limit = 3, n = 256
20	Dis_Reorder	RW	0	When the DDR command reorder engine checks the bank, the row information used to reduce the bank conflict performance will be impacted. Note: This bit should be set to '1' for the IP versions of (2.4.0), (2.5.0), and (2.6.0). These versions do not support the reorder function. 1 = Disable the DDR command reorder 0 = Enable the DDR command reorder
19	LPDDR_mode	RW	0	[19:16] = {LPDDR_mode, BYONE, DDR_mode} 0000: DDR3 1:2 mode 0011: DDR2 1:2 mode 0110: DDR1 1:1 mode 0111: DDR2 1:1 mode 1110: LPDDR1 1:1 mode 1011: LPDDR2 1:2 mode Others: Reserved The read values of [19:16] and the input signals of this register will be automatically set by the hardware definitions. PURE DDR1: **`define FTDDR3030_DDR_SUPPORT_TYPE_1** Hardware configuration will set BYONE = '1' and DDR_mode = '10'. PURE DDR2: **`define FTDDR3030_DDR_SUPPORT_TYPE_2** Hardware configuration will set LPDDR_mode = '0' and DDR_mode = '11'. PURE DDR3: **`define FTDDR3030_DDR_SUPPORT_TYPE_3** Hardware configuration will set LPDDR_mode = '0', BYONE = '0', and DDR_mode = '0'. COMBO DDR2/DDR3: **`define FTDDR3030_DDR_SUPPORT_TYPE_4** Hardware configuration will set LPDDR_mode = '0'. Combo DDR2/DDR3/LPDDR2: **`define FTDDR3030_DDR_SUPPORT_TYPE_5**
18	BYONE	RW	0	MClk versus DRAM clock frequency ratio 1: DRAM clock equals to the MClk clock frequency. 0: DRAM clock is twice of the MClk clock frequency.

Vangotech

				Please refer to bit 19 for more details.
[17:16]	DDR_mode	RW	0	DDR_mode 00 = DDR3 mode 01 = Reserved 10 = DDR1 mode 11 = DDR2 mode Please refer to bit 19 for more details.
[15:13]	Post-refresh command counts threshold	RW	0	Post-refresh command count threshold The DDR controller will post a refresh command until the count of the posted refresh commands exceed the threshold. 000 = One refresh command when the TREFI (Register Offset 0x1C[7:0]) counter is timed out. 001 = One refresh command when the TREFI (Register Offset 0x1C[7:0]) counter is timed out. 010 = Two posted refresh command thresholds 011 = Three posted refresh command thresholds 100 = Four posted refresh command thresholds 101 = Not used 110 = Not used 111 = Not used
[12:10]	Auto-refresh commands in the initial SDRAM sequence	RW	0	Only for the DDR2 SDRAM initialization sequence 000 = Reserved 001 = Two auto-refresh commands 010 = Three auto-refresh commands 011 = Four auto-refresh commands 100 = Five auto-refresh commands 101 = Six auto-refresh commands 110 = Seven auto-refresh commands 111 = Eight auto-refresh commands
[9:8]	Memory width	RW	0	Memory width 0 = 8-bit memory 1 = 16-bit memory 2 = 32-bit memory 3 = 64-bit memory
[7:6]	-	-	Rsvd	Reserved
[5:4]	DDRx memory address mapping table selection (AMTSEL)	RW	0	2'b00 = The mapping method of the AHB/AXI address to the DDRx SDRAM memory address is RA, BA, and CA. 2'b01 = The mapping method of the AHB/AXI address to the DDRx SDRAM memory address is BA,

Vangotech

				RA, and CA. 2'b10 = The mapping method of the Memory Bank Swizzle, the AHB address to the DDRx SDRAM memory address is RA, CA, BA, and CA. Please note that this field will work when the hardware configuration of FTDDR3030_BANK_SWIZZLE_YES is enabled. **AMTSEL = 2'b10 does not support the 8-bit memory.** For the 64-bit memory interface, the bank address mapping is inserted between the column addresses. ADDR maps to RA, CA, BA, and CA. Bits[9:8] of the system address are configured as the bank addresses for four banks of DDRx SDRAM. Bits[10:8] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM. For the 32-bit memory interface, the bank address mapping is inserted between the column addresses. ADDR maps to RA, CA, BA, and CA. Bits[8:7] of the system address are configured as the bank addresses for four banks of DDRx SDRAM. Bits[9:7] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM. For the 16-bit memory interface, the bank address mapping is inserted between the column addresses. ADDR maps to RA, CA, BA, and CA. Bits[7:6] of the system address are configured as the bank addresses for four banks of DDRx SDRAM. Bits[8:6] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM. 2'b11 = Reserved
3	DataCohe_en	RW	0	0: Data coherence check disable between multi-channels
[2:0]	Generate DQS sampling window (GDS)	RW	0	Before accessing the DDR memory, this field must be scanned from 0 to 7 until the read data are correctly captured. GDS depends on the PCB package trace length between the ASIC chip and DDRx SDRAM. Longer trace length will induce larger GDS value. Please refer to the DDRx PHY application note for more information. Note: The GDS value should be programmable at any

Vangotech

				time. If the GDS value is set in the ROM code, it should have the possibility to modify the GDS value by latching from jumper setting or other mechanism when booting from DDR SDRAM is failed.

4.5.6.2. Memory Controller State Control Register

The memory command can be issued once at the same time. For example, setting both bit[0] and bit[1] = 1 at the same time is not allowed. Users should separately set the required command one by one.

Table 4-11 Memory Controller State Control Register (MCSR, Offset = 0x04)

Bit	Name	Type	Default Value	Description
31	-	-	RsvdO	Reserved and opaque
[30:27]	-	-	Rsvd	Reserved
26	MR_rank_sel	RW	0	0 = MRS command is used for rank 0. 1 = MRS command is used for rank 1. Note: This setting is used for DDR1/DDR2/DDR3. There are separate registers (Register 0x114) for the LPDDR2 mrw command.
[25]	DDR access disable	RW	0	1 = Disable DDR access, the DDR commands will not be issued to DDR SDRAM. 0 = Enable DDR access
24	Warm Start	RW	0	Users can set this bit to '1' to the warm-start controller without initial DDRx SDRAM. This mode can be used when an ASIC chip is recovered from the power-down state and DDRx SDRAM is in the self-refresh mode. Please note that when an ASIC chip is at the power-down state, CKE of DDRx SDRAM should be kept low and the RESET_DRAM (DDR3 only) pin should be kept high from the power-on domain of an ASIC chip. When the command is completed, this bit will be cleared to zero. 1 = Warm start 0 = No effect
[23:19]	-	-	Rsvd	Reserved
18	DDR access disable complete	RO	0	1 = The DDR commands are all flushed, the system can change frequency. This bit is cleared when 0x04[25] write 0. 0 = The DDR access disable operation is not

Vangotech

				completed or not enabled.
17	Write-leveling fail or hardware disable	RO	1	Only for the DDR3 mode 1 = Any byte write-leveling failed or the hardware is disabled in DDR3 SDRAM. 0 = Write-leveling hardware is enabled and all bytes pass in DDR3 SDRAM.
16	Read-leveling fail or hardware disable	RO	1	Only for the DDR3 mode or LPDDR2 mode 1 = Any byte read-leveling fails or hardware is disabled in DDR3/LPDDR2 SDRAM. 0 = Read-leveling hardware is enabled and all bytes pass in DDR3/LPDDR2 SDRAM.
15	AHB/AXI command queue empty	RO	1	1 = All AHB/AXI command queues are empty. 0 = All AHB/AXI command queues are not empty
14	Memory controller command queue empty	RO	1	1 = Memory controller command queue is empty. 0 = Memory controller command queue is not empty.
13	ZQCL state	RO	0	1 = DDR3 SDRAM is at the ZQCL state (Only for DDR3). 0 = DDR3 SDRAM is at other states.
12	ZQCS state	RO	0	1 = DDR3 SDRAM is at the ZQCS state (Only for DDR3). 0 = DDR3 SDRAM is at other states.
11	Automatic power-down state	RO	0	1 = DDR3 SDRAM is at the automatic power-down state. 0 = DDR3 SDRAM is at other states.
10	Self-refresh state	RO	0	1 = DDRx SDRAM is at the self-refresh state. 0 = DDRx SDRAM is at other states.
9	Initial state	RO	0	1 = DDRx SDRAM is at the initial state. 0 = DDRx SDRAM is not at the initial state.
8	Initial OK	RO	0	1 = DDRx SDRAM initial is completed. 0 = DDRx SDRAM initial is not completed.
7	ZQCL	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be cleared to zero. 1 = Move the memory controller to the ZQCL state (Only for DDR3) 0 = No effect

Vangotech

6	ZQCS	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be cleared to zero. 1 = Move the memory controller to the ZQCS state (Only for DDR3) 0 = No effect
[5:4]	Register mode	RW	0	00 = Mode register (MR0 in the DDR3 mode) 01 = Extended mode register 1 (MR1 in the DDR3 mode) 10 = Extended mode register 2 (MR2 in the DDR3 mode) 11 = Extended mode register 3 (MR3 in the DDR3 mode)
3	Exit self-refresh command	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be cleared to zero. 1 = Move the memory controller from the self-refresh state 0 = No effect
2	MemCmd_srf (Self-refresh command)	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be cleared to zero. 1 = Move the memory controller to the self-refresh state 0 = No effect
1	MRS command	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be cleared to zero. Users should program MR/EMR/EMRS2/EMRS3 in the Register Offsets = 0x08 and 0x0C and offset = 0x04, bits[5:4] register mode. Before starting the MRS command: 1 = Move the memory controller to mode register state 0 = No effect
0	Initial command	RW	0	This bit is programmed to move the controller to enter the targeted state. When the command is completed, this bit will be

Vangotech

				cleared to zero. Setting this bit to '1' will start to perform the DDRx SDRAM initialization and DDR PHY compensation block calibration. After these tasks are completed, the signal bit[8] of initial_ok will be set to '1'. 1 = Move the memory controller to the DDRx SDRAM initial state 0 = No effect

4.5.6.3. Mode Register Set Value Register of MR and EMR (MRSVR0, Offset = 0x08)

This register defines the values of the mode registers and the extended mode registers to be used in the initial sequence. During the initialization of DDR2/DDR3 SDRAM, FTDDR3030 uses the values in this register for the MRS command. FTDDR3030 only supports AL (Additive Latency) = '0' in the DDR2/DDR3 mode.

If users want to modify the values of MR and EMR after initialization, Register Offset 0x08 should be programmed, and then bits[5:0] of Register Offsets 0x04 should be set to:

If 0x04[26] "MR_rank_sel" = 0,

0x04[5:0] = 000010: Send the MRS command to modify the MR value of DDRx SDRAM rank0

0x04[5:0] = 010010: Send the MRS command to modify the EMR value of DDRx SDRAM rank0

If 0x04[26] "MR_rank_sel" = 1,

0x04[5:0] = 000010: Send the MRS command to modify the MR value of DDRx SDRAM rank1

0x04[5:0] = 010010: Send the MRS command to modify the EMR value of DDRx SDRAM rank1

For the LPDDR1 case, the EMR value of LPDDR1 is put in 0x08[29:16] of the initial sequence. After the initial sequence, if users want to issue the MRS command to modify EMR of LPDDR1,

0x0C bits[13:0] should be filled, and then set bits[5:0] of Register Offsets 0x04 to issue one MRS

command:

If 0x04[26] "MR_rank_sel" = 0,

0x04[5:0] =100010 = Send the MRS command to modify the EMR value of LPDDR1 SDRAM rank0

Notes:

1. The mode register should be set as sequential burst length 4 in the DDR1/LPDDR1/DDR2 1:1 mode.

2. The mode register should be set as sequential burst length 8 in the DDR2/LPDDR2 1:2 mode.

3. The mode register should be set as Nibble sequential burst length fixed 8 in the DDR3 mode.

4. The mode register content write recovery should be set to the same value of the Timing Parameter1, Register Offset = 0x18, bits[23:20], t_{WR}.

5. AL should be set to '0' for the DDR2/DDR3 applications to improve the Write-to-Read or Read-to-Write performance.

6. In the DDR2 mode, bit 26 of the Register Offset = 0x08 DQS_B (DDR2 mode) and bit 3 of the Register Offset = 0x020 should be set to the same value. For example, DDR2 SDRAM and DDR PHY should operate in the same DQS mode, either in the differential mode or single-ended mode. DDR3 always operates in different DQS modes.

7. The "DLL_RESET" bit of mode register will be automatically controlled by FTDDR3030 in the DDR initialization sequence.

Table 4-12 Mode Register Set Value Register of MR and EMR (MRSVR0, Offset = 0x08)

Bit	Name	Type	Default Value	Description
[31:30]	-	-	Rsvd	Reserved
[29:16]	Extended mode register	RW	0	Extended mode register (MR1 in the DDR3 mode)
[15:14]	-	-	Rsvd	Reserved
[13:0]	Mode register	RW	0	Mode register (MR0 in the DDR3 mode)

4.5.6.4. Mode Register Set Value Register of EMR 2 and EMR 3 (MRSVR1, Offset = 0x0C)

This register defines the mode register value and the extended mode register value to be used in the

MRS command after initiating DDR2/DDR3 SDRAM.

If users want to modify the values of EMR2 and EMR3 after initialization, Register Offset 0x10 should be programmed first, and then bits[5:0] of Register Offsets 0x04 should be set to:

If 0x04[26] "MR_rank_sel" = 0,

0x04[5:0] = 5'b100010: Send the MRS command to modify the EMR2 value of DDRx SDRAM rank0

0x04[5:0] = 5'b110010: Send the MRS: command to modify the EMR3 value of DDRx SDRAM rank0

If 0x04[26] "MR_rank_sel" = 1,

0x04[5:0] = 5'b100010: Send the MRS command to modify the EMR2 value of DDRx SDRAM rank1

0x04[5:0] = 5'b110010: Send the MRS: command to modify the EMR3 value of DDRx SDRAM rank1

Table 4-13 Mode Register Set Value Register of EMR 2 and EMR 3 (MRSVR1, Offset = 0x0C)

Bit	Name	Type	Default Value	Description
[31:30]	-	-	Rsvd	Reserved
[29:16]	Extended mode register 3	RW	0	Extended mode register 3 (MR3 in the DDR3 mode)
[15:14]	-	-	Rsvd	Reserved
[13:0]	Extended mode register 2	RW	0	Extended mode register 2 (MR2 in the DDR3 mode)

4.5.6.5. External Rank0/Rank1 Register (EXRANKR, Offset = 0x10)

Please note that the rank1 base address will be the rank0 base address plus the rank0 size, and rank0 is always enabled.

Please refer to Table 4-14 ~ Table 4-16 for the rank type and size setting.

Please refer to Table 4-15 if FTDDR3030_LOGIC_ADDR_YES is configured.

Please refer to Table 4-16 if FTDDR3030_LOGIC_ADDR_NO is configured.

If FTDDR3030_ECC_YES is configured, only MA table 1 and 3 can be used.

Table 4-14 DDR RNK_TYPE Summary

FTDDR3030_LOGIC_ADDR_YES		**FTDDR3030_LOGIC_ADDR_NO**	
FTDDR3030_DDR_SUPPORT_TYPE_2,3,4,5		**FTDDR3030_DDR_SUPPORT_TYPE_1**	
RNK_TYPE	Row x Column x Bank	RNK_TYPE	Row x Column x Bank
0	13x10x3	0	12x8x2
1	14x10x3	1	12x9x2
2	15x10x3	2	13x9x2
3	16x10x3	3	12x10x2
4	16x11x3	4	13x10x2
5	14x9x3	5	14x10x2
6	14x11x3	6	Reserved
7	15x11x3	7	Reserved
(3) MA_table3		(4) MA_table4	

FTDDR3030_LOGIC_ADDR_NO		FTDDR3030_LOGIC_ADDR_NO	
FTDDR3030_DDR_SUPPORT_TYPE_2,3,4		FTDDR3030_DDR_SUPPORT_TYPE_5	
RNK_TYPE	Row x Column x Bank	RNK_TYPE	Row x Column x Bank
0	13x9x2	0	13x9x2
1	13x10x2	1	13x9x3
2	14x10x2	2	14x9x3
3	12x10x3	3	12x10x3
4	13x10x3	4	13x10x3
5	14x10x3	5	14x10x3
6	15x10x3	6	15x10x3
7	16x10x3	7	13x10x2

Table 4-15 External Rank0/Rank1 Register (EXRANKR, Offset = 0x10) when FTDDR3030_LOGIC_ADDR_YES is Configured

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:24]	RNK0_BASE	RW	0	This field is the 8-bit base address of the external rank0. RNK0 (Chip select 0) address range: {RNK0_BASE, 26'h0} ~ {{RNK0_BASE, 26'h0} + RNK0_SIZE - 1'b1} RNK1 (Chip select 1) address range: {{RNK0_BASE, 26'h0} + RNK0_SIZE} ~ {{RNK0_BASE, 26'h0} + RNK0_SIZE + RNK1_SIZE - 1'b1}
[23:17]	-	-	Rsvd	Reserved
16	RNK1_EN	RW	0	Rank1 enable
15	-	-	Rsvd	Reserved
[14:12]	RNK1_TYPE	RW	0	Select the rank type based on the MA table for rank1 **FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5** can be configured. For example, RA X CA X BA = (Row Address) X (Column Address) X (Bank Address) = 13 X 10 X 3, which indicates that the row address is 13 bits, the column address is 10 bits, and the bank address is 3 bits. 000 = 13 X 10 X 3 001 = 14 X 10 X 3 010 = 15 X 10 X 3 011 = 16 X 10 X 3 100 = 16 X 11 X 3 101 = 14 X 9 X 3

Vangotech

				110 = 14 X11X 3 111 = 15 X 11 X 3
11	-	-	Rsvd	Reserved
[10:8]	RNK1_SIZE	RW	0	Memory size of rank1 000: 64M bytes 001: 128M bytes 010: 256M bytes 011: 512M bytes 100: 1G bytes 101: 2G bytes 110: 4G bytes 111: 8G bytes
7	-	-	Rsvd	Reserved
[6:4]	RNK0_TYPE	RW	0	Select the rank type based on the MA table for rank0 FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5 can be configured. RA X CA X BA = (Row Address) X (Column Address) X (Bank Address) = 13 X 10 X 3, which indicates that the row address is 13 bits, the column address is 10 bits, and the bank address is 3 bits. 000= 13 X 10 X 3 001= 14 X 10 X 3 010= 15 X 10 X 3 011= 16 X 10 X 3 100= 16 X 11 X 3 101= 14 X 9 X 3 110= 14 X11X 3 111= 15 X 11 X 3
3	-	-	Rsvd	Reserved
[2:0]	RNK0_SIZE	RW	0	Memory size of rank0 000: 64M bytes 001: 128M bytes 010: 256M bytes 011: 512M bytes 100: 1G bytes 101: 2G bytes 110: 4G bytes 111: 8G bytes

Vangotech

Table 4-16 External Rank0/Rank1 Register (EXRANKR, Offset = 0x10) when FTDDR3030_LOGIC_ADDR_NO is Configured

Bit	Name	Type	Default Value	Description
[31:24]	RNK0_BASE	RW	0	This field is the 8-bit base address of the external rank0. RNK0 (Chip select 0) address range: {RNK0_BASE, 24'h0} ~ {{RNK0_BASE, 24'h0} + RNK0_SIZE - 1'b1} RNK1 (Chip select 1) address range: {{RNK0_BASE, 24'h0} + RNK0_SIZE } ~ {{RNK0_BASE, 24'h0} + RNK0_SIZE + RNK1_SIZE - 1'b1}
23	RNK0_BASE	RW	0	If **FTDDR3030_DDR_SUPPORT_TYPE_1** is configured, bits[31:23] is the 9-bit base address of the external rank0. This bit is reserved for other hardware configuration. RNK0 (Chip select 0) address range: {RNK0_BASE, 23'h0} ~ {{RNK0_BASE, 23'h0} + RNK0_SIZE - 1'b1}} RNK1 (Chip select 1) address range: {{RNK0_BASE, 23'h0} + RNK0_SIZE } ~ {{RNK0_BASE, 23'h0} + RNK0_SIZE + RNK1_SIZE - 1'b1} If **FTDDR3030_DDR_SUPPORT_TYPE_2/3/4/5** is configured, this bit is 0.
[22:17]	-	-	Rsvd	Reserved
16	RNK1_EN	RW	0	Rank1 enable
15	-	-	Rsvd	Reserved
[14:12]	RNK1_TYPE	RW	0	If FTDDR3030_DDR_SUPPORT_TYPE_1 is configured as (DDR1/LPDDR1), RNK1_TYPE will be as follows: 000 = 12 X 8 X 2 001 = 12 X 9 X 2 010 = 13 X 9 X 2 011 = 12 X 10 X 2 100 = 13 X 10 X 2 101 = 14 X 10 X 2 Others = Reserved If FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4 is configured as (DDR2/DDR3), RNK1_TYPE will be as follows: For example, RA X CA X BA = (Row Address) X (Column Address) X (Bank Address) = 13 X 10 X 2, which indicates that the row address is 13 bits, the column address is 10 bits, and the bank address is 2 bits. Note: If the hardware configuration, `define

Vangotech

				FTDDR3030_BANKNUM_4, is configured, BA (Bank Address) will only support 2 bits. 000 = 13 X 9 X 2 001 = 13 X 10 X 2 010 = 14 X 10 X 2 011 = 12 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 100 = 13 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 101 = 14 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 110 = 15 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 111 = 16 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) If FTDDR3030_DDR_SUPPORT_TYPE_5 is configured as (DDR2/DDR3/LPDDR2), RNK1_TYPE will be as follows: 000 = 13 X 9 X 2 001 = 13 X 9 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 010 = 14 X 9 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 011 = 12 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 100 = 13 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 101 = 14 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 110 = 15 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 111 = 13 X 10X2
11	-	-	Rsvd	Reserved
[10:8]	RNK1_SIZE	RW	0	Memory size of rank1 000: Reserved 001: 32M bytes 010: 64M bytes 011: 128M bytes 100: 256M bytes 101: 512M bytes 110: 1G bytes 111: 2G bytes If FTDDR3030_DDR_SUPPORT_TYPE_1 is configured,

Vangotech

				RNK_SIZE will be as follows: 000: 8M bytes 001: 16M bytes 010: 32M bytes 011: 64M bytes 100: 128M bytes 101: 256M bytes 110: 512M bytes 111: Reserved
7	-	-	Rsvd	Reserved
[6:4]	RNK0_TYPE	RW	0	If FTDDR3030_DDR_SUPPORT_TYPE_1 is configured as (DDR1/LPDDR1), RNK0_TYPE will be as follows: 000 = 12 X 8 X 2 001 = 12 X 9 X 2 010 = 13 X 9 X 2 011 = 12 X 10 X 2 100 = 13 X 10 X 2 101 = 14 X 10 X 2 Others = Reserved If FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4 is configured as (DDR2/DDR3), RNK0_TYPE will be as follows: For example, RA X CA X BA = (Row Address) X (Column Address) X (Bank Address) = 13 X 10 X 2, which indicates that the row address is 13 bits, the column address is 10 bits, and the bank address is 2 bits. Note: If the hardware configuration, `define FTDDR3030_BANKNUM_4, is configured, BA (Bank Address) will only support 2 bits. 000 = 13 X 9 X 2 001 = 13 X 10 X 2 010 = 14 X 10 X 2 011 = 12 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 100 = 13 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 101 = 14 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 110 = 15 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 111 = 16 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4)

Vangotech

				If FTDDR3030_DDR_SUPPORT_TYPE_5 is configured as (DDR2/DDR3/LPDDR2), RNK0_TYPE will be as follows: 000 = 13 X 9 X 2 001 = 13 X 9 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 010 = 14 X 9 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 011 = 12 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 100 = 13 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 101 = 14 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 110 = 15 X 10 X 3 (Not supported on `define FTDDR3030_BANKNUM_4) 111 = 13 X 10 X 2
3	-	-	Rsvd	Reserved
[2:0]	RNK0_SIZE	RW	0	Memory size of rank0 000: Reserved 001: 32M bytes 010: 64M bytes 011: 128M bytes 100: 256M bytes 101: 512M bytes 110: 1G bytes 111: 2G bytes If FTDDR3030_DDR_SUPPORT_TYPE_1 is configured, RNK0_SIZE will be as follows: 000: 8M bytes 001: 16M bytes 010: 32M bytes 011: 64M bytes 100: 128M bytes 101: 256M bytes 110: 512M bytes 111: Reserved

Table 4-17 through **Table 4-20** list the available combinations of RNK_TYPE and RNK_SIZE for different DDR1 memory widths.

Table 4-17 DDR1 Memory Width = 8 Bits

Vangotech

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (8M bytes)	-	-	-
3'h1 (16M bytes)	3'h3 (12 X 10 X 2)	16 Mb x 8 (128 Mb)	1
3'h2 (32M bytes)	3'h4 (13 X 10 X 2)	32 Mb x 8 (256 Mb)	1
3'h3 (64M bytes)	-	-	-
3'h4 (128M bytes)	-	-	-
3'h5 (256M bytes)	-	-	-
3'h6 (512M bytes)	-	-	-
3'h7 (1G bytes)	-	-	-

Table 4-18 DDR1 Memory Width = 16 Bits

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (8M bytes)	3'h0 (12X8X2)	4 Mb x16 (64 Mb)	1
3'h1 (16M bytes)	3'h1 (12X9X2)	8 Mb x16 (128 Mb)	1
3'h2 (32M bytes)	3'h2 (13X9X2)	16 Mb x16 (256 Mb)	1
3'h3 (12X10X2)	16 Mb x8 (128 Mb)		2
3'h3 (64M bytes)	3'h4 (13X10X2)	32 Mb x16 (512 Mb)	1
32 Mb x8 (256 Mb)		2	
3'h4 (128M bytes)	3'h5 (14X10X2)	64 Mb x16 (1 Gb)	1
3'h5 (256M bytes)	-	-	-
3'h6 (512M bytes)	-	-	-
3'h7 (1G bytes)	-	-	-

Table 4-19 DDR1 Memory Width = 32 Bits

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (8M bytes)	-	-	-
3'h1 (16M bytes)	3'h0 (12X8X2)	4 Mb x16 (64 Mb)	2
3'h2 (32M bytes)	3'h1 (12X9X2)	8 Mb x16 (128 Mb)	2
3'h3 (64M bytes)	3'h2 (13X9X2)	16 Mb x16 (256 Mb)	2
	3'h3 (12X10X2)	16 Mb x8 (128 Mb)	4
3'h4 (128M bytes)	3'h4 (13X10X2)	32 Mb x16 (512 Mb)	2
		32 Mb x8 (256 Mb)	4
3'h5 (256M bytes)	3'h5 (14X10X2)	64 Mb x16 (1 Gb)	2
3'h6 (512M bytes)	-	-	-
3'h7 (1G bytes)	-	-	-

Table 4-20 DDR1 Memory Width = 64 Bits

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number

Vangotech

3'h0 (8M bytes)	-	-	-
3'h1 (16M bytes)	-	-	-
3'h2 (32M bytes)	3'h0 (12X8X2)	4 Mb x16 (64 Mb)	4
3'h3 (64M bytes)	3'h1 (12X9X2)	8 Mb x16 (128 Mb)	4
3'h4 (128M bytes)	3'h2 (13X9X2)	16 Mb x16 (256 Mb)	4
	3'h3 (12X10X2)	16 Mb x8 (128 Mb)	8
3'h5 (256M bytes)	3'h4 (13X10X2)	32 Mb x16 (512 Mb)	4
		32 Mb x8 (256 Mb)	8
3'h6 (512M bytes)	3'h5 (14X10X2)	64 Mb x16 (1 Gb)	4
3'h7 (1G bytes)	-	-	-

Table 4-21 lists the DDR2 configuration by different MA tables.

Table 4-21 DDR2 Configuration by Different MA Tables

DDR2	**FTDDR3030_LOGIC_ADDR_YES**	**FTDDR3030_LOGIC_ADDR_NO**	
	RNK_TYPE (MA1)	**RNK_TYPE (MA3)**	**RNK_TYPE (MA4)**
13x9x2	N/A	0	0
13x10x2	N/A	1	N/A
14x10x2	N/A	2	N/A
13x10x3	0	4	4
14x10x3	1	5	5
15x10x3	2	6	6
16x10x3	3	7	N/A

Table 4-22 through Table 2-19 list the available combinations of RNK_TYPE and RNK_SIZE for different DDR2 memory widths when **FTDDR3030_LOGIC_ADDR_NO** is defined.

Table 4-22 DDR2 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	(13X10X2)	32 Mb x8 (256 Mb)	1
3'h2 (64M bytes)	(14X10X2)	64 Mb x8 (512 Mb)	1
3'h3 (128M bytes)	(14X10X3)	128 Mb x8 (1 Gb)	1
3'h4 (256M bytes)	(15X10X3)	256 Mb x8 (2 Gb)	1
3'h5 (512M bytes)	(16X10X3)	512 Mb x8 (4 Gb)	1
3'h6 (1G bytes)	-	-	-
3'h7 (2G bytes)	-	-	-

Table 4-23 DDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

vangotech

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	(13X9X2)	16 Mb x16 (256 Mb)	1
3'h2 (64M bytes)	(13X10X2)	32 Mb x16 (512 Mb)	1
		32 Mb x8 (256 Mb)	2
3'h3 (128M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	1
	(14X10X2)	64 Mb x8 (512 Mb)	2
3'h4 (256M bytes)	(14X10X3)	128 Mb x16 (2 Gb)	1
		128 Mb x8 (1 Gb)	2
3'h5 (512M bytes)	(15X10X3)	256 Mb x16 (4 Gb)	1
		256 Mb x8 (2 Gb)	2
3'h6 (1G bytes)	(16X10X3)	512 Mb x8 (4 Gb)	2
3'h7 (2G bytes)	-	-	-

Table 4-24 DDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	(13X9X2)	16 Mb x16 (256 Mb)	2
3'h3 (128M bytes)	(13X10X2)	32 Mb x16 (512 Mb)	2
		32 Mb x8 (256 Mb)	4
3'h4 (256M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	2
	(14X10X2)	64 Mb x8 (512 Mb)	4
3'h5 (512M bytes)	(14X10X3)	128 Mb x16 (2 Gb)	2
		128 Mb x8 (1 Gb)	4
3'h6 (1G bytes)	(15X10X3)	256 Mb x16 (4 Gb)	2
		256 Mb x8 (2 Gb)	4
3'h7 (2G bytes)	(16X10X3)	512 Mb x8 (4 Gb)	4

Table 4-25 DDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	-	-	-
3'h3 (128M bytes)	(13X9X2)	16 Mb x16 (256 Mb)	4
3'h4 (256M bytes)	(13X10X2)	32 Mb x16 (512 Mb)	4
		32 Mb x8 (256 Mb)	8
3'h5 (512M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	4
	(14X10X2)	64 Mb x8 (512 Mb)	8

Vangotech

3'h6 (1G bytes)	(14X10X3)	128 Mb x16 (2 Gb)	4
		128 Mb x8 (1 Gb)	8
3'h7 (2G bytes)	(15X10X3)	256 Mb x16 (4 Gb)	4
		256 Mb x8 (2 Gb)	8

Table 4-26 through Table 2-23 list the available combinations of RNK_TYPE and RNK_SIZE for different DDR2 memory widths when **FTDDR3030_LOGIC_ADDR_YES** is defined.

Table 4-26 DDR2 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	3'h0 (13X10X3)	64 Mb x8 (512 Mb)	1
3'h1 (128M bytes)	3'h1 (14X10X3)	128 Mb x8 (1 Gb)	1
3'h2 (256M bytes)	3'h2 (15X10X3)	256 Mb x8 (2 Gb)	1
3'h3 (512M bytes)	3'h3 (16X10X3)	512 Mb x8 (4 Gb)	1
3'h4 (1G bytes)	-	-	-
3'h5 (2G bytes)	-	-	-
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-27 DDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	3'h0 (13X10X3)	64Mb x16 (1Gb)	1
		64Mb x8 (512 Mb)	2
3'h2 (256M bytes)	3'h1 (14X10X3)	128 Mb x16 (2 Gb)	1
		128 Mb x8 (1 Gb)	2
3'h3 (512M bytes)	3'h2 (15X10X3)	256 Mb x16 (4 Gb)	1
		256 Mb x8 (2 Gb)	2
3'h4 (1G bytes)	3'h3 (16X10X3)	512 Mb x8 (4 Gb)	2
3'h5 (2G bytes)	-	-	-
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-28 DDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	3'h0 (13X10X3)	64Mb x8 (512 Mb)	4
		64Mb x16 (1 Gb)	2
3'h3 (512M bytes)	3'h1 (14X10X3)	128 Mb x8 (1 Gb)	4
		128 Mb x16 (2 Gb)	2

Vangotech

3'h4 (1G bytes)	3'h2 (15X10X3)	256 Mb x8 (2 Gb)	4
		256 Mb x16 (4 Gb)	2
3'h5 (2G bytes)	3'h3 (16X10X3)	512 Mb x8 (4 Gb)	4
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-29 DDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	-	-	-
3'h3 (512M bytes)	3'h0 (13x10x3)	64Mb x8 (512 Mb)	8
		64Mb x16 (1 Gb)	4
3'h4 (1G bytes)	3'h1 (14x10x3)	128Mb x8 (1 Gb)	8
		128Mb x16 (2 Gb)	4
3'h5 (2G bytes)	3'h2 (15x10x3)	256Mb x8 (2 Gb)	8
		256Mb x16 (4 Gb)	4
3'h6 (4G bytes)	3'h3 (16x10x3)	512Mb x8 (4 Gb)	8
3'h7 (8G bytes)	-	-	-

Table 4-30 lists the DDR3 configuration by different MA tables.

Table 4-30 DDR3 Configuration by Different MA Tables

DDR3	**FTDDR3030_LOGIC_ADDR_YES**	**FTDDR3030_LOGIC_ADDR_NO**	
	RNK_TYPE (MA1)	**RNK_TYPE (MA3)**	**RNK_TYPE (MA4)**
12x10x3	N/A	3	3
13x10x3	0	4	4
14x10x3	1	5	5
15x10x3	2	6	6
16x10x3	3	7	N/A
16x11x3	4	N/A	N/A

Table 4-31 through Table 2-28 list the available combinations of RNK_TYPE and RNK_SIZE in different DDR3 memory widths when **FTDDR3030_LOGIC_ADDR_NO** is defined.

Table 4-31 DDR3 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	(13X10X3)	64 Mb x8 (512 Mb)	1
3'h3 (128M bytes)	(14X10X3)	128 Mb x8 (1 Gb)	1
3'h4 (256M bytes)	(15X10X3)	256 Mb x8 (2 Gb)	1

Vangotech

3'h5 (512M bytes)	(16X10X3)	512 Mb x8 (4 Gb)	1
3'h6 (1G bytes)	-	-	-
3'h7 (2G bytes)	-	-	-

Table 4-32 DDR3 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	(12X10X3)	32 Mb x16 (512 Mb)	1
3'h3 (128M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	1
		64 Mb x8 (512 Mb)	2
3'h4 (256M bytes)	(14X10X3)	128 Mb x16 (2 Gb)	1
		128 Mb x8 (1 Gb)	2
3'h5 (512M bytes)	(15X10X3)	256 Mb x16 (4 Gb)	1
		256 Mb x8 (2 Gb)	2
3'h6 (1G bytes)	(16X10X3)	512 Mb x16 (8 Gb)	1
		512 Mb x8 (4 Gb)	2
3'h7 (2G bytes)	-	-	-

Table 4-33 DDR3 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	-	-	-
3'h3 (128M bytes)	(12X10X3)	32 Mb x16 (512 Mb)	2
3'h4 (256M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	2
		64 Mb x8 (512 Mb)	4
3'h5 (512M bytes)	(14X10X3)	128 Mb x16 (2 Gb)	2
		128 Mb x8 (1 Gb)	4
3'h6 (1G bytes)	(15X10X3)	256 Mb x16 (4 Gb)	2
		256 Mb x8 (2 Gb)	4
3'h7 (2G bytes)	(16X10X3)	512 Mb x16 (8 Gb)	2
		512 Mb x8 (4 Gb)	4

Table 4-34 DDR3 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	-	-	-
3'h3 (128M bytes)	-	-	-
3'h4 (256M bytes)	(12X10X3)	32 Mb x16 (512 Mb)	4
3'h5 (512M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	4

Vangotech

		64 Mb x8 (512 Mb)	8
3'h6 (1G bytes)	(14X10X3)	128 Mb x16 (2 Gb)	4
		128 Mb x8 (1 Gb)	8
3'h7 (2G bytes)	(15X10X3)	256 Mb x16 (4 Gb)	4
		256 Mb x8 (2 Gb)	8

Table 4-35 through Table 2-32 list the available combinations of RNK_TYPE and RNK_SIZE in different DDR3 memory widths when **FTDDR3030_LOGIC_ADDR_YES** is defined.

Table 4-35 DDR3 Memory Width = 8 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	3'h0(13X10X3)	64 Mb x8 (512 Mb)	1
3'h1 (128M bytes)	3'h1(14X10X3)	128 Mb x8 (1 Gb)	1
3'h2 (256M bytes)	3'h2(15X10X3)	256 Mb x8 (2 Gb)	1
3'h3 (512M bytes)	3'h3(16X10X3)	512 Mb x8 (4 Gb)	1
3'h4 (1G bytes)	3'h4(16X11X3)	1Gb x8 (8 Gb)	1
3'h5 (2G bytes)	-	-	-
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-36 DDR3 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	3'h0 (13X10X3)	64Mb x16 (1Gb)	1
		64Mb x8 (512 Mb)	2
3'h2 (256M bytes)	3'h1 (14X10X3)	128 Mb x16 (2 Gb)	1
		128 Mb x8 (1 Gb)	2
3'h3 (512M bytes)	3'h2 (15X10X3)	256 Mb x16 (4 Gb)	1
		256 Mb x8 (2 Gb)	2
3'h4 (1G bytes)	3'h3 (16X10X3)	512 Mb x16 (8 Gb)	1
		512 Mb x8 (4 Gb)	2
3'h5 (2G bytes)	3'h4 (16X11X3)	1Gb x8 (8 Gb)	2
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-37 DDR3 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	3'h0 (13X10X3)	64Mb x8 (512 Mb)	4
		64Mb x16 (1 Gb)	2

Vangotech

3'h3 (512M bytes)	3'h1 (14X10X3)	128 Mb x8 (1 Gb)	4
		128 Mb x16 (2 Gb)	2
3'h4 (1G bytes)	3'h2 (15X10X3)	256 Mb x8 (2 Gb)	4
		256 Mb x16 (4 Gb)	2
3'h5 (2G bytes)	3'h3 (16X10X3)	512 Mb x8 (4 Gb)	4
		512 Mb x16 (8 Gb)	2
3'h6 (4G bytes)	3'h4 (16X11X3)	1 Gb x8 (8 Gb)	4
3'h7 (8G bytes)	-	-	-

Table 4-38 DDR3 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	**RNK_TYPE**	**DDR Chip**	**DDR Chip Number**
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	-	-	-
3'h3 (512M bytes)	3'h0 (13x10x3)	64Mb x8 (512 Mb)	8
		64Mb x16 (1 Gb)	4
3'h4 (1G bytes)	3'h1 (14x10x3)	128Mb x8 (1 Gb)	8
		128Mb x16 (2 Gb)	4
3'h5 (2G bytes)	3'h2 (15x10x3)	256Mb x8 (2 Gb)	8
		256Mb x16 (4 Gb)	4
3'h6 (4G bytes)	3'h3 (16x10x3)	512Mb x8 (4 Gb)	8
		512Mb x16 (8 Gb)	4
3'h7 (8G bytes)	3'h4 (16X11X3)	1Gb x8 (8 Gb)	8

Table 4-39 lists the LPDDR2 configuration by different MA tables.

Table 4-39 LPDDR2 Configuration by Different MA Tables

LPDDR2	**FTDDR3030_LOGIC_ADDR_YES**	**FTDDR3030_LOGIC_ADDR_NO**	
	RNK_TYPE (MA1)	**RNK_TYPE (MA3)**	**RNK_TYPE (MA4)**
13x9x2	N/A	0	0
13x10x2	N/A	1	7
13x9x3	N/A	N/A	1
13x10x3	0	4	4
14x9x3	5	N/A	2
14x10x3	1	5	5
15x10x3	2	6	6
15x11x3	7	N/A	N/A

Table 4-40 through **Table 4-42** list the available combinations of RNK_TYPE and RNK_SIZE in different

Vangotech

LPDDR2 memory widths when FTDDR3030_LOGIC_ADDR_NO is defined.

Table 4-40 LPDDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	(13X9X2)	16 Mb x16 (256 Mb)	1
3'h2 (64M bytes)	(13x10x2)	32Mb x16 (512 Mb)	1
3'h3 (128M bytes)	(13X10X3)	64 Mb x16 (1 Gb)	1
3'h4 (256M bytes)	(14X10X3)	128 Mb x16 (2 Gb)	1
3'h5 (512M bytes)	-	-	-
3'h6 (1G bytes)	-	-	-
3'h7 (2G bytes)	-	-	-

Table 4-41 LPDDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	(13X9X2)	16 Mb x32 (512 Mb)	1
		16 Mb x16 (256 Mb)	2
3'h3 (128M bytes)	(13X9X3)	32 Mb x32 (1 Gb)	1
	(13x10x2)	32Mb x16 (512 Mb)	2
3'h4 (256M bytes)	(14X9X3)	64 Mb x32 (2 Gb)	1
	(13X10X3)	64 Mb x16 (1 Gb)	2
3'h5 (512M bytes)	(14X10X3)	128 Mb x32 (4 Gb)	1
		128 Mb x16 (2 Gb)	2
3'h6 (1G bytes)	(15X10X3)	256 Mb x32 (8 Gb)	1
3'h7 (2G bytes)	-	-	-

Table 4-42 LPDDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_NO is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (16M bytes)	-	-	-
3'h1 (32M bytes)	-	-	-
3'h2 (64M bytes)	-	-	-
3'h3 (128M bytes)	(13X9X2)	16 Mb x32 (512 Mb)	2
		16 Mb x16 (256 Mb)	4
3'h4 (256M bytes)	(13X9X3)	32 Mb x32 (1 Gb)	2
	(13x10x2)	32Mbx16 (512 Mb)	4
3'h5 (512M bytes)	(14X9X3)	64 Mb x32 (2 Gb)	2
	(13X10X3)	64 Mb x16 (1 Gb)	4
3'h6 (1G bytes)	(14X10X3)	128 Mb x32 (4 Gb)	2
		128 Mb x16 (2 Gb)	4

vangotech

3'h7 (2G bytes)	(15X10X3)	256 Mb x32 (8 Gb)	2

Table 4-43 through **Table 4-45** list the available combinations of RNK_TYPE and RNK_SIZE in different LPDDR2 memory widths when FTDDR3030_LOGIC_ADDR_YES is defined.

Table 4-43 LPDDR2 Memory Width = 16 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	3'h0 (13x10x3)	64Mbx16 (1Gb)	1
3'h2 (256M bytes)	3'h1 (14x10x3)	128Mbx16(2Gb)	1
3'h3 (512M bytes)	-	-	-
3'h4 (1G bytes)	3'h7(15x11x3)	512Mbx16(8Gb)	1
3'h5 (2 G bytes)	-	-	-
3'h6 (4 G bytes)	-	-	-
3'h7 (8 G bytes)	-	-	-

Table 4-44 LPDDR2 Memory Width = 32 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	3'h0 (13x10x3)	64Mbx16 (1Gb)	2
	3'h5 (14x9x3)	64Mbx32 (2Gb)	1
3'h3 (512M bytes)	3'h1 (14x10x3)	128Mbx16(2Gb)	2
		128Mbx32(4Gb)	1
3'h4 (1G bytes)	3'h2(15x10x3)	256Mbx32(8Gb)	1
3'h5 (2G bytes)	3'h7(15x11x3)	512Mbx16(8Gb)	2
3'h6 (4G bytes)	-	-	-
3'h7 (8G bytes)	-	-	-

Table 4-45 LPDDR2 Memory Width = 64 Bits when FTDDR3030_LOGIC_ADDR_YES is defined

RNK_SIZE	RNK_TYPE	DDR Chip	DDR Chip Number
3'h0 (64M bytes)	-	-	-
3'h1 (128M bytes)	-	-	-
3'h2 (256M bytes)	-	-	-
3'h3 (512M bytes)	3'h0 (13x10x3)	64Mbx16 (1Gb)	4
	3'h5 (14x9x3)	64Mbx32 (2Gb)	2
3'h4 (1G bytes)	3'h1 (14x10x3)	128Mbx16(2Gb)	4
		128Mbx32(4Gb)	2
3'h5 (2G bytes)	3'h2(15x10x3)	256Mbx32(8Gb)	2
3'h6 (4G bytes)	3'h7(15x11x3)	512Mbx16(8Gb)	2
3'h7 (8G bytes)	-	-	-

Vangotech

4.5.6.6. Timing Parameter 0 Register (TMPR0, Offset = 0x14)

Please note that all parameters are related to the MClk cycles. If the "MClk:DRAM clock = 1:2" mode is used, the parameters should be set to **half value** of the DRAM clock cycles. The programmable value of 0 is invalid.

Table 4-46 Timing Parameter 0 Register (TMPR0, Offset = 0x14)

Bit	Name	Type	Default Value	Description
[31:24]	TRFC	RW	0	Refresh-to-Active/Refresh command period 8'h0 = Not used 8'h1 = 2 MClk cycles 8'h2 = 3 MClk cycles ... 8'hff = 256 MClk cycles
[23:21]	-	-	Rsvd	Reserved
[20:16]	TFAW	RW	0	4-bank activate windows 5'h0 = Not used 5'h1 = 2 MClk cycles 5'h2 = 3 MClk cycles ... 5'h1f = 32 MClk cycles Notes: 1. The 5th Active command next to four consecutive Active commands will be constrained by the greater value of TRC, TRAS, and TFAW. 2. The minimum Active-to-Active command period for different banks will also be constrained by the priority of command scheduler selecting Write/Read/Active commands. The Write/Read commands have higher priority than the Active command.
[15:14]	-	-	Rsvd	Reserved
[13:8]	TRC	RW	0	Active-to-Active command period for the same bank 6'h0 = Not used 6'h1 = 2 MClk cycles 6'h2 = 3 MClk cycles ... 6'h3f = 64 MClk cycles
[7:5]	-	-	Rsvd	Reserved
[4:0]	TRAS	RW	0	Active-to-Precharge period This parameter specifies the minimum period of an opened row.

Vangotech

				5'h0 = Not used 5'h1 = 2 MClk cycles 5'h2 = 3 MClk cycles … 5'h1f = 32 MClk cycles

4.5.6.7. Timing Parameter 1 Register (TMPR1, Offset = 0x18)

Please note that some parameters are related to the MClk cycles. If the "MClk:DRAM clock = 1:2" mode is used, the parameters should be set to **half value** of the DRAM clock cycles, except **TWR** and **TWTR**. **TWR** and **TWTR** should be set in terms of the DRAM clock cycles. t_{WL} is the DRAM write latency.

For DDR1, t_{WL} = 1

For DDR2, t_{WL} = AL (Additive Latency) + CL (CAS Latency) - 1

For DDR3, t_{WL} = AL (Additive Latency) + CWL (CAS Write Latency)

(FTDDR3030 only supports AL = '0'.)

Table 4-47 Timing Parameter 1 Register (TMPR1, Offset = 0x18)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:28]	TWTR	RW	0	Internal write-to-read delay READ command after the last write data delay cycle If the MClk:DRAM clock is in the "1:1" mode: 3'h0 = Not used 3'h1 = 2 DRAM clock cycles 3'h2 = 3 DRAM clock cycles 3'h3 = 4 DRAM clock cycles 3'h4 = 5 DRAM clock cycles 3'h5 = 6 DRAM clock cycles 3'h6 = 7 DRAM clock cycles 3'h7 = 8 DRAM clock cycles If the MClk:DRAM clock is in the "1:2" mode: tWL is odd. 3'h0 = Not used 3'h1 = 3 DRAM clock cycles 3'h2 = 5 DRAM clock cycles 3'h3 = 5 DRAM clock cycles 3'h4 = 7 DRAM clock cycles 3'h5 = 7 DRAM clock cycles

Vangotech

				3'h6 = 9 DRAM clock cycles 3'h7 = 9 DRAM clock cycles tWL is even. 3'h0 = Not used 3'h1 = 4 DRAM clock cycles 3'h2 = 4 DRAM clock cycles 3'h3 = 6 DRAM clock cycles 3'h4 = 6 DRAM clock cycles 3'h5 = 8 DRAM clock cycles 3'h6 = 8 DRAM clock cycles 3'h7 = 10 DRAM clock cycles
27	-	-	Rsvd	Reserved
[26:24]	TRTP	RW	0	Internal read-to-precharge delay If FTDDR3030_DDR_SUPPORT_TYPE_1 is configured, this field should be set as 3'h1. 3'h0 = Not used 3'h1 = 2 MClk cycles 3'h2 = 3 MClk cycles ... 3'h7 = 8 MClk cycles
[23:20]	TWR	RW	0	Write recovery time The last non-write data cycle at the MCLK rising edge to the precharge command cycle. If the MClk:DRAM clock is in the "1:1" mode: 4'h0 = Not used 4'h1 = 2 DRAM clock cycles 4'h2 = 3 DRAM clock cycles 4'h3 = 4 DRAM clock cycles ... 4'hf = 16 DRAM clock cycles If the MClk:DRAM clock is in the "1:2" mode: tWL is odd. 3'h0 = Not used 3'h1 = 3 DRAM clock cycles 3'h2 = 5 DRAM clock cycles 3'h3 = 5 DRAM clock cycles 3'h4 = 7 DRAM clock cycles 3'h5 = 7 DRAM clock cycles 3'h6 = 9 DRAM clock cycles 3'h7 = 9 DRAM clock cycles 3'h8 = 11 DRAM clock cycles 3'h9 = 11 DRAM clock cycles

Vangotech

				3'ha = 13 DRAM clock cycles 3'hb = 13 DRAM clock cycles 3'hc = 15 DRAM clock cycles 3'hd = 15 DRAM clock cycles 3'he = 17 DRAM clock cycles 3'hf = 17 DRAM clock cycles **If the MClk:DRAM clock is in the "1:2" mode:** **tWL is even.** 3'h0 = Not used 3'h1 = 4 DRAM clock cycles 3'h2 = 4 DRAM clock cycles 3'h3 = 6 DRAM clock cycles 3'h4 = 6 DRAM clock cycles 3'h5 = 8 DRAM clock cycles 3'h6 = 8 DRAM clock cycles 3'h7 = 10 DRAM clock cycles 3'h8 = 10 DRAM clock cycles 3'h9 = 12 DRAM clock cycles 3'ha = 12 DRAM clock cycles 3'hb = 14 DRAM clock cycles 3'hc = 14 DRAM clock cycles 3'hd = 16 DRAM clock cycles 3'he = 16 DRAM clock cycles 3'hf = 18 DRAM clock cycles
[19:16]	TMOD	RW	0	The mode register sets the command update delay. 4'h0 = Not used 4'h1 = 2 MClk cycles 4'h2 = 3 MClk cycles ... 4'hf = 16 MClk cycles
[15:12]	TMRD	RW	0	Cycle time of the load mode register command 4'h0 = Not used 4'h1 = 2 MClk cycles 4'h2 = 3 MClk cycles 4'h3 = 4 MClk cycles ... 4'hf = 16 MClk cycles
[11:8]	TRP	RW	0	Precharge period 4'h0 = Not used 4'h1 = 2 MClk cycles 4'h2 = 3 MClk cycles ...

Vangotech

				4'hf = 16 MClk cycles
[7:4]	TRRD	RW	0	Active-to-Active command period for different banks 4'h0 = Not used 4'h1 = 2 MClk cycles 4'h2 = 3 MClk cycles ... 4'hf = 16 MClk cycles Note: The minimum Active-to-Active command period for different banks will be constrained by the priority of Write/Read/Active commands selected by the command scheduler. The Write/Read commands have higher priority than the Active command.
[3:0]	TRCD	RW	0	The minimum delay between the active and read/write commands 4'h0 = Not used 4'h1 = 2 MClk cycles 4'h2 = 3 MClk cycles ... 4'hf = 16 MClk cycles

4.5.6.8. Timing Parameter 2 Register (TMPR2, Offset = 0x1C)

Please note that some parameters are related to the MClk cycles. If the "MClk:DRAM clock = 1:2" mode is used, the parameters should be set to **half value** of the DRAM clock cycles.

t_{WL} = Write latency in terms of the DRAM clock cycles

t_{RL} = Read latency in terms of the DRAM clock cycles

FTDDR3030 only supports AL = 0.

For DDR1, t_{WL} = 1

For DDR2, t_{WL} = AL (Additive Latency) + CL (CAS Latency) - 1

For DDR3, t_{WL} = AL (Additive Latency) + CWL (CAS Write Latency)

For DDR1, t_{RL} = CL (CAS Latency)

For DDR2, DDR3, t_{RL} = AL (Additive Latency) + CL (CAS Latency)

Users can refer to Appendix 4.5 for the interval counting examples of the Write-to-Read, Write-to-Write, Read-to-Read, and Read-to-Write commands.

Vangotech

Table 4-48 Timing Parameter 2 Register (TMPR2, Offset = 0x1C)

Bit	Name	Type	Default Value	Description
[31:30]	TWtoR_ctrl	RW	0	Controller specific Additional delay cycles from the write command to the read command for accessing different ranks The programmed value compensates the unbalanced trace or I/O delay between different ranks of DDRx SDRAM to avoid the DQS driving contention. **If the MClk:DRAM clock is in the "1:1" mode:** 2'h0 = No additional delay clock cycles 2'h1 = Add additional one DRAM clock cycle 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional three DRAM clock cycles **If the MClk:DRAM clock is in the "1:2" mode**: **(tRL - tWL) is odd.** 2'h0 = No additional delay clock cycles 2'h1 = Add additional two DRAM clock cycles 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional four DRAM clock cycles **(tRL - tWL) is even.** 2'h0 = No additional delay clock cycles 2'h1 = No additional delay clock cycles 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional two DRAM clock cycles
[29:28]	TWtoW_ctrl	RW	0	Controller specific Additional delay cycles from the write command to the write command If the MClk:DRAM clock is in the "1:1" mode: 2'h0 = No additional delay clock cycles 2'h1 = Add additional one DRAM clock cycle 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional two DRAM clock cycles If the MClk:DRAM clock is in the "1:2" mode: 2'h0 = No additional delay clock cycles 2'h1 = Add additional two DRAM clock cycles 2'h2 = Add additional four DRAM clock cycles 2'h3 = Add additional six DRAM clock cycles
[27:26]	TRtoR_ctrl	RW	0	Controller specific Additional delay cycles from the read command to the read command If the MClk:DRAM clock is in the "1:1" mode:

Vangotech

				2'h0 = No additional delay clock cycles 2'h1 = Add additional one DRAM clock cycle 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional three DRAM clock cycles If the MClk:DRAM clock is in the "1:2" mode: 2'h0 = No additional delay clock cycles 2'h1 = Add additional two DRAM clock cycles 2'h2 = Add additional four DRAM clock cycles 2'h3 = Add additional six DRAM clock cycles
[25:24]	TRtoW_ctrl	RW	0	Controller specific Additional delay cycles from the read command to the write command DQS will be driven by different devices in the read and write operations. The programmed value will compensate the round-trip delay between the chip and DDRx SDRAM to avoid the DQS driving contention. Users should adjust this value to avoid the Read DQS/DQ and the Write DQS/DQ driving contention. If the MClk:DRAM clock is in the "1:1" mode: 2'h0 = No additional delay clock cycles 2'h1 = Add additional one DRAM clock cycle 2'h2 = Add additional two DRAM clock cycles 2'h3 = Add additional three DRAM clock cycles **If the MClk:DRAM clock is in the "1:2" mode:** 2'h0 = No additional delay clock cycles 2'h1 = Add additional two DRAM clock cycles 2'h2 = Add additional four DRAM clock cycles 2'h3 = Add additional six DRAM clock cycles
[23:16]	-	-	Rsvd	Reserved
[15:8]	TXSR	RW	0	Exit from the self-refresh mode to a command period This value should be programmed to the maximum of tXSNR and tXSRD in the DDR1/DDR2 mode. This value should be programmed to the maximum of tXS and tXSDLL in the DDR3 mode. This value should be programmed tXSR in the LPDDR1 and LPDDR2 mode. 8'h0 = Not used 8'h1 = 1x8 MClk cycles 8'h2 = 2x8 MClk cycles ... 8'hff = 255x8 MClk cycles Notes:

Vangotech

				(1) If FTDDR030_CMDBUF_YES is configured, there will be four additional MClk guard time to compensate the internal exiting self-refresh command to the DDR PHY I/O delay If FTDDR030_CMDBUF_NO is configured, there will be three additional MClk guard time to compensate the internal exiting self-refresh command to the DDR PHY I/O delay (2) The first active command after exiting the self-refresh state should conform to all the following timing constraints: 1. TXSR 2. DDRx PHY BYTEn_UPDATE_ACK assertion time (For the DDR23 Combo PHY) 3. tZQCS and tZQoper if Register Offset 0x4C bits[17:16] ZQCS/ZQCL update is enabled.
[7:0]	TREFI	RW	0	Average periodic refresh interval One refresh command should be issued if the refresh counter equals to the refresh interval. Please note that bit 7 is only for selecting between x32T and x8T. Bit 7 = 1'b0: 8'b0_000_0000 = Not used 8'b0_000_0001 = 1 x32 MClk cycles 8'b0_000_0010 = 2 x32 MClk cycles ... 8'b0_111_1111 = 127 x32 MClk cycles bit 7 = 1'b1: 8'b1_000_0000 = Not used 8'b1_000_0001 = 1 x8 MClk cycles 8'b1_000_0010 = 2 x8 MClk cycles ... 8'b1_111_1111 = 127 x8 MClk cycles

4.5.6.9. DDRx PHY Command and Data Block Control Register (PHYCR0, Offset = 0x20)

When both bit 15, "auto_io_deep_pdn", and bit 7, "auto_io_ctrl_pdn", are set to '1' at the same time, the I/O enable will follow the setting of bit 15, "auto_io_deep_pdn".

In order to prevent the I/O enable switch too frequently, if "auto_io_deep_pdn" or "auto_ioctrl_pdn" is set to '1':

(1) dqie, CMDADDROEN will be disabled when the power-down period is larger than 10 MClk cycles.

(2) CLKOEN0 will be disabled when the self-refresh period is larger than 10 MClk cycles.

Note: BYTE0_FLDO, BYTE2_FLDO, BYTE4_FLDO, and BYTE6_FLDO are used in the 16-bit DDR2 PHY.

Table 4-49 DDRx PHY Command and Data Block Control Register (PHYCR0, Offset = 0x20)

Bit	Name	Type	Default Value	Description
31	BYTE7_FLDO	RO	0	Byte 7 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
30	BYTE6_FLDO	RO	0	Byte 6 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
29	BYTE5_FLDO	RO	0	Byte 5 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
28	BYTE4_FLDO	RO	0	Byte 4 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
27	BYTE3_FLDO	RO	0	Byte 3 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 or FTDDR3030_MAX_MEM_WIDTH_32 is configured.
26	BYTE2_FLDO	RO	0	Byte 2 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64

Vangotech

				FTDDR3030_MAX_MEM_WIDTH_32 is configured.
25	BYTE1_FLDO	RO	0	Byte 1 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state.
24	BYTE0_FLDO	RO	0	Byte 0 DLL lock flag indicates that DLL is at the lock state. 1'b1 = DLL is at the lock state. 1'b0 = DLL is not at the lock state.
[23:18]	-	-	Rsvd	Reserved
17	phy_odd	RW	0	RE_ODD and WR_ODD signal usages for DDRx PHY 1'b1 = DDR PHY Data block is controlled by DFI_RDDATA_EN_P0_BYTEn, RE_ODD, DFI_WRDATA_EN_P0_BYTEn, and WR_ODD. 1'b0 = Data clock is controlled by DFI_RDDATA_EN_P0_BYTEn, DFI_RDDATA_EN_P1_BYTEn, DFI_WRDATA_EN_P0_BYTEn, and DFI_WRDATA_EN_P1_BYTEn. RE_ODD and WR_ODD are not used by DDRx PHY.
16	fldo_cmd	RO	0	DDR3 PHY only 1'b1 = Command block DLL is at the lock state. 1'b0 = Command block DLL is not at the lock state.
15	auto_io_deep_pdn	RW	0	Automatic control of the I/O input buffer at the power-down state 1'b1 = Enable 1'b0 = Disable When auto_io_deep_pdn is enabled, CMDADDROEN and dqie will be disabled at the automatic power-down state; and CLKOEN0, CMDADDROEN, dqie, and ODTOE will be disabled at the self-refresh state.
14	SELFBIAS	RW	0	VREF selection 1'b1 = DDR PHY uses the internal generated VREF. 1'b0 = DDR PHY uses the external VREF.
13	CONUPDATE	RW	1	DDR PHY DLL update enable control 1'b1 = FTDDR3030 will set "BYTEn_UPDATE_EN" to '1' to DDRx PHY at the initial stage and wait "BYTEn_UPDATE_ACK" = 1 to finish the DDRx PHY DLL update sequence.

Vangotech

				1'b0 = FTDDR3030 will not set "BYTEn_UPDATE_EN" to '1' to DDRx PHY at the initial stage.
12	CLKPHASESHIFT	RW	0	DDR PHY input clock for the DDR SDRAM phase 1'b0 = Clock position is at the 1/2 clock cycle phase of the DDR command. 1'b1 = Clock position is at the 3/4 clock cycle phase of the DDR command.
11	dqie	RW	1	DQ and DQS receivers enable 1'b1 = Enable 1'b0 = Disable Note: In the automatic power-down mode, the dqie bit will be controlled by hardware if auto_ioctrl_pdn or auto_io_deep_pdn is set to '1'. DDRx PHY should be reset before normal operation if dqie is set to '0'. dqie is duplicated to the BYTEn_DQIE pins for each byte (n = 0 ~ 7).
10	CLKOEN0	RW	1	The input controls DDR PHY CK and CKB. 1'b1 = Enable 1'b0 = Disable Note: In the self-refresh mode, the CLKOEN0 bit will be controlled by hardware if auto_ioctrl_pdn or auto_io_deep_pdn is set to '1'.
9	CMDADDROEN	RW	1	The input controls of the DDR PHY ADDR, BA, RAS, CAS, WE, and CS drivers 1'b1 = Enable 1'b0 = Disable Note: In the automatic power-down mode, the CMDADDROEN bit will be controlled by hardware if auto_ioctrl_pdn or auto_io_deep_pdn is set to '1'.
8	ODTOEN	RW	1	Enable input drivers of ODT 1'b1: Enable 1'b0: Disable Note: In the self-refresh mode, the ODTOE bit will be controlled by hardware if auto_io_deep_pdn is set to '1'.
7	auto_io_ctrl_pdn	RW	0	Automatic control of the I/O Input buffer at power-down state 1'b1 = Enable 1'b0 = Disable When auto_io_ctrl_pdn is enabled, cmdaddroen and dqie will be disabled at the automatic power-down

Vangotech

				state and clkoen will be disabled at the self-refresh state.
[6:4]	ODTMD_DQS [2:0]	RW	1	Set the ODT value of DDR2 PHY, only ODTMD[1:0] is used for DDR2 PHY: 3'b010: ODT = 150 Ω 3'b001: ODT = 75 Ω 3'b011: ODT = 50 Ω 3'b000: ODT is disabled. Set the ODT value of the DDR2/DDR3 Combo PHY: 3'b111: ODT = 20 Ω 3'b110: ODT = 24 Ω 3'b101: ODT = 30 Ω 3'b100: ODT = 40 Ω 3'b011: ODT = 40 Ω 3'b010: ODT = 60 Ω 3'b001: ODT = 120 Ω 3'b000: ODT is disabled.
3	phy_resetn_SRX_ enable	RW	0	Enable automatically reset D-PHY after exiting the self-refresh mode 1'b1 = Enable 1'b0 = Disable
[2:0]	ODTMD_DQ [2:0]	RW	1	Set the ODT value of DDR2 PHY, only ODTMD[1:0] is used for DDR2 PHY: 3'b010: ODT = 150 Ω 3'b001: ODT = 75 Ω 3'b011: ODT = 50 Ω 3'b000: ODT is disabled. Set the ODT value of the DDR2/DDR3 Combo PHY: 3'b111: ODT = 20 Ω 3'b110: ODT = 24 Ω 3'b101: ODT = 30 Ω 3'b100: ODT = 40 Ω 3'b011: ODT = 40 Ω 3'b010: ODT = 60 Ω 3'b001: ODT = 120 Ω 3'b000: ODT is disabled.

Figure 4-15 illustrates the DDR PHY I/O control timing based on MClk when entering into the self-refresh mode if "auto_io_deep_pdn" is set to '1'.

Vangotech

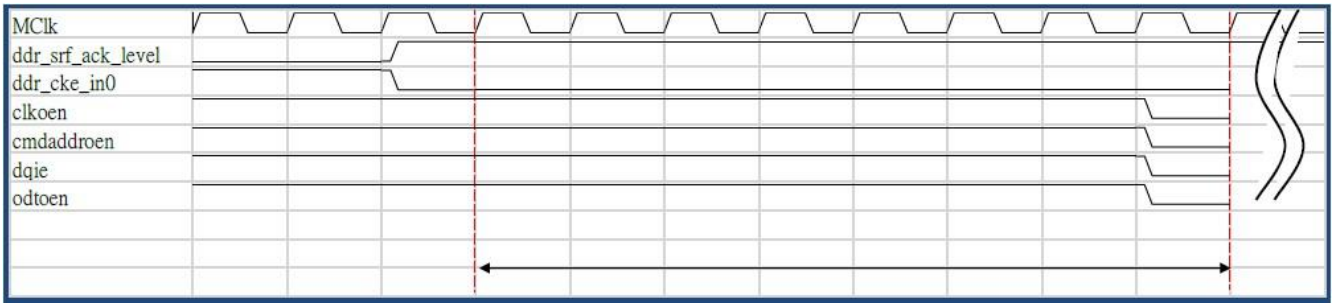


Figure 4-15. DDR PHY I/O Control Timing when Entering into Self-refresh Mode

Figure 4-16 illustrates the DDR PHY I/O control timing based on MClk when exiting from the self-refresh mode if "auto_io_deep_pdn" is set to '1'.

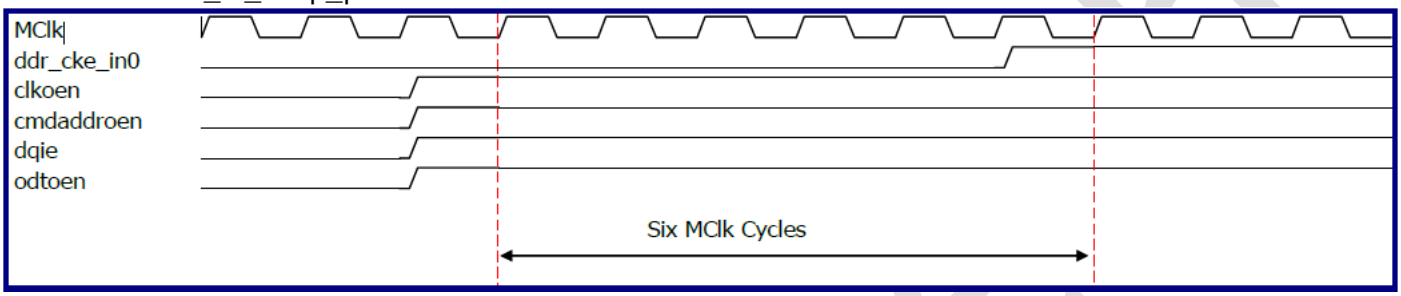


Figure 4-16. DDR PHY I/O Control Timing when Exiting from Self-refresh Mode

4.5.6.10. DDRx PHY Read Path DLL Delay Tuning Register (PHYRDTR, Offset = 0x24)

dllsel is used to adjust the read DQ and read DQS. The read DQS should be at the center of the DQ data eye for more timing margin. Please refer to the corresponding DDR PHY application note for more detailed information.

For DDR 502 series PHY, the delay direction (delay DQ or delay DQS) is indicated in bit 3 of BYTEn_DLSEL_RD. For other PHYs, please refer to Register Offset 0x6C for more details.

Note: BYTE0_DLSEL_RD, BYTE2_DLSEL_RD, BYTE4_DLSEL_RD, and BYTE6_DLSEL_RD are used in the 16-bit DDR2 PHY.

Table 4-50 DDRx PHY Read Path DLL Delay Tuning Register (PHYRDTR, Offset = 0x24)

Bit	Name	Type	Default Value	Description
[31:28]	BYTE7_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
[27:24]	BYTE6_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details.

				This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
[23:20]	BYTE5_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
[19:16]	BYTE4_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 is configured.
[15:12]	BYTE3_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 or FTDDR3030_MAX_MEM_WIDTH_32 is configured.
[11:8]	BYTE2_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details. This bit is existed when FTDDR3030_MAX_MEM_WIDTH_64 or FTDDR3030_MAX_MEM_WIDTH_32 is configured.
[7:4]	BYTE1_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details.
[3:0]	BYTE0_DLSEL_RD	RW	7	Delay value control to add a delay into the read DQS or read DQ Please refer to the Register Offset 0x6C for details.

4.5.6.11. COMPBLK Control Register (COMPBLKCR, Offset = 0x28)

Table 4-51 COMPBLK Control Register (COMPBLKCR, Offset = 0x28)

Bit	Name	Type	Default Value	Description
[31:25]	-	-	Rsvd	Reserved
[24:19]	DON	RO	0x20	Value of the COMPBLK pull-down NMOS impedance setting from the input, “DON”, of compensation block of DDRx PHY
[18:13]	DOP	RO	0x20	Value of the COMPBLK pull-up PMOS impedance setting from the input, “DOP”, of compensation block of DDRx PHY
[12:7]	DIN	RW	0x1f	Preset value for the COMPBLK pull-down NMOS impedance

[6:1]	DIP	RW	0x1f	Preset value for the COMPBLK pull-up PMOS impedance
0	COMP_SEL	RW	1	Compensation method selection 1'b1 = Enable the COMPBLK auto-calibration before initiating DDRx SDRAM 1'b0 = Use the preset values of DIP and DIN

4.5.6.12. Automatic Power-down/Self-refresh Control Register (APDCR, Offset = 0x2C)

Table 4-52 Automatic Power-down Control Register (AODCR, Offset = 0x2C)

Bit	Name	Type	Default Value	Description
[31:29]	-	-	Rsvd	Reserved
[28]	auto_srf_en	RW	0	Automatic self-refresh mode control 1 = Enable the automatic self-refresh 0 = Disable the automatic self-refresh
[27:16]	auto_srf_timer	RW	0	Automatic self-refresh timer When the command queue is empty, the automatic self-refresh timer will start counting down. When the timer reaches zero, the self-refresh request will be triggered and DDRx SDRAM will enter the self-refresh mode. 12'h000 = Not used 12'h001 = 1 x 4 = 4 MClk cycles 12'h002 = 2 x 4 = 8 MClk cycles … 12'hfff = 4095 x 4 = 16380 MClk cycles
[15:13]	-	-	Rsvd	Reserved
12	auto_pdn_en	RW	0	Automatic power-down mode control 1 = Enable the automatic power-down mode 0 = Disable the automatic power-down mode
[11:0]	auto_pdn_timer	RW	0	Automatic power-down timer When the command queue is empty, the automatic power-down timer will start counting down. When the timer reaches zero, CKE will be automatically pulled low. DDRx SDRAM will enter the active power-down mode or precharge power-down mode. 12'h000 = Not used 12'h001 = 1x4 = 4 MClk cycles 12'h002 = 2x4 = 8 MClk cycles … 12'hfff = 4095x4 = 16380 MClk cycles

Vangotech

4.5.6.13. Channel Arbitration Setup Register A (CHARBRA, Offset = 0x30)

Table 4-53 lists and describes the parameter setup for the channel arbitration. There are two arbitration modes:

1. Two-level round-robin with the channel grant count

2. Read-Write group with grant_count_group and the channel grant count

When the Read-Write group arbitration is enabled, whether the priority, burst-orient, and grant count value for the read and write operations can be identical or not will be decided by the independent read-write setting enable bit, "Indepent_RW".

Table 4-53 Setting for Read/Write Operation

Read-Write Group	**Indepent_RW Enable**	**Priority, Burst-orient**	**Grant Count for Group**	**Grant Count for Channel**
Enable	Enable	Register 0x30 defines the read setting. Register 0xC4 defines the write setting.	Register 0x30 defines the read group grant count and the read grant count constraints the read operations in the same group. Register 0xC4 defines the write group grant count and the write group grant count constraints the write operations in the same group.	Registers 0x34 and 0x38 define the read grant count for each channel and the read grant count constraints the read operations. Registers 0xC8 and 0xCC define the write grant count for each channel and the write grant count constraints the write operations.
	Disable	Register 0x30 defines both the read and write settings.	Register 0x30 defines the read and write group grant counts.	Registers 0x34 and 0x38 define the read and write grant counts for each channel.
Disable	Enable or disable	Register 0x30 defines both the read and write settings.	N/A	Registers 0x34 and 0x38 define the grant count for each channel and the grant count constraints both the read and write operations.

If Register Offset 0x30 bit[30] "Indep_RW_EN" = '1', Register Offset 0x30[28:0] will be used for the read commands. If Register Offset 0x30 bit[30] "Indep_RW_EN" = '0', Register Offset 0x30[28:0] will be used

for the read and write commands.

Table 4-54 Channel Arbitration Setup Register (CHARBRA, Offset = 0x30)

Bit	Name	Type	Default Value	Description
31	RW_Grp_EN	RW	0	Enable the Read-Write group arbitration Set this bit to '1' to enable the Read-Write group arbitration. Otherwise, the normal two-level round-robin arbitration will be used. 1'b1 = Read-Write group arbitration is enabled. 1'b0 = Read-Write group arbitration is disabled. This bit cannot be dynamically changed. Users can change this setting only when there is no DDR traffic.
30	Indep_RW_EN	RW	0	Independent R/W control 1 = Independent R/W is enabled. 0 = Independent R/W is disabled.
29	-	-	Rsvd	Reserved
[28:24]	group_grant_count_low	RW	10	Total group_grant_count[7:0] = {group_grant_count_high, group_grant_count_low} 0xC4[18:16] indicates "group_grant_count_high" and 0x30[28:24] indicates "group_grant_count_low". For the Read-Write group arbitration, this field sets the initial value of the group grant window count. This field will not be used if the Read-Write group arbitration is disabled. Total group_grant_count: 0 = A maximum of 1 command is issued when each group is granted. 1 = A maximum of 2 commands are issued when each group is granted. 2 = A maximum of 3 commands are issued when each group is granted. … 254 = A maximum of 255 commands are issued when each group is granted. 255 = Not used
[23:16]	-	-	RsvdO	Reserved and Opaque

Vangotech

[15:8]	BstOriArb	RW	0	Burst oriented arbitration enable bit (AHB only) When this bit is set, the burst transaction will not be broken during arbitration. Bit 8 = 1'b1 indicates that the burst oriented arbitration of channel 0 is enabled. Bit 8 = 1'b0 indicates that the burst oriented arbitration of channel 0 is disabled. Bit 9 = 1'b1 indicates that the burst oriented arbitration of channel 1 is enabled. Bit 9 = 1'b0 indicates that the burst oriented arbitration of channel 1 is disabled. Bit 10 = 1'b1 indicates that the burst oriented arbitration of channel 2 is enabled. Bit 10 = 1'b0 indicates that the burst oriented arbitration of channel 2 is disabled. Bit 11 = 1'b1 indicates that the burst oriented arbitration of channel 3 is enabled. Bit 11 = 1'b0 indicates that the burst oriented arbitration of channel 3 is disabled. Bit 12 = 1'b1 indicates that the burst oriented arbitration of channel 4 is enabled. Bit 12 = 1'b0 indicates that the burst oriented arbitration of channel 4 is disabled. Bit 13 = 1'b1 indicates that the burst oriented arbitration of channel 5 is enabled. Bit 13 = 1'b0 indicates that the burst-oriented arbitration of channel 5 is disabled. Bit 14 = 1'b1 indicates that the burst oriented arbitration of channel 6 is enabled. Bit 14 = 1'b0 indicates that the burst oriented arbitration of channel 6 is disabled. Bit 15 = 1'b1 indicates that the burst oriented arbitration of channel 7 is enabled. Bit 15 = 1'b0 indicates that the burst oriented arbitration of channel 7 is disabled.
[7:0]	Ch_hi_prior	RW	0	All channels are divided into two priority levels: The high-priority level and low-priority level. These bits indicate which channels belong to the high-priority level. Setting each bit to '1' indicates that the corresponding channel belongs to the high-priority level. Bit 0 = 1 indicates that channel 0 belongs to the

Vangotech

				high-priority group. Bit 0 = 0 indicates that channel 0 belongs to the low-priority group. Bit 1 = 1 indicates that channel 1 belongs to the high-priority group. Bit 1 = 0 indicates that channel 1 belongs to the low-priority group. Bit 2 = 1 indicates that channel 2 belongs to the high-priority group. Bit 2 = 0 indicates that channel 2 belongs to the low-priority group. Bit 3 = 1 indicates that channel 3 belongs to the high-priority group. Bit 3 = 0 indicates that channel 3 belongs to the low-priority group. Bit 4 = 1 indicates that channel 4 belongs to the high-priority group. Bit 4 = 0 indicates that channel 4 belongs to the low-priority group. Bit 5 = 1 indicates that channel 5 belongs to the high-priority group. Bit 5 = 0 indicates that channel 5 belongs to the low-priority group. Bit 6 = 1 indicates that channel 6 belongs to the high-priority group. Bit 6 = 0 indicates that channel 6 belongs to the low-priority group. Bit 7 = 1 indicates that channel 7 belongs to the high-priority group. Bit 7 = 0 indicates that channel 7 belongs to the low-priority group.

4.5.6.14. Channel Arbiter Grant Count Register - A (CHGNTRA, Offset = 0x34)

Each channel has specific grant window counter. When the arbiter performs re-arbitration, the value set in this register will be loaded into the grant window counters of channel 0, channel 1, channel 2, and channel 3. Once the granted channel pops a command to the memory controller, the grant window counter of this channel will be decreased by one. If the burst-oriented function is disabled and the grant window counter of the current granted channel becomes '0', the arbiter will switch the grant to another channel. If the burst-oriented function is enabled, the arbiter will switch grant when the grant window counter becomes '0' and the popped command will be the EOB (End of burst) command of a burst transaction, which is that the burst transaction will not be broken by the arbiter even when the grant window counter becomes '0'.

Vangotech

The grant window counter of each channel defines the allowed maximum service command counts for each channel when that channel is granted by the arbiter. Please note that a command indicates a DDR2 WRAP4 R/W command or DDR3 WRAP8 R/W command.

If Register Offset 0x30 bit[31] "RW_Group_En" = '1' and bit[30] "Indep_RW_EN" = '1', Register Offset 0x34 will be used for the read commands; otherwise, Register offset 0x34 will be used for the read and write commands.

Table 4-55 Channel Arbiter Grant Count Register - A (CHGNTRA, Offset = 0x34)

Bit	Name	Type	Default Value	Description
[31:24]	ARB_CNT3	RW	0xff	Once channel 3 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[23:16]	ARB_CNT2	RW	0xff	Once channel 2 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[15:8]	ARB_CNT1	RW	0xff	Once channel 1 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[7:0]	ARB_CNT0	RW	0xff	Once channel 0 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands

Vangotech

4.5.6.15. Channel Arbiter Grant Count Register - B (CHGNTRB, Offset = 0x38)

Similar to Register Offset = 0x34, this register defines the maximum service command counts of channel 4, channel 5, channel 6, and channel 7. If Register Offset 0x30 bit[31] "RW_Group_En" = '1' and bit[30] "Indep_RW_EN" = '1', Register Offset 0x38 will be used for the read commands; otherwise, Register Offset 0x38 will be used for the read and write commands.

Table 4-56 Channel Arbiter Grant Count Register - B (CHGNTRB, Offset = 0x38)

Bit	Name	Type	Default Value	Description
[31:24]	ARB_CNT7	RW	0xff	Once channel 7 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[23:16]	ARB_CNT6	RW	0xff	Once channel 6 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[15:8]	ARB_CNT5	RW	0xff	Once channel 5 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands
[7:0]	ARB_CNT4	RW	0xff	Once channel 4 is granted, the maximum allowed commands will be issued. 0 = 1 command 1 = 2 commands 2 = 3 commands … 255 = 256 commands

Vangotech

4.5.6.16. DDRx PHY Write/Read Data Timing Control Register (PHYWRTMR, Offset = 0x3C)

tphy_wrlat specifies the number of MClk clock cycles between a write command sent on the PHY control interface and the asserted dfi_wrdata_en signal. tphy_wrdata specifies the number of MClk clock cycles between the asserted dfi_wrdata_en signal and the associated write data driven on the dfi_wrdata signal.

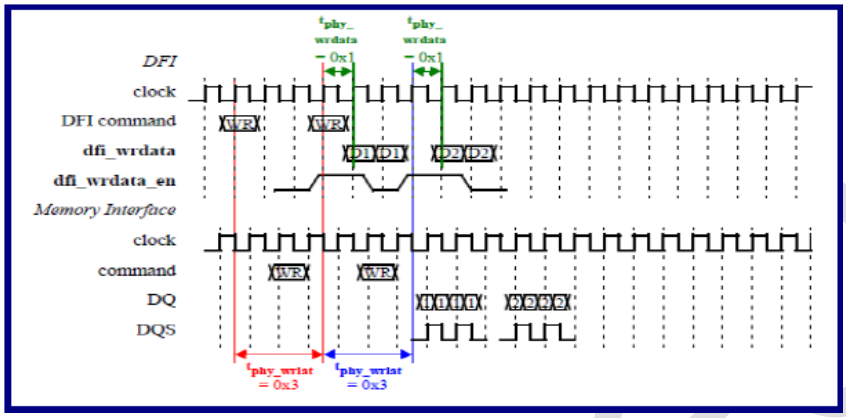


Figure 4-17. tphy_wrlat and tphy_wrdata Timing in DFI Specification 2.1

The dfi_rddata_en signal must be asserted for trddata_en cycles after the assertion of a read command on the PHY control interface and remains valid for the duration of the contiguous read data expected on the dfi_rddata bus.

Some legacy DDR PHY does not have "dfi_rddata_valid" signal. So, the read data are expected to be received at FTDDR3030 within tphy_rdlat cycles after the dfi_rddata_en signal is asserted. If DDR PHY has "dfi_rddata_valid" signal, users can set "tphy_rdlat" to zero, and the controller will use DDR PHY output, "dfi_rddata_valid", to latch read data.

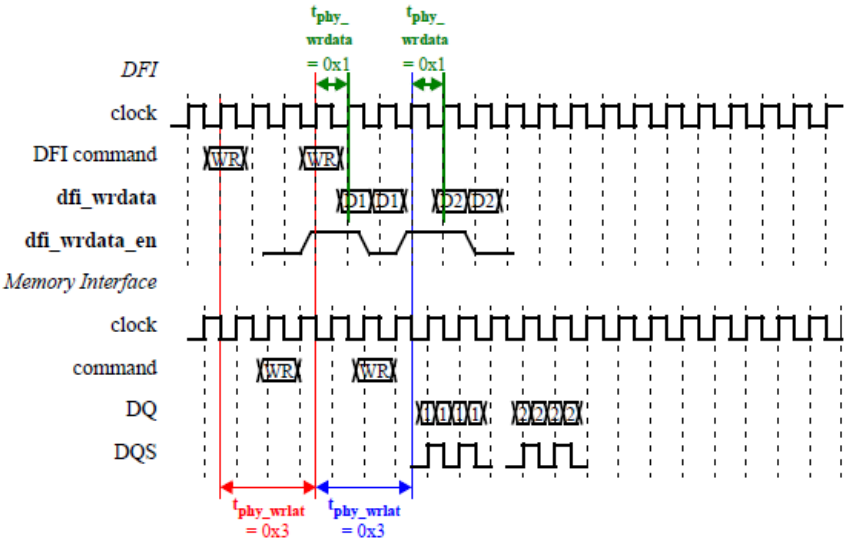


Figure 4-18. trddata_en and tphy_rdlat in DFI Specification 2.1

Table 4-57 DDRx PHY Write/Read Data Timing Control Register (PHYWRTMR, Offset = 0x3C)

Bit	Name	Type	Default Value	Description
[31:24]	-	-	Rsvd	Reserved
[23:20]	tphy_rdlat	RW	0	EFIFO effects tphy_rdlat. Please set tphy_rdlat as '0', when DDR PHY has dfi_rddata_valid signal. The read data are expected to be received at FTDDR3030 within **tphy_rdlat** cycles after the **rdcmd** signal is asserted. 0 = Use DDR PHY output signal, "dfi_rddata_valid", to latch read data 1 = Not used 2 = dfi_rddata is expected to receive 2 MClk cycles after rdcmd is asserted. ... 15 = dfi_rddata is expected to receive 15 MClk cycles after rdcmd is asserted.
[19:16]	trddata_en	RW	0	dfi_rddata_en signal must be asserted for trddata_en cycles after the assertion of a read command on the PHY control interface and remains valid for the duration of contiguous read data expected on the dfi_rddata bus. The valid values are between 0 and 10. 0 = rdcmd asserts at the same cycle as the read command asserts. 1 = rdcmd asserts 1 MClk cycle after the read command asserts. 2 = rdcmd asserts 2 MClk cycles after the read command asserts. … 10 = rdcmd asserts 10 MClk cycles after the read command asserts. Others = Reserved
[15:6]	-	-	Rsvd	Reserved
[5:4]	tphy_wrdata	RW	0	tphy_wrdata specifies the number of MClk clock cycles between when the dfi_wrdata_en signal is asserted and when the associated write data are driven on the dfi_wrdata signal. 0 = dfi_wrdata will be asserted at the same time when dfi_wrdata_en is asserted. 1 = dfi_wrdata will assert one MClk cycle after dfi_wrdata_en is asserted. 2 = dfi_wrdata will assert two MClk cycles after dfi_wrdata_en is asserted. 3 = dfi_wrdata will assert three MClk cycles after

Vangotech

				dfi_wrdata_en is asserted.
[3:0]	tphy_wrlat	RW	0	tphy_wrlat specifies the number of MClk clock cycles between when a write command is sent on the DFI control interface and when the dfi_wrdata_en signal is asserted. The valid values are between 0 and 9. 0 = dfi_wrdata_en is asserted at the same time with the write command is asserted. 1 = dfi_wrdata_en will assert 1 MClk cycle after the write command is asserted. 2 = dfi_wrdata_en will assert 2 MClk cycles after the write command is asserted. … 10 = dfi_wrdata_en will assert 10 MClk cycles after the write command is asserted. Others = Reserved

t_{WL} = Write latency in terms of the DRAM clock cycles

t_{RL} = Read latency in terms of the DRAM clock cycles

FTDDR3030 only supports AL = '0'.

For DDR1, t_{WL} = 1

For DDR2, t_{WL} = AL (Additive Latency) + CL (CAS Latency) - 1

For DDR3, t_{WL} = AL (Additive Latency) + CWL (CAS Write Latency)

For DDR1, t_{RL} = CL (CAS Latency)

For DDR2, DDR3, t_{RL} = AL (Additive Latency) + CL (CAS Latency)

(1) For **FXDDRII 17x DDR2 PHY** series without "**dfi_rddata_valid**":

Table 4-58 FXDDRII17x DDR PHY Timing Parameter

FXDDRII 17x DDR2 PHY (without "dfi_rddata_valid")	
Parameter	**Setting**
tphy_wrlat	t_{WL} - 1
tphty_wrdata	1
trddata_en	t_{RL}
tphy_rdlat	GDS + 2, GDS + 3, GDS + 4 (Need to be scanned and confirmed by pre-sim and

	post-sim)

For example,

GDS = 1, t_{WL} = 4, t_{RL} = 5, tphy_wrlat = 4'h3, tphy_wrdata = 2'h1, trddata_en = 4'h5,

tphy_rdlat = 4'h3 ~ 4'h5

GDS = 2, t_{WL} = 4, t_{RL} = 5, tphy_wrlat = 4'h3, tphy_wrdata = 2'h1, trddata_en = 4'h5,

tphy_rdlat = 4'h4 ~ 4'h6

GDS = 3, t_{WL} = 4, t_{RL} = 5, tphy_wrlat = 4'h3, tphy_wrdata = 2'h1, trddata_en = 4'h5,

tphy_rdlat = 4'h5 ~ 4'h7

(2) For FXDDR2A/D 200 DDR12 Combo PHY without "dfi_rddata_valid"

Table 4-59 FXDDR2A/D 200 DDR12 Combo PHY Timing Parameter

FXDDR2A/D 200 DDR12 PHY (without "dfi_rddata_valid")		
Parameter	**0x00[18] byone = 1**	**0x00[18] byone = 0**
tphy_wrlat	tWL - 1	tWL = odd, ((tWL – 1)/2) - 1
		tWL = even, (tWL/2) - 1
tphty_wrdata	1	1
trddata_en	tRL - 2	tRL = odd, ((tRL – 1)/2) - 1
		tRL = even, (tRL/2) - 1
tphy_rdlat	GDS + 5, GDS + 6, GDS + 7 (Need to be scanned and confirmed by pre-sim and post-sim)	GDS + 3, GDS + 4, GDS + 5 (Need to be scanned and confirmed by pre-sim and post-sim)

(3) For FXDDR10x, 30x, 40x, 50x DDR23/LPDDR2 Combo PHY with "dfi_rddata_valid"

Table 4-60 FXDDR10x, 30x, 40x, 50x DDR23/LPDDR2 Combo PHY Timing Parameter

FXDDRA/D 10x, 30x, 40x, 50x series PHY (with "dfi_rddata_valid")		
Parameter	**0x00[18] byone = 1**	**0x00[18] byone = 0**
tphy_wrlat	tWL - 1	tWL = odd, ((tWL – 1)/2) - 1
		tWL = even, (tWL/2) - 1
tphty_wrdata	1	1
trddata_en	tRL - 2	tRL = odd, ((tRL – 1)/2) - 1
		tRL = even, (tRL/2) - 1
tphy_rdlat	0	0

For example,

BYONE = 0

GDS = 0, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 1, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 2, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 3, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 4, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 5, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 6, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

GDS = 7, t_{WL} = 7, t_{RL} = 9, tphy_wrlat = 4'h2, tphy_wrdata = 2'h1, trddata_en = 4'h3, tphy_rdlat = 0

4.5.6.17. Command Flush Control Register (FLUSHCR, Offset = 0x40)

Please note that bit[16] is existed only when the hardware configuration,

`define FTDDR3030_DLOGIC_YES, is configured.

Table 4-61 Command Flush Control Register (FLUSHCR, Offset = 0x40)

Bit	Name	Type	Default Value	Description
[31:17]	-	-	Rsvd	Reserved
16	debug_int_en	RW	0	Interrupt enable for the debug function
[15:8]	flush_int_en	RW	0	Interrupt enable for the command flushing When the command flushing is completed, FTDDR3030 will trigger an interrupt if the corresponding bit of this field is set to '1'. Bit 8 for channel 0 Bit 9 for channel 1 Bit 10 for channel 2 Bit 11 for channel 3 Bit 12 for channel 4 Bit 13 for channel 5 Bit 14 for channel 6 Bit 15 for channel 7

Vangotech

[7:0]	cmd_flush_en	RW	0	Enable of command flushing Set the corresponding bit of this field to '1' to start the command flushing Bit 0 for channel 0 Bit 1 for channel 1 Bit 2 for channel 2 Bit 3 for channel 3 Bit 4 for channel 4 Bit 5 for channel 5 Bit 6 for channel 6 Bit 7 for channel 7 When the command flushing is completed or there are no commands in the command queue of the AHB channel, the corresponding bit of this field will be automatically cleared to '0' by FTDDR3030. Please note that the AXI interface does not support the command flushing, writing '1' of the corresponding AXI channel will not be automatically cleared to '0'.

4.5.6.18. Command Flush Status Register (FLUSHSR, Offset = 0x44)

Please note that bit[8] is existed only when the hardware configuration,

`define FTDDR3030_DLOGIC_YES, is configured.

Table 4-62 Command Flush Status Register (FLUSHSR, Offset = 0x44)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:9]	-	-	Rsvd	Reserved
8	debug_hit	RW1C	0	Status of the debug function
[7:0]	flush_done	RW1C	0	Status of the command flushing This field records the status of the command flushing for each channel. When a channel terminates the command flushing, the corresponding bit of this field will be set to '1' by FTDDR3030. Bit 0 for channel 0 Bit 1 for channel 1 Bit 2 for channel 2 Bit 3 for channel 3 Bit 4 for channel 4 Bit 5 for channel 5 Bit 6 for channel 6

				Bit 7 for channel 7

4.5.6.19. AHB SPLIT Control Register (SPLITCR, Offset = 0x48)

If the AHB channels are configured with the supporting SPLIT response, the SPLIT function can be disabled by setting the corresponding bit in this register.

Table 4-63 AHB SPLIT Control Register (SPLITCR, Offset = 0x48)

Bit	Name	Type	Default Value	Description
[31:24]	-	-	Rsvd	Reserved
[23:22]	CH7_hprot_sel	RW	0	2'b01 = Always set channel 7 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 7 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[21:20]	CH6_hprot_sel	RW	0	2'b01 = Always set channel 6 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 6 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[19:18]	CH5_hprot_sel	RW	0	2'b01 = Always set channel 5 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 5 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[17:16]	CH4_hprot_sel	RW	0	2'b01 = Always set channel 4 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 4 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[15:14]	CH3_hprot_sel	RW	0	2'b01 = Always set channel 3 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 3 to non-bufferable regardless of the value in the HPROT input port

Vangotech

				Others = Depending on the value in the HPROT input port
[13:12]	CH2_hprot_sel	RW	0	2'b01 = Always set channel 2 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 2 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[11:10]	CH1_hprot_sel	RW	0	2'b01 = Always set channel 1 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 1 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
[9:8]	CH0_hprot_sel	RW	0	2'b01 = Always set channel 0 to bufferable regardless of the value in the HPROT input port 2'b10 = Always set channel 0 to non-bufferable regardless of the value in the HPROT input port Others = Depending on the value in the HPROT input port
7	CH7_split_disable	RW	0	This bit works only when channel 7 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 7
6	CH6_split_disable	RW	0	This bit works only when channel 6 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 6
5	CH5_split_disable	RW	0	This bit works only when channel 5 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 5
4	CH4_split_disable	RW	0	This bit works only when channel 4 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 4
3	CH3_split_disable	RW	0	This bit works only when channel 3 is enabled and

Vangotech

				configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 3
2	CH2_split_disable	RW	0	This bit works only when channel 2 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 2
1	CH1_split_disable	RW	0	This bit works only when channel 1 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 1
0	CH0_split_disable	RW	0	This bit works only when channel 0 is enabled and configured as an AHB channel with the SPLIT response. Set this bit to '1' to disable the SPLIT response of channel 0

4.5.6.20. DDRx PHY and DDR3 Update Control Register (UPDCR, Offset = 0x4C)

This register is for the DDR3 PHY DLL update and write-leveling calibration register.

Table 4-64 DDRx PHY and DDR3 Update Control Register (UPDCR, Offset = 0x4C)

Bit	Name	Type	Default Value	Description
[31:18]	-	-	Rsvd	Reserved
[17:16]	ZQCS and ZQCL update	RW	0	2'b01 = Automatically enable the ZQCS command after exiting from the self-refresh state 2'b10 = Automatically enable the ZQCL command after exiting from the self-refresh state Others = Not used and no ZQCS or ZQCL command after exiting from the self-refresh state If ZQCS or ZQCL update is enabled, the ZQCS or ZQCL command will be issued after tXS (tRFC+10 ns) is satisfied.
[15:8]	tWLEVEL_UP	RW	0	Write-leveling calibration timing interval 0 = Disable the write-leveling calibration interval 1 = 1x1024 MClk cycles 2 = 2x1024 MClk cycles

Vangotech

				… 255 = 255x1024 MClk cycles Note: This value should be reasonably set; otherwise, it may impact the performance.
[7:0]	tDLL_UP	RW	0	Digital DLL update delay control code timing interval 0 = Disable the digital DLL update 1 = 1x1024 MClk cycles 2 = 2x1024 MClk cycles … 255 = 255x1024 MClk cycles Note: This value should be reasonably set; otherwise, it may impact the performance.

4.5.6.21. Revision Register (REVR, Offset = 0x50)

This register shows the current revision of FTDDR3030. Table 2-59lists and describes the format of the revision register.

Table 4-65 Revision Register (REVR, Offset = 0x50)

Bit	Name	Type	Default Value	Description
[31:24]	-	-	Rsvd	Reserved and read as zero
[23:16]	MAJOR_VER	RO	0x02	This field shows the major version number of the DDRx memory controller.
[15:8]	MINOR_VER	RO	0x26	This field shows the minor version number of the DDRx memory controller.
[7:0]	REL_VER	RO	0x00	This field shows the release number of the current revision.

4.5.6.22. Feature Register 1 (FEATR1, Offset = 0x54)

This register shows the current feature of FTDDR3030. Table 4-66 lists and describes the format of the feature register for the global part of this IP.

Table 4-66 Feature Register 1 (FEATR1, Offset = 0x54)

Bit	Name	Type	Default Value	Description
[31:26]	-	-	Rsvd	Reserved
25	Bank swizzle	RO	1	1: Bank swizzle is configured. 0: Bank swizzle is not configured.
24	Bank Number	RO	1	1: 8 banks per rank are configured

Vangotech

				0: 4 banks per rank are configured.
[23:20]	-	-	Rsvd	Reserved
19	Channel-3 data width	RO	1	1: 64-bit data width is configured for channel 3. 0: 32-bit data width is configured for channel 3.
18	Channel-2 data width	RO	1	1: 64-bit data width is configured for channel 2. 0: 32-bit data width is configured for channel 2.
17	Channel-1 data width	RO	1	1: 64-bit data width is configured for channel 1. 0: 32-bit data width is configured for channel 1.
16	Channel-0 data width	RO	0	1: 64-bit data width is configured for channel 0. 0: 32-bit data width is configured for channel 0.
[15:12]	-	-	Rsvd	Reserved
11	Channel-3 type	RO	1	1: AXI interface is configured for channel 3. 0: AHB interface is configured for channel 3.
10	Channel-2 type	RO	1	1: AXI interface is configured for channel 2. 0: AHB interface is configured for channel 2.
9	Channel-1 type	RO	1	1: AXI interface is configured for channel 1. 0: AHB interface is configured for channel 1.
8	Channel-0 type	RO	1	1: AXI interface is configured for channel 0. 0: AHB interface is configured for channel 0.
[7:6]	Supported the maximum memory width	RO	3	3: The maximum 64-bit memory width is configured. 2: The maximum 32-bit memory width is configured. 01: The maximum 16-bit memory width is configured. 00: The maximum 8-bit memory width is configured.
[5:3]	Atomic monitor	RO	2	3’h2 = Atomic monitor has two entries.
[2:0]	Channel counts	RO	3	4’h3 = 4 channels are configured.

Vangotech

4.5.6.23. Feature Register 2 (FEATR2, Offset = 0x58)

This register shows the current feature of FTDDR3030. **Table 4-67** lists and describes the feature register for the AHB part of this IP.

Table 4-67 Feature Register 2 (FEATR2, Offset = 0x58)

Bit	Name	Type	Default Value	Description
[31:16]	-	-	-	Reserved
15	Channel-3 bus connect type	RO	0	1: Channel 3 connected to bus is configured. 0: Channel 3 connected to master is configured.
14	Channel-3 split type	RO	0	1: Channel 3 split feature is configured. 0: Channel 3 split feature is not configured.
[13:12]	Channel-3 clock mode	RO	1	1: Channel 3 asynchronous mode is configured. 0: Channel 3 synchronous mode is configured.
11	Channel-2 bus connect type	RO	0	1: Channel 2 connected to bus is configured. 0: Channel 2 connected to master is configured.
10	Channel-2 split type	RO	0	1: Channel 2 split feature is configured. 0: Channel 2 split feature is not configured.
[9:8]	Channel-2 clock mode	RO	1	1: Channel 2 asynchronous mode is configured. 0: Channel 2 synchronous mode is configured.
7	Channel-1 bus connect type	RO	0	1: Channel 1 connected to bus is configured. 0: Channel 1 connected to master is configured.
6	Channel-1 split type	RO	0	1: Channel 1 split feature is configured. 0: Channel 1 split feature is not configured.
[5:4]	Channel-1 clock mode	RO	1	1: Channel 1 asynchronous mode is configured. 0: Channel 1 synchronous mode is configured.
3	Channel-0 bus connect type	RO	0	1: Channel 0 connected to bus is configured. 0: Channel 0 connected to master is

Vangotech

				configured.
2	Channel-0 split type	RO	0	1: Channel 0 split feature is configured. 0: Channel 0 split feature is not configured.
[1:0]	Channel-0 clock mode	RO	1	1: Channel 0 asynchronous mode is configured. 0: Channel 0 synchronous mode is configured.

4.5.6.24. User Defined Register (UDEFR, Offset = 0x5C)

Table 4-68 User Defined Register (UDEFR, Offset = 0x5C)

Bit	Name	Type	Default Value	Description
[31:0]	User Defined Register	RW	0x0000000	User-defined register for specific usage The reset value can be hardware-configured in the hardware configuration file. The value of the register will decide the output value of the 32-bit interface, "USER_DEF_REG".

4.5.6.25. Write-leveling Control Register (WLEVELCR, Offset = 0x60) (Only for DDR3 Combo PHY)

FTDDR3030 supports one rank (Chip select0) for the hardware write-leveling calibration. FTDDR3030 supports one rank DIMM or two on–board ranks. If two on-board ranks are used, the length of the PCB trace and the layout of the on-board rank1 DDR3 SDRAM should physically and symmetrically be the same as rank0.

The write-leveling control register is used to adjust the skew between command path and the DQ/DQS path by using the Fly-By DDR3 SDRAM configuration. Users can use the hardware scan by setting each byte of wlevel_byten_hw_disable to 1'b0 (Only for DDR3 mode) or trace the skew between Command/CK DQS path in whole ASIC and DDR path to fill the PDL_set registers (Offset 0x64 and Offset 0x68). tWLO should be set to half value of the DRAM timing parameters according to the "MClk:DRAM clock = 1:2" mode.

Table 4-69 Write-leveling Control Register (WLEVELCR, Offset = 0x60) (Only for DDR3 Combo PHY)

Bit	Name	Type	Default Value	Description
31	wlevel_byte7_pass	RO	0	Byte 7 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
30	wlevel_byte6_pass	RO	0	Byte 6 of the write-leveling status register 1'b1 = Write-leveling passed.

				1'b0 = Write-leveling failed or write-leveling hardware disabled.
29	wlevel_byte5_pass	RO	0	Byte 5 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
28	wlevel_byte4_pass	RO	0	Byte 4 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
27	wlevel_byte3_pass	RO	0	Byte 3 of the write-leveling status register 1'b1 = Write-leveling passed 1'b0 = Write-leveling failed or write-leveling hardware disabled
26	wlevel_byte2_pass	RO	0	Byte 2 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
25	wlevel_byte1_pass	RO	0	Byte 1 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
24	wlevel_byte0_pass	RO	0	Byte 0 of the write-leveling status register 1'b1 = Write-leveling passed. 1'b0 = Write-leveling failed or write-leveling hardware disabled.
23	wlevel_byte7_hw_disable	RW	1	Byte 7 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
22	wlevel_byte6_hw_disable	RW	1	Byte 6 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
21	wlevel_byte5_hw_disable	RW	1	Byte 5 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
20	wlevel_byte4_hw_disable	RW	1	Byte 4 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled)

Vangotech

				1'b0 = Write-leveling by hardware
19	wlevel_byte3_hw_disable	RW	1	Byte 3 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
18	wlevel_byte2_hw_disable	RW	1	Byte 2 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
17	wlevel_byte1_hw_disable	RW	1	Byte 1 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
16	wlevel_byte0_hw_disable	RW	1	Byte 0 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware
[15:8]	-	-	Rsvd	Reserved
[7:0]	tWLO	RW	8	Write-leveling DQS enable to latch the DQ data delay time 8'h1 = 1 MClk cycle … 8'hff = 255 MClk cycles Recommended value = 10 MClk cycles

4.5.6.26. Write-leveling Byte 7 ~ Byte 4 Control Register (WLEVELBHR, Offset = 0x64) (Only for DDR3 Combo PHY)

Software programmable skew adjustment is between Command/CK and DQS path. This register can adjust the multiphase DLL of DDR23 Combo PHY to compensate the skew between Fly-By Command/CK and DQS. Please refer to Table 2-64 for more details. The value of BYTEn_PDLSET reflects the write-leveling result if Register Offset 0x60, wlevel_byten_hw_disable, is set to '0' (Only for DDR3 mode). The value of BYTEn_PDLSET is the software programmed value if Register Offset 0x60, wlevel_byten_hw_disable, is set to '1'.

Table 4-70 Write-leveling Byte 7 ~ Byte 4 Control Register (WLEVELBHR, Offset = 0x64) (Only for DDR3 Combo PHY)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:24]	BYTE7_PDLSET	RW	0	Programmable DLL delay for byte 7 wlevel_byte7_hw_disable = '0': Hardware tuning

				value wlevel_byte7_hw_disable = '1': Software programmed value
23	-	-	Rsvd	Reserved
[22:16]	BYTE6_PDLSET	RW	0	Programmable DLL delay for byte 6 wlevel_byte6_hw_disable = '0': Hardware tuning value wlevel_byte6_hw_disable = '1': Software programmed value
15	-	-	Rsvd	Reserved
[14:8]	BYTE5_PDLSET	RW	0	Programmable DLL delay for byte 5 wlevel_byte5_hw_disable = '0': Hardware tuning value wlevel_byte5_hw_disable = '1': Software programmed value
7	-	-	Rsvd	Reserved
[6:0]	BYTE4_PDLSET	RW	0	Programmable DLL delay for byte 4 wlevel_byte4_hw_disable = '0': Hardware tuning value wlevel_byte4_hw_disable = '1': Software programmed value

4.5.6.27. Write-leveling Byte 3 ~ Byte 0 Control Register (WLEVELBLR, Offset = 0x68) (Only for DDR3 Combo PHY)

Software programmable skew adjustment is between Command/CK and DQS path. This register can adjust the multiphase DLL of DDR23 Combo PHY to compensate the skew between Fly-By Command/CK and DQS. The value of BYTEn_PDLSET will reflect the write-leveling result if Register Offset 0x60, wlevel_byten_hw_disable, is set to '0' (Only for DDR3 mode). The value of BYTEn_PDLSET is the software programmed value if Register Offset 0x60, wlevel_byten_hw_disable, is set to '1'.

Table 4-71 Write-leveling Byte 3 ~ Byte 0 Control Register (WLEVELBLR, Offset = 0x68) (Only for DDR3 Combo PHY)

Bit	**Name**	**Type**	**Default Value**	**Description**
31	-	-	Rsvd	Reserved
[30:24]	*BYTE3_PDLSET*	RW	0	Programmable DLL delay for byte 3 wlevel_byte3_hw_disable = '0': Hardware tuning value wlevel_byte3_hw_disable = '1': Software programmed value
23	-	-	Rsvd	Reserved

[22:16]	*BYTE2_PDLSET*	RW	0	Programmable DLL delay for byte 2 wlevel_byte2_hw_disable = '0': Hardware tuning value wlevel_byte2_hw_disable = '1': Software programmed value
15	-	-	Rsvd	Reserved
[14:8]	*BYTE1_PDLSET*	RW	0	Programmable DLL delay for byte 1 wlevel_byte1_hw_disable = '0': Hardware tuning value wlevel_byte1_hw_disable = '1': Software programmed value
7	-	-	Rsvd	Reserved
[6:0]	*BYTE0_PDLSET*	RW	0	Programmable DLL delay for byte0 wlevel_byte0_hw_disable = '0': Hardware tuning value wlevel_byte0_hw_disable = '1': Software programmed value

4.5.6.28. DDR3 PHY MISC Control Register 1 (PHYMISCR1, Offset = 0x6C)

For DDR 502 series PHY, this register is not required, the delay direction (Delay DQ or delay DQS) is indicated in bit[3] of BYTEn_DLSEL_RD and BYTEn_DLSEL_WR.

Table 4-72 DDR3 PHY MISC Control Register 1 (PHYMISCR1, Offset = 0x6C)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:30]	-	-	Rsvd	Reserved
29	dqs_sel_wr_byte7	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE7_DLSEL_WR* (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
28	dqs_sel_byte7	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE7_DLSEL_RD* (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[27:26]	-	-	Rsvd	Reserved
25	dqs_sel_wr_byte6	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE6_DLSEL_WR* (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
24	dqs_sel_byte6	RW	0	Add the delay in DQ or DQS, the delay value is

Vangotech

				controlled by *BYTE6_DLSEL_RD* (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[23:22]	-	-	Rsvd	Reserved
21	dqs_sel_wr_byte5	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE5_DLSEL_WR* (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
20	dqs_sel_byte5	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE5_DLSEL_RD* (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[19:18]	-	-	Rsvd	Reserved
17	dqs_sel_wr_byte4	RW	0	Add the delay in DQ or DQS, the delay value is controlled by *BYTE4_DLSEL_WR* (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
16	dqs_sel_byte4	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE4_DLSEL_RD (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[15:14]	-	-	Rsvd	Reserved
13	dqs_sel_wr_byte3	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE3_DLSEL_WR (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
12	dqs_sel_byte3	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE3_DLSEL_RD (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[11:10]	-	-	Rsvd	Reserved
9	dqs_sel_wr_byte2	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE2_DLSEL_WR (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
8	dqs_sel_byte2	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE2_DLSEL_RD (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[7:6]	-	-	Rsvd	Reserved

Vangotech

5	dqs_sel_wr_byte1	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE1_DLSEL_WR (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
4	dqs_sel_byte1	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE1_DLSEL_RD (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.
[3:2]	-	-	Rsvd	Reserved
1	dqs_sel_wr_byte0	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE0_DLSEL_WR (Offset 0x78). 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ.
0	dqs_sel_byte0	RW	0	Add the delay in DQ or DQS, the delay value is controlled by BYTE0_DLSEL_RD (Offset 0x24). 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ.

4.5.6.29. Read-leveling Control Register (RLEVELCR, Offset = 0x70) (Only for DDR3 Combo PHY)

The read-leveling control register adjusts the read DQS filtering window. Users can use the hardware scan by setting each byte of rlevel_sel_byten_hw_disable to 1'b0 (Only for DDR3/LPDDR2 mode) or use the software scan to fill the msdly register (Offset = 0x74).

Table 4-73 Read-leveling Control Register (Offset = 0x70) (RLEVELCR, Only for DDR3 Combo PHY)

Bit	Name	Type	Default Value	Description
31	rlevel_byte7_pass	RO	0	Read-leveling status for byte 7 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
30	rlevel_byte6_pass	RO	0	Read-leveling status for byte 6 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
29	rlevel_byte5_pass	RO	0	Read-leveling status for byte 5 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
28	rlevel_byte4_pass	RO	0	Read-leveling status for byte 4 1'b1 = Read-leveling passed.

				1'b0 = Read-leveling failed or read-leveling hardware disabled.
27	rlevel_byte3_pass	RO	0	Read-leveling status for byte 3 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
26	rlevel_byte2_pass	RO	0	Read-leveling status for byte 2 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
25	rlevel_byte1_pass	RO	0	Read-leveling status for byte 1 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
24	rlevel_byte0_pass	RO	0	Read-leveling status for byte 0 1'b1 = Read-leveling passed. 1'b0 = Read-leveling failed or read-leveling hardware disabled.
[23:17]	-	-	Rsvd	Reserved
16	periodic_train_en	R/W	0	1 = Periodic read leveling enable (Only for LPDDR2 mode) The read leveling will be performed every "rlevel_train_period" timing interval. 0 = Disable periodic read leveling
[15:8]	rlevel_train_period	R/W	0	Periodic read-leveling training timing interval 0 = Not used if periodic_train_en = 1 1 = 1 x16 MClk cycles … 255 = 255 x16 MClk cycles For example, DDR3 1600 case, MClk = 2.5 ns Set 255 = the read leveling will be performed around every 10.2 us.
7	rlevel_byte7_hw_disable	RW	1	Read-leveling hardware disable for byte 7 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
6	rlevel_byte6_hw_disable	RW	1	Read-leveling hardware disable for byte 6 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
5	rlevel_byte5_hw_disable	RW	1	Read-leveling hardware disable for byte 5

Vangotech

				1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
4	rlevel_byte4_hw_disable	RW	1	Read-leveling hardware disable for byte 4 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
3	rlevel_byte3_hw_disable	RW	1	Read-leveling hardware disable for byte 3 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
2	rlevel_byte2_hw_disable	RW	1	Read-leveling hardware disable for byte 2 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
1	rlevel_byte1_hw_disable	RW	1	Read-leveling hardware disable for byte 1 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware
0	rlevel_byte0_hw_disable	RW	1	Read-leveling hardware disable for byte 0 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware

4.5.6.30. msdly Byte Control Register (MSDLYCR, Offset = 0x74)

The read DQS control register is treated as a clock in the read path. This register is used to adjust the read DQS filtering window at DDR PHY to filter out the unwanted pulses or glitches of read DQS. The value of BYTEn_MSDLY will reflect the read-leveling result if Register Offset 0x70 rlevel_byten_hw_disable is set to '0' (Only for DDR3/LPDDR2 mode). The value of BYTEn_MSDLY will use the software programmed value if Register Offset 0x70 rlevel_byten_hw_disable is set to '1'.

Table 4-74 msdly Byte Control Register (MSDLYCR, Offset = 0x74)

Bit	Name	Type	Default Value	Description
[31:28]	*BYTE7_MSDLY*	RW	1	Programmable read DQS filtering window for byte 7 rlevel_byte7_hw_disable = '0', hardware tuning value rlevel_byte7_hw_disable = '1', software programmed value
[27:24]	*BYTE6_MSDLY*	RW	1	Programmable read DQS filtering window for byte 6 rlevel_byte6_hw_disable = '0', hardware tuning value rlevel_byte6_hw_disable = '1', software programmed

				value
[23:20]	*BYTE5_MSDLY*	RW	1	Programmable read DQS filtering window for byte 5 rlevel_byte5_hw_disable = '0', hardware tuning value rlevel_byte5_hw_disable = '1', software programmed value
[19:16]	*BYTE4_MSDLY*	RW	1	Programmable read DQS filtering window for byte 4 rlevel_byte4_hw_disable = '0', hardware tuning value rlevel_byte4_hw_disable = '1', software programmed value
[15:12]	*BYTE3_MSDLY*	RW	1	Programmable read DQS filtering window for byte 3 rlevel_byte3_hw_disable = '0', hardware tuning value rlevel_byte3_hw_disable = '1', software programmed value
[11:8]	*BYTE2_MSDLY*	RW	1	Programmable read DQS filtering window for byte 2 rlevel_byte2_hw_disable = '0', hardware tuning value rlevel_byte2_hw_disable = '1', software programmed value
[7:4]	*BYTE1_MSDLY*	RW	1	Programmable read DQS filtering window for byte 1 rlevel_byte1_hw_disable = '0', hardware tuning value rlevel_byte1_hw_disable = '1', software programmed value
[3:0]	*BYTE0_MSDLY*	RW	1	Programmable read DQS filtering window for byte 0 rlevel_byte0_hw_disable = '0', hardware tuning value rlevel_byte0_hw_disable = '1', software programmed value

4.5.6.31. wrdllsel Control Register (WRDLLCR, Offset = 0x78)

For DDR 502 series PHY, the delay direction (Delay DQ or delay DQS) is indicated in bit 3 of BYTEn_DLSEL_WR. For other versions of PHY, please refer to Register Offset 0x6C for more details.

Table 4-75 wrdllsel Control Register (WRDLLCR, Offset = 0x78)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:28]	*BYTE7_DLSEL_WR*	RW	8	Delay control for byte 7 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[27:24]	*BYTE6_DLSEL_WR*	RW	8	Delay control for byte 6 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[23:20]	*BYTE5_DLSEL_WR*	RW	8	Delay control for byte 5 to add a delay in DQS or DQ

Vangotech

				Please refer to Offset 0x6C for details.
[19:16]	*BYTE4_DLSEL_WR*	RW	8	Delay control for byte 4 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[15:12]	*BYTE3_DLSEL_WR*	RW	8	Delay control for byte 3 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[11:8]	*BYTE2_DLSEL_WR*	RW	8	Delay control for byte 2 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[7:4]	*BYTE1_DLSEL_WR*	RW	8	Delay control for byte 1 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.
[3:0]	*BYTE0_DLSEL_WR*	RW	8	Delay control for byte 0 to add a delay in DQS or DQ Please refer to Offset 0x6C for details.

4.5.6.32. Traffic Monitor Clock Cycle Register (TRAFMR, Offset = 0x7C)

Table 4-76 Traffic Monitor Clock Cycle Register (TRAFMR, Offset = 0x7C)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:0]	TM_clk_cycle_reg	RW	0	This register works only when the traffic monitor is enabled. The value of this register will be decreased by 1 for every clock cycle of MClk. When the value of this register is not zero, the traffic monitor will count the issued commands for each channel.

4.5.6.33. Command Count Register for Channel 0 (CMDCNTR0, Offset = 0x80)

Table 4-77 Command Count Register for Channel 0 (CMDCNTR0, Offset = 0x80)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:0]	cmd_cnt_reg0	RO	0	This register works only when the traffic monitor and channel 0 are enabled. This register records the issued commands from channel 0, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.34. Command Count Register for Channel 1 (CMDCNTR1, Offset = 0x84)

Table 4-78 Command Count Register for Channel 1 (CMDCNTR1, Offset = 0x84)

Vangotech

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg1	RO	0	This register works only when the traffic monitor and channel 1 are enabled. This register records the issued commands from channel 1, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.35. Command Count Register for Channel 2 (CMDCNTR2, Offset = 0x88)

Table 4-79 Command Count Register for Channel 2 (CMDCNTR2, Offset = 0x88)

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg2	RO	0	This register works only when the traffic monitor and channel 2 are enabled. This register records the issued commands from channel 2, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.36. Command Count Register for Channel 3 (CMDCNTR3, Offset = 0x8C)

Table 4-80 Command Count Register for Channel 3 (CMDCNTR3, Offset = 0x8C)

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg3	RO	0	This register works only when the traffic monitor and channel 3 are enabled. This register records the issued commands from channel 3, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.37. Command Count Register for Channel 4 (CMDCNTR4, Offset = 0x90)

Table 4-81 Command Count Register for Channel 4 (CMDCNTR4, Offset = 0x90)

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg4	RO	0	This register works only when the traffic monitor and channel 4 are enabled. This register records the issued commands from channel 4, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.38. Command Count Register for Channel 5 (CMDCNTR5, Offset = 0x94)

Table 4-82 Command Count Register for Channel 5 (CMDCNTR5, Offset = 0x94)

Vangotech

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg5	RO	0	This register works only when the traffic monitor and channel 5 are enabled. This register records the issued commands from channel 5, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.39. Command Count Register for Channel 6 (CMDCNTR6, Offset = 0x98)

Table 4-83 Command Count Register for Channel 6 (CMDCNTR6, Offset = 0x98)

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg6	RO	0	This register works only when the traffic monitor and channel 6 are enabled. This register records the issued commands from channel 6, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.40. Command Count Register for Channel 7 (CMDCNTR7, Offset = 0x9C)

Table 4-84 Command Count Register for Channel 7 (CMDCNTR7, Offset = 0x9C)

Bit	Name	Type	Default Value	Description
[31:0]	cmd_cnt_reg7	RO	0	This register works only when the traffic monitor and channel 7 are enabled. This register records the issued commands from channel 7, and will be automatically cleared when the register, TM_clk_cycle_reg, is written by users.

4.5.6.41. AHB INCR Read Prefetch Length 1 (AHBRPRER1, Offset = 0xA0)

FTDDR3030 provides two mechanisms to prefetch the read data for the AHB unspecified-length (INCR) read bursts: Limited prefetching and unlimited prefetching.

For limited prefetching, users should define a value used as "total number" of the read commands. FTDDR3030 will generate the DDR read commands when receiving an AHB INCR read burst. For example, if the user-defined value is N, FTDDR3030 will generate N+1 DDR read commands to prefetch the read data for any AHB INCR burst.

For unlimited prefetching, users should define a value used as "speed factor" of the read commands. In this mode, FTDDR3030 will keep generating the read commands until it receives an AHB early termination condition. The user-defined value determines the maximum number of the prefetch read commands that are on-going in pipeline. For example, if the user-defined value is N, FTDDR3030 will first generate N+1

Vangotech

DDR read commands and then will suspend the command generation until the on-going command count reaches N+1. When the data belonged to the first read command arrives the read FIFO, FTDDR3030 will generate another read command if the on-going command count is less than N+1. This mechanism will be repeatedly performed until the AHB early termination occurs.

Table 4-85 AHB INCR Read Prefetch Length 1 (AHBRPRER1, Offset = 0xA0)

Bit	Name	Type	Default Value	Description
31	CH3_limited_pref	RW	0	Prefetch mechanism of channel 3 1: Limited prefetching 0: Unlimited prefetching
[30:29]	-	RW	RsvdO	Reserved and Opaque
[28:24]	CH3_pref_value	RW	3	User-defined value of channel 3 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
23	CH2_limited_pref	RW	0	Prefetch mechanism of channel 2 1: Limited prefetching 0: Unlimited prefetching
[22:21]	-	RW	RsvdO	Reserved and Opaque
[20:16]	CH2_pref_value	RW	3	User-defined value of channel 2 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
15	CH1_limited_pref	RW	0	Prefetch mechanism of channel 1 1: Limited prefetching 0: Unlimited prefetching
[14:13]	-	RW	RsvdO	Reserved and Opaque
[12:8]	CH1_pref_value	RW	3	User-defined value of channel 1 for the INCR read prefetching 0: 1 command 1: 2 commands

Vangotech

				2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
7	CH0_limited_pref	RW	0	Prefetch mechanism of channel 0 1: Limited prefetching 0: Unlimited prefetching
[6:5]	-	RW	RsvdO	Reserved and Opaque
[4:0]	CH0_pref_value	RW	3	User-defined value of channel 0 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.

4.5.6.42. AHB INCR Read Prefetch Length 2 (AHBRPRER2, Offset = 0xA4)

Table 4-86 AHB INCR Read Prefetch Length 2 (AHBRPRER2, Offset = 0xA4)

Bit	Name	Type	Default Value	Description
31	CH7_limited_pref	RW	0	Prefetch mechanism of channel 7 1: Limited prefetching 0: Unlimited prefetching
[30:29]	-	RW	RsvdO	Reserved and Opaque
[28:24]	CH7_pref_value	RW	3	User-defined value of channel 7 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
23	CH6_limited_pref	RW	0	Prefetch mechanism of channel 6 1: Limited prefetching 0: Unlimited prefetching
[22:21]	-	RW	RsvdO	Reserved and Opaque
[20:16]	CH6_pref_value	RW	3	User-defined value of channel 6 for the INCR read

Vangotech

				prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
15	CH5_limited_pref	RW	0	Prefetch mechanism of channel 5 1: Limited prefetching 0: Unlimited prefetching
[14:13]	-	RW	RsvdO	Reserved and Opaque
[12:8]	CH5_pref_value	RW	3	User-defined value of channel 5 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.
7	CH4_limited_pref	RW	0	Prefetch mechanism of channel 4 1: Limited prefetching 0: Unlimited prefetching
[6:5]	-	-	RsvdO	Reserved and Opaque
[4:0]	CH4_pref_value	RW	3	User-defined value of channel 4 for the INCR read prefetching 0: 1 command 1: 2 commands 2: 3 commands … 31: 32 commands Note: 0 and 31 are prohibited in the IP version before 2.11.0.

4.5.6.43. Initialization of Waiting Cycle Count 1 (INITWCR1, Offset = 0xA8)

Table 4-87 Initialization of Waiting Cycle Count 1 (INITWCR1, Offset = 0xA8)

Bit	Name	Type	Default Value	Description
[31:20]	-	-	Rsvd	Reserved
[19:0]	wait_cycle_200us	RW	0x3E80	After powering up the ramp of SDRAM, a waiting time of

				200 μs is required before the RESET signal becomes inactive. This register defines the waiting clock cycles of FTDDR3030 to satisfy the requirement of 200 μs.

4.5.6.44. Initialization of Waiting Cycle Count 2 (INITWCR2, Offset = 0xAC)

Table 4-88 Initialization of Waiting Cycle Count 2 (INITWCR2, Offset = 0xAC)

Bit	Name	Type	Default Value	Description
[31:20]	-	-	Rsvd	Reserved
[19:0]	wait_cycle_500us	RW	00061a80	For DDR3 SDRAM, a waiting time of 500 μs is required after the RESET signal becomes inactive and before the CKE signal becomes active. This register defines the waiting clock cycles of FTDDR3030 to satisfy the requirement of 500 μs.

4.5.6.45. QoS Control Register (QOSCR, Offset = 0xB0)

Table 4-89 Setting for QoS Command Count

Read-Write Group	QoS Indepent_RW Enable	QoS Command Count
Enable	Enable	Registers 0xB4 and 0xB8 define the QoS command count for the read operation, and the command count constraints the read operations. Registers 0xBC and 0xC0 define the QoS command count for the write operation, and the command count constraints the write operations.
	Disable	Registers 0xB4 and 0xB8 define the QoS command count for the read operation, and the command count constraints the read and write operations.
Disable	Enable or Disable	Registers 0xB4 and 0xB8 define the QoS command count, and the command count constraints the read and write operations.

Table 4-90 QoS Control Register (QOSCR, Offset = 0xB0)

Bit	Name	Type	Default Value	Description
[31:4]	-	-	Rsvd	Reserved
3	QoS_Indep_RW_en	RW	0	QoS independent R/W enable If this bit is set to '1', the setting of QoS_CmdCnt in 0xB4 and 0xB8 will be the limitation only for the read command. If this bit is set to '0', the setting of QoS_CmdCnt in 0xB4 and 0xB8 will be the limitation for both the read

Vangotech

				and write commands.
[2:1]	QoS_PeriodMode	RW	0	QoS period mode 2'b00: 512 clock cycles 2'b01: 1024 clock cycles 2'b10: 2048 clock cycles 2'b11: 4096 clock cycles
0	QoS_En	RW	0	QoS enable

4.5.6.46. QoS Command Count Register A (QOSCNTRA, Offset = 0xB4)

If QoS_Indep_RW_en is set to '1', Register Offset 0xB4 will be used for the read command. If QoS_Indep_RW_en is set to '0', Register Offset 0xB4 will be used for both the read and write commands.

Table 4-91 QoS Command Count Register A (QOSCNTRA, Offset = 0xB4)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:24]	QoS_Ch3CmdCnt	RW	0	Command count limitation for channel 3 8'h00: No limit 8'h01: 1x8 commands 8'h02: 2x8 commands ... 8'hff: 255x8 commands
[23:16]	QoS_Ch2CmdCnt	RW	0	Command count limitation for channel 2 8'h00: No limit 8'h01: 1x8 commands 8'h02: 2x8 commands ... 8'hff: 255x8 commands
[15:8]	QoS_Ch1CmdCnt	RW	0	Command count limitation for channel 1 8'h00: No limit 8'h01: 1x8 commands 8'h02: 2x8 commands ... 8'hff: 255x8 commands
[7:0]	QoS_Ch0CmdCnt	RW	0	Command count limitation for channel 0 8'h00: No limit 8'h01: 1x8 commands 8'h02: 2x8 commands ... 8'hff: 255x8 commands

Vangotech

4.5.6.47. QoS Command Count Register B (QOSCNTRB, Offset = 0xB8)

If QoS_Indep_RW_en is set to '1', Register Offset 0xB8 will be used for the read command. If QoS_Indep_RW_en is set to '0', Register Offset 0xB8 will be used for both the read and write commands.

Table 4-92 QoS Command Count Register B (QOSCNTRB, Offset = 0xB8)

Bit	Name	Type	Default Value	Description
[31:24]	QoS_Ch7CmdCnt	RW	0	Read command count limitation for channel 7 8'h00: No limit 8'h01: 1x8 read commands 8'h02: 2x8 read commands ... 8'hff: 255x8 read commands
[23:16]	QoS_Ch6CmdCnt	RW	0	Read command count limitation for channel 6 8'h00: No limit 8'h01: 1x8 read commands 8'h02: 2x8 read commands ... 8'hff: 255x8 read commands
[15:8]	QoS_Ch5CmdCnt	RW	0	Read command count limitation for channel 5 8'h00: No limit 8'h01: 1x8 read commands 8'h02: 2x8 read commands ... 8'hff: 255x8 read commands
[7:0]	QoS_Ch4CmdCnt	RW	0	Read command count limitation for channel 4 8'h00: No limit 8'h01: 1x8 read commands 8'h02: 2x8 read commands ... 8'hff: 255x8 read commands

4.5.6.48. QoS Command Count Register C (QOSCNTRC, Offset = 0xBC)

If QoS_Indep_RW_en is set to '1', Register Offset 0xBC will be used for the write command. If QoS_Indep_RW_en is set to '0', Register Offset 0xBC will be meaningless.

Table 4-93 QoS Command Count Register C (QOSCNTRC, Offset = 0xBC)

Bit	Name	Type	Default Value	Description
[31:24]	QoS_Ch3CmdCnt_w	RW	0	Write command count limitation for channel 3 8'h00: No limit 8'h01: 1x8 write commands

Vangotech

				8'h02: 2x8 write commands ... 8'hff: 255x8 write commands
[23:16]	QoS_Ch2CmdCnt_w	RW	0	Write command count limitation for channel 2 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands
[15:8]	QoS_Ch1CmdCnt_w	RW	0	Write command count limitation for channel 1 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 wrote commands ... 8'hff: 255x8 write commands
[7:0]	QoS_Ch0CmdCnt_w	RW	0	Write command count limitation for channel 0 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands

4.5.6.49. QoS Command Count Register D (QOSCNTRD, Offset = 0xC0)

If QoS_Indep_RW_en is set to '1', Register Offset 0xC0 will be used for the write command. If QoS_Indep_RW_en is set to '0', Register Offset 0xC0 will be meaningless.

Table 4-94 QoS Command Count Register D (QOSCNTRD, Offset = 0xC0)

Bit	Name	Type	Default Value	Description
[31:24]	QoS_Ch7CmdCnt_w	RW	0	Write Command Count Limitation for Channel 7 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands
[23:16]	QoS_Ch6CmdCnt_w	RW	0	Write Command Count Limitation for Channel 6 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands

Vangotech

[15:8]	QoS_Ch5CmdCnt_w	RW	0	Write Command Count Limitation for Channel 5 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands
[7:0]	QoS_Ch4CmdCnt_w	RW	0	Write Command Count Limitation for Channel 4 8'h00: No limit 8'h01: 1x8 write commands 8'h02: 2x8 write commands ... 8'hff: 255x8 write commands

4.5.6.50. Channel Arbitration Setup Register B (CHARBRB, Offset = 0xC4)

If Register Offset 0x30 bit[31] "RW_Group_En" = '1' and bit[30] "Indep_RW_EN" = '1', Register Offset 0xC4 will be used for the write command; otherwise, Register Offset 0xC4 will be meaningless.

Table 4-95 Channel Arbitration Setup Register B (CHARBRB, Offset = 0xC4)

Bit	Name	Type	Default Value	Description
[31:24]	group_grant_count_w	RW	0x10	This field will not be used if the Read-Write group arbitration is disabled. 0 = Amaximum of 1 command is issued when the write group is granted. 1 = Amaximum of 2 commands are issued when the write group is granted. 2 = Amaximum of 3 commands are issued when the write group is granted. … 254 = A maximum of 255 commands are issued when the write group is granted. 255 = Not used
[23:19]	-	-	Rsvd	Reserved
[18:16]	group_grant_count_high	RW	0	Please refer to the descriptions at Register Offset 0x30. Total group_grant_count[7:0] = {group_grant_count_high, group_grant_count_low} 0xC4[18:16] indicates "group_grant_count_high". 0x30[28:24] indicates

Vangotech

				"group_grant_count_low".
[15:8]	BstOriArb_w	RW	0	Burst oriented arbitration enable bit (AHB only) When these bits are set, the burst transaction will not be broken during arbitration. Bit 8 = 1'b1 indicates that the burst oriented arbitration of channel 0 is enabled for the write commands. Bit 8 = 1'b0 indicates that the burst oriented arbitration of channel 0 is disabled for the write commands. Bit 9 = 1'b1 indicates that the burst oriented arbitration of channel 1 is enabled for the write commands. Bit 9 = 1'b0 indicates that the burst oriented arbitration of channel 1 is disabled for the write commands. Bit 10 = 1'b1 indicates that the burst oriented arbitration of channel 2 is enabled for the write commands. Bit 10 = 1'b0 indicates that the burst oriented arbitration of channel 2 is disabled for the write commands. Bit 11 = 1'b1 indicates that the burst oriented arbitration of channel 3 is enabled for the write commands. Bit 11 = 1'b0 indicates that the burst oriented arbitration of channel 3 is disabled for the write commands. Bit 12 = 1'b1 indicates that the burst oriented arbitration of channel 4 is enabled for the write commands. Bit 12 = 1'b0 indicates that the burst oriented arbitration of channel 4 is disabled for the write commands. Bit 13 = 1'b1 indicates that the burst oriented arbitration of channel 5 is enabled for the write commands. Bit 13 = 1'b0 indicates that the burst oriented arbitration of channel 5 is disabled for the write commands. Bit 14 = 1'b1 indicates that the burst oriented

Vangotech

				arbitration of channel 6 is enabled for the write commands. Bit 14 = 1'b0 indicates that the burst oriented arbitration of channel 6 is disabled for the write commands. Bit 15 = 1'b1 indicates that the burst oriented arbitration of channel 7 is enabled for the write commands. Bit 15 = 1'b0 indicates that the burst oriented arbitration of channel 7 is disabled for the write commands.
[7:0]	Ch_hi_prior_w	RW	0	All channels are divided into two priority levels: High-priority level and low-priority level. These bits indicate which channels belong to the high-priority level. Setting each bit to '1'b1' indicates that the corresponding channels belong to the high-priority level. Bit 0 = 1'b1 indicates that the write commands of channel 0 belong to the high-priority group. Bit 0 = 1'b0 indicates that the write commands of channel 0 belong to the low-priority group. Bit 1 = 1'b1 indicates that the write commands of channel 1 belong to the high-priority group. Bit 1 = 1'b0 indicates that the write commands of channel 1 belong to the low-priority group. Bit 2 = 1'b1 indicates that the write commands of channel 2 belong to the high-priority group. Bit 2 = 1'b0 indicates that the write commands of channel 2 belong to the low-priority group. Bit 3 = 1'b1 indicates that the write commands of channel 3 belong to the high-priority group. Bit 3 = 1'b0 indicates that the write commands of channel 3 belong to the low-priority group. Bit 4 = 1'b1 indicates that the write commands of channel 4 belong to the high-priority group. Bit 4 = 1'b0 indicates that the write commands of channel 4 belong to the low-priority group. Bit 5 = 1'b1 indicates that the write commands of channel 5 belong to the high-priority group. Bit 5 = 1'b0 indicates that the write commands of channel 5 belong to the low-priority group. Bit 6 = 1'b1 indicates that the write commands of

Vangotech

				channel 6 belong to the high-priority group. Bit 6 = 1'b0 indicates that the write commands of channel 6 belong to the low-priority group. Bit 7 = 1'b1 indicates that the write commands of channel 7 belong to the high-priority group. Bit 7 = 1'b0 indicates that the write commands of channel 7 belong to the low-priority group.

4.5.6.51. Channel Arbiter Grant Count Register - C (CHGNTRC, Offset = 0xC8)

Each channel has a grant window counter. When the arbiter performs the re-arbitration, the value set in this register will be loaded into the grant window counters for channel 0, channel 1, channel 2, and channel 3. Once the granted channel pops a command to the memory controller, the grant window counter of this channel will be decreased by one. If the burst oriented function is disabled and the grant window counter of the current granted channel becomes '0', the arbiter will switch the grant to another channel. If the burst oriented function is enabled, the arbiter will switch grant when the grant window counter becomes '0' and the popped command will be the EOB (End Of Burst) command of a burst transaction that will not be broken by the arbiter even when the grant window counter becomes '0'. The grant window counter of each channel defines the maximum service command count allowed for each channel when that channel is granted by the arbiter. Please note that a command indicates a DDR2 WRAP4 R/W command or DDR3 WRAP8 R/W command.

If Register Offset 0x30 bit[31] "RW_Group_En" is '1' and bit[30] "Indep_RW_EN" is '1', Register Offset 0xC8 will be used for the write command. Otherwise, Register Offset 0xC8 will be meaningless.

Table 4-96 Channel Arbiter Grant Count Register - C (CHGNTRC, Offset = 0xC8)

Bit	Name	Type	Default Value	Description
[31:24]	ARB_CNT3_w	RW	0xff	Once channel 3 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands
[23:16]	ARB_CNT2_w	RW	0xff	Once channel 2 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands

[15:8]	ARB_CNT1_w	RW	0xff	Once channel 1 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands
[7:0]	ARB_CNT0_w	RW	0xff	Once channel 0 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands

4.5.6.52. Channel Arbiter Grant Count Register - D (CHGNTRD, Offset = 0xCC)

Similar to Register Offset = 0x34, this register defines the maximum service command counts for channel 4, channel 5, channel 6, and channel 7. If Register Offset 0x30 bit[30] "Indep_RW_EN" = '1', Register Offset 0xCC will be used for the write commands. If Register Offset 0x30 bit[30] "Indep_RW_EN" = '0', Register Offset 0xCC will be meaningless.

Table 4-97 Channel Arbiter Grant Count Register - D (CHGNTRD, Offset = 0xCC)

Bit	**Name**	**Type**	**Default Value**	**Description**
[31:24]	ARB_CNT7_w	RW	0xff	Once channel 7 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands
[23:16]	ARB_CNT6_w	RW	0xff	Once channel 6 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands
[15:8]	ARB_CNT5_w	RW	0xff	Once channel 5 is granted, the maximum allowed write commands will be issued. 0 = 1 write command

Vangotech

				1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands
[7:0]	ARB_CNT4_w	RW	0xff	Once channel 4 is granted, the maximum allowed write commands will be issued. 0 = 1 write command 1 = 2 write commands 2 = 3 write commands … 255 = 256 write commands

4.5.6.53. Debug Address (DBGADR, Offset = 0xD0)

Please note that Register Offset 0xD0 ~ 0xFC exist when `define FTDDR3030_DLOGIC_YES is configured and this feature is not supported in this IP version.

Table 4-98 Debug Address (DBGADR, Offset = 0xD0)

Bit	Name	Type	Default Value	Description
[31:0]	Base_addr	RW*	0	The register stores the base address to be monitored.

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.54. Debug Address Mask (DBGADMR, Offset = 0xD4)

Table 4-99 Debug Address Mask (DBGADMR, Offset = 0xD4)

Bit	Name	Type	Default Value	Description
[31:0]	Addr_mask	RW*	0	The register controls the mask of the base address. Addr_mask[n] = 1, enable address bit n comparison for the trigger detection Addr_mask[n] = 0, disable address bit n comparison for the trigger detection Masked address should be consecutive and starts from LSB. For example, 0xffff_ff0f and 0x0000_ffff are forbidden. 0xffff_ff00 is legal.

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.55. Debug Write Data (DBGWDR, Offset = 0xD8)

Table 4-100 Debug Write Data (DBGWDR, Offset = 0xD8)

Bit	Name	Type	Default Value	Description
[31:0]	Write_data	RW*	0	This register stores data for comparing with the write data of access.

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.56. Debug Write Data Mask (DBGWDMR, Offset = 0xDC)

Table 4-101 Debug Write Data Mask (DBGWDMR, Offset = 0xDC)

Bit	Name	Type	Default Value	Description
[31:0]	Data_mask	RW*	0	This register controls the mask of write data. Data_mask[n] = 1, enable data bit n comparison for the trigger detection Data_mask[n] = 0, disable data bit n comparison for the trigger detection

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.57. Debug Master Control (DBGMSTCR, Offset = 0xE0)

Table 4-102 Debug Master Control (DBGMSTR, Offset = 0xE0)

Bit	Name	Type	Default Value	Description
[31:20]	-	-	Rsvd	Reserved
19	Mst_extype	RW*	0	Exclusive type Mst_extype = '1' Access from the monitored master will trigger the interrupt. Mst_extype = '0' Access from any other master will trigger the interrupt.
[18:16]	Mst_ch	RW*	0	Monitored master channel number
[15:0]	Mst_id	RW*	0	Monitored master ID number

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.58. Debug Access Control (DBGACCR, Offset = 0xE4)

Table 4-103 Debug Access Control (DBGACCR, Offset = 0xE4)

Vangotech

Bit	Name	Type	Default Value	Description
[31:2]	-	-	Rsvd	Reserved
1	lsw_sel	RW*	0	Last Significant Word selection (Used only for double-word access) lsw_sel = '1' The content of the debug write data register (Offset = 0xD8) will be compared with LSW of the current write access [31:0]. lsw_sel = '0' The content of the debug write data register (Offset = 0xD8) will be compared with MSW of the current write access [63:32].
0	Rw_sel	RW*	0	Read/Write selection 0: Monitor only the read access 1: Monitor only the write access

* When the Debug Control register (Offset = 0xEC) is enabled, the write operation to this register will be invalid.

4.5.6.59. Debug Policy Control (DBGPCR, Offset = 0xE8)

Please note that it is not reasonable to simultaneously enable "read access" and "write data". If the "read access" policy is enabled, the "write data" policy will be "don't care".

Table 4-104 Debug Policy Control (DBGPCR, Offset = 0xE8)

Bit	Name	Type	Default Value	Description
[31:5]	-	-	Rsvd	Reserved
4	Rw_en	RW*	0	Read/Write access policy enable
3	Mst_ch_en	RW*	0	Master channel policy enable
2	Mst_id_en	RW*	0	Master ID policy enable
1	Wdata_en	RW*	0	Write data policy enable
0	Add_en	RW*	0	Address policy enable

* When the Debug Control register (Offset = 0xEC) is enabled, write operation to this register is invalid.

4.5.6.60. Debug Control (DBGENR, Offset = 0xEC)

Table 4-105 Debug Control (DBGENR, Offset = 0xEC)

Bit	Name	Type	Default Value	Description
[31:1]	-	-	Rsvd	Reserved
0	Debug_en	RW*	0	Debug_enable

* The enable bit automatically becomes low after interrupt.

Vangotech

4.5.6.61. Debug Address Status (DBGADSR, Offset = 0xF0)

Table 4-106 Debug Address Status (DBGADSR, Offset = 0xF0)

Bit	Name	Type	Default Value	Description
[31:0]	Addr_status	RO	0	This register records the address of an access.

4.5.6.62. Debug Write Data Status (DBGWDSR, Offset = 0xF4)

Table 4-107 Debug Write Data Status (DBGWDSR, Offset = 0xF4)

Bit	Name	Type	Default Value	Description
[31:0]	Data_status	RO	0	This register records the write data of an access.

4.5.6.63. Debug Master Status (DBGMSTSR, Offset = 0xF8)

Table 4-108 Debug Master Status (DBGMSTSR, Offset = 0xF8)

Bit	Name	Type	Default Value	Description
[31:19]	-	-	Rsvd	Reserved
[18:16]	Mst_ch_status	RO	0	The field records the master channel number of an access.
[15:0]	Mst_id_status	RO	0	The field records the master ID number of an access.

4.5.6.64. Debug Access Status (DBGACCSR, Offset = 0xFC)

Table 4-109 Debug Access Status (DBGACCSR, Offset = 0xFC)

Bit	Name	Type	Default Value	Description
[31:1]	-	-	Rsvd	Reserved
0	Ac_stutus	RO	0	The field records the access type of an access.

4.5.6.65. LPDDR2 Mode Register MR1/MR2/MR3 (LP2MRA, Offset = 0x100)

Please note that Register Offset 0x100 ~ 0x124 exist when `define FTDDR3030_LPDDR2_YES is configured.

Note: The MR1 register should be set to a sequential wrap burst length of 8 in the LPDDR2 mode.

Table 4-110 LPDDR2 Mode Register MR1/MR2/MR3 (LP2MRA, Offset = 0x100)

Bit	Name	Type	Default Value	Description
[31:24]	-	-	Rsvd	Reserved
[23:16]	MR3	RW	0	I/O configure 1 LPDDR2 Mode Register MR3
[15:8]	MR2	RW	0	Device feature 2 LPDDR2 Mode Register MR2
[7:0]	MR1	RW	0	Device feature 1

Vangotech

				LPDDR2 Mode Register MR1

4.5.6.66. LPDDR2 Mode Register MR9/MR10 (LP2MRB, Offset = 0x104)

Table 4-111 LPDDR2 Mode Register MR9/MR10 (LP2MRB, Offset = 0x104)

Bit	Name	Type	Default Value	Description
[31:16]	-	-	Rsvd	Reserved
[15:8]	MR10	RW	0	I/O calibration LPDDR2 Mode Register MR10
[7:0]	MR9	RW	0	Test mode LPDDR2 Mode Register MR9

4.5.6.67. LPDDR2 Mode Register MR16/MR17 (LP2MRC, Offset = 0x108)

Table 4-112 LPDDR2 Mode Register MR16/MR17 (LP2MRC, Offset = 0x108)

Bit	Name	Type	Default Value	Description
[31:16]	-	-	Rsvd	Reserved
[15:8]	MR17	RW	0	PASR_Seg LPDDR2 Mode Register MR17
[7:0]	MR16	RW	0	PASR_Bank LPDDR2 Mode Register MR16

4.5.6.68. LPDDR2 Mode Register MR41/MR42/MR48 (LP2MRD, Offset = 0x10C)

This register is reserved for future usage.

Table 4-113 LPDDR2 Mode Register MR41/MR42/MR48 (LP2MRD, Offset = 0x10C)

Bit	Name	Type	Default Value	Description
[31:24]	-	-	Rsvd	Reserved
[23:16]	MR48	RW	0	Reserved
[15:8]	MR42	RW	0	Reserved
[7:0]	MR41	RW	0	Reserved

4.5.6.69. LPDDR2 Mode Register MR63 (LP2MRE, Offset = 0x110)

Table 4-114 LPDDR2 Mode Register MR63 (LP2MRE, Offset = 0x110)

Bit	Name	Type	Default Value	Description
[31:8]	-	-	Rsvd	Reserved
[7:0]	MR63	RW	0	Reset LPDDR2 Mode Register MR63

Vangotech

4.5.6.70. LPDDR2 MRW and MRR Command Register (LP2MRCR, Offset = 0x114)

Table 4-115 LPDDR2 MRW and MRR Command Register (LP2MRCR, Offset = 0x114)

Bit	Name	Type	Default Value	Description
[31:26]	-	-	Rsvd	Reserved
[25:24]	LP2_MRR_BYTE_sel	RW	0	If ASIC to DDR is not byte swapped, set 0 to select DQ[7:0] as the MRR read value input, otherwise, the setting is based on the connection of ASIC to DDR 0 = Latch MRR command read value from byte0 of read data 1 = Latch MRR command read value from byte1 of read data 2 = Latch MRR command read value from byte2 of read data, this setting is valid if FTDDR3030_MAX_MEM_WIDTH_32 or FTDDR3030_MAX_MEM_WIDTH_64 is configured. 3 = Latch MRR command read value from byte3 of read data, this setting is valid if FTDDR3030_MAX_MEM_WIDTH_32 or FTDDR3030_MAX_MEM_WIDTH_64 is configured
[23:17]	-	-	Rsvd	Reserved
16	LP2_Rank_sel	RW	0	Target rank for MRW/MRR 0 = Issue MRW/MRR command to rank0 1 = Issue MRW/MRR command to rank1
[15:8]	LP2_MR_sel	RW	0	Target Mode Register for MRW/MRR 8'h0 = MR0 8'h1 = MR1 … 8'hff = MR255
[7:2]	-	-	Rsvd	Reserved
[1:0]	LP2_MR_cmd	RW	0	LP Mode Register command 2'b01 = MRW command 2'b10 = MRR command for LPDDR2 Others = Reserved

If users want to issue a MRW or MRR command to LPDDR2 SDRAM:

- Users should first fill the value of the LPDDR2 mode register from Register Offset 0x100 to Register Offset 0x110.

Vangotech

● Users can set LP2_MR_cmd and LP2_MR_sel to issue a MRW or MRR command to the mode register which LP2_MR_sel indicates.

For example, Read MR0 of Rank0 DDR, and the DDR byte0 is not byte swapped.

APB Write register offset 0x114 with value 0x00000002 (Issue MRR command)

APB Read register offset 0x118

● If a MRR command is issued, the read value will be put in Register Offset 0x118. Users should check only when Register 0x118 [31] "MRR_Valid" = 1, the MRR read data is valid corresponding to the latest MRR command.

4.5.6.71. LPDDR2 MRR Command Read Value Register (LP2MRVR, Offset = 0x118)

Please note that this register is read only and cannot be written.

Table 4-116 LPDDR2 MRR Read Value Register (LP2MRVR, Offset = 0x118)

Bit	Name	Type	Default Value	Description
31	MRR_valid	RO	0	1 = The MRR_value is valid corresponding to the latest MRR command. 0 = The MRR_value is not valid
[30:8]	-	-	Rsvd	Reserved
[7:0]	MRR_value	RO	0	Mode register read value

4.5.6.72. LPDDR2 Added Latency Register (LP2ADLR, Offset = 0x11C)

Table 4-117 LPDDR2 Write Latency and Read Latency Register (LP2ADLR, Offset = 0x11C)

Bit	Name	Type	Default Value	Description
[31:2]	-	-	Rsvd	Reserved
1	LP2_RL_normal	RW	0	LPDDR2 read latency is normal. 1 = Read latency is the value set by mode register MR2. 0 = Read latency is the value set by mode register MR2 and adds extra one cycle.
0	LP2_WL_normal	RW	0	LPDDR2 write latency is normal. 1 = Write latency is the value set by mode register MR2. 0 = Write latency is the value set by mode register MR2 and adds extra one cycle.

When users set LPDDR2 SDRAM write latency and read latency in LPDDR2 SDRAM mode register 2 (MR2), Actual write latency will add one if LP2_WL_normal = '0' and actual read latency will add one if

LP2_RL_normal = '0'.

For example, when LPDDR2 MR2 (RL and WL) is set to 4'b100, which t_{WL} = 3 and t_{RL} = 6,

If LP2_WL_normal = '0', actual t_{WL} used in the DDR controller will be 4.

If LP2_WL_normal = '1', actual t_{WL} used in the DDR controller will be 3.

If LP2_RL_normal = '0', actual t_{RL} used in the DDR controller will be 7.

If LP2_RL_normal = '1', actual t_{RL} used in the DDR controller will be 6.

4.5.6.73. LPDDR2 1-μs Wait Cycle Register (LP2WCR1, Offset = 0x120)

Table 4-118 LPDDR2 1-μs Wait Cycle Register (LP2WCR1, Offset = 0x120)

Bit	Name	Type	Default Value	Description
[31:20]	-	-	Rsvd	Reserved
[19:0]	1us_wait_cycle	RW	0x00216	LPDDR2 1-μs wait cycle register in the initial sequence Default = 20'h216

4.5.6.74. LPDDR2 10-μs Wait Cycle Register (LP2WCR2, Offset = 0x124)

Table 4-119 LPDDR2 10-μs Wait Cycle Register (LP2WCR2, Offset = 0x124)

Bit	Name	Type	Default Value	Description
[31:20]	-	-	Rsvd	Reserved
[19:0]	10us_wait_cycle	RW	0x014d6	LPDDR2 10-μs wait cycle register in the initial sequence Default = 20'h14d6

4.5.6.75. Reserved Register (Offset = 0x128)

4.5.6.76. Re-arbitration Disable Register (Offset = 0x12C)

When channel write request is de-asserted, arbiter will not perform arbitration during a small period. The period is defined by the ArbDisCnt register. This is used to let the same burst not to be broken. The read request will not be de-asserted at the same burst, thus, this register is only for the write operation.

Table 4-120 Re-arbitration Disable Register (REARBDISR, Offset = 0x12C)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	-

[30:28]	Ch7_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 7 When the write request of channel 7 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch7_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.
27	-	-	Rsvd	-
[26:24]	Ch6_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 6 When the write request of channel 6 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch6_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted … 3'h7: The arbiter will do arbitration 7 cycles later after the write request is de-asserted.
23	-	-	Rsvd	-
[22:20]	Ch5_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 5 When the write request of channel 5 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch5_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.
19	-	-	Rsvd	-
[18:16]	Ch4_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 4 When the write request of channel 4 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch4_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting)

Vangotech

				3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.
15	-	-	Rsvd	-
[14:12]	Ch3_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 3 When the write request of channel 3 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch3_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.
11	-	-	Rsvd	-
[10:8]	Ch2_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 2 When the write request of channel is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch2_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.
7	-	-	Rsvd	-
[6:4]	Ch1_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 1 When the write request of channel 1 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch1_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.

Vangotech

3	-	-	Rsvd	
[2:0]	Ch0_ArbDisCnt	RW	0	Re-arbitration disable counter of channel 0 When the write request of channel 0 is de-asserted, the arbiter will not do arbitration during a small period. The period is defined by the Ch0_ArbDisCnt register. 3'h0: The arbiter will immediately do arbitration once the write request is de-asserted. (Default setting) 3'h1: The arbiter will do arbitration one cycle after the write request is de-asserted. … 3'h7: The arbiter will do arbitration 7 cycles after the write request is de-asserted.

4.5.6.77. DDR PHY Read Path DLL Delay Tuning for Falling Edge Register (PHYRDTFR, Offset = 0x130)

DLSEL_RD_FALL_byten is used to adjust the timing relation between read DQ and read DQS at the falling edge, Register Offset 0x24 is used to adjust at the rising edge.

Table 4-121 DDR PHY Read Path DLL Delay Tuning for Falling Edge Register (PHYRDTFR, Offset = 0x130)

Bit	Name	Type	Default Value	Description
[31:28]	BYTE7_DLSEL_RD_FALL	RW	7	Delay setting of byte 7 to adjust the falling-edge delay of DQS or DQ for read
[27:24]	BYTE6_DLSEL_RD_FALL	RW	7	Delay setting of byte 6 to adjust the falling-edge delay of DQS or DQ for read
[23:20]	BYTE5_DLSEL_RD_FALL	RW	7	Delay setting of byte 5 to adjust the falling-edge delay of DQS or DQ for read
[19:16]	BYTE4_DLSEL_RD_FALL	RW	7	Delay setting of byte 4 to adjust the falling-edge delay of DQS or DQ for read
[15:12]	BYTE3_DLSEL_RD_FALL	RW	7	Delay setting of byte 3 to adjust the falling-edge delay of DQS or DQ for read
[11:8]	BYTE2_DLSEL_RD_FALL	RW	7	Delay setting of byte 2 to adjust the falling-edge delay of DQS or DQ for read
[7:4]	BYTE1_DLSEL_RD_FALL	RW	7	Delay setting of byte 1 to adjust the falling-edge delay of DQS or DQ for read
[3:0]	BYTE0_DLSEL_RD_FALL	RW	7	Delay setting of byte 0 to adjust the falling-edge delay of DQS or DQ for read

4.5.6.78. DDR PHY MISC2 Control Register (PHYMISCR2, Offset = 0x134)

Please refer to the DDRx PHY application note for more detailed information.

Vangotech

Table 4-122 DDR PHY MISC2 Register (PHYMISCR2, Offset = 0x134)

Bit	Name	Type	Default Value	Description
[31:28]	odt_mode_rd	RW	0	Read command ODT output option: B[28] = 1: Read rank0, ODT[0] = 1 B[28] = 0: Read rank0, ODT[0] = 0 B[29] = 1: Read rank1, ODT[0] = 1 B[29] = 0: Read rank1, ODT[0] = 0 B[30] = 1: Read rank0, ODT[1] = 1 B[30] = 0: Read rank0, ODT[1] = 0 B[31] = 1: Read rank1, ODT[1] = 1 B[31] = 0: Read rank1, ODT[1] = 0 Each bit is independent, for example: case1: odt_mode_rd = 4'b1001 when read rank0, ODT[0] = 1, ODT[1] = 0 when read rank1, ODT[0] = 0, ODT[1] = 1 case2: odt_mode_rd = 4'b1101 when read rank0, ODT[0] = 1, ODT[1] = 1 when read rank1, ODT[0] = 0, ODT[1] = 1 When in the DDR2 mode, RL < 4 is not supported.
[27:24]	odt_mode_wr	RW	9	Write command ODT output option: B[24] = 1: Write rank0, ODT[0] = 1 B[24] = 0: Write rank0, ODT[0] = 0 B[25] = 1: Write rank1, ODT[0] = 1 B[25] = 0: Write rank1, ODT[0] = 0 B[26] = 1: Write rank0, ODT[1] = 1 B[26] = 0: Write rank0, ODT[1] = 0 B[27] = 1: Write rank1, ODT[1] = 1 B[27] = 0: Write rank1, ODT[1] = 0 Each bit is independent, for example: case1: odt_mode_wr = 4'b0101 when write rank0, ODT[0] = 1, ODT[1] = 1 when write rank1, ODT[0] = 0, ODT[1] = 0 case2: odt_mode_wr = 1100: when write rank0, ODT[0] = 0, ODT[1] = 1, when write rank1, ODT[0] = 0, ODT[1] = 1 When in the DDR2 mode, RL < 4 is not supported.
23	-	-	Rsvd	Reserved
[22:20]	CMD_DLSEL	RW	7	DRAM command/address versus CK position adjustment
[19:18]	-	-	Rsvd	Reserved

Vangotech

17	RDLVL_GATE_EN	RW	0	MSDLY margin check 1 = Enable 0 = Disable
16	EYE_RDLVL_EN	RW	0	Eye margin check of DQ for DQS latch 1 = Enable 0 = Disable
15	-	-	Rsvd	Reserved
[14:12]	RONMD_DQS	RW	7	The driving impedance setting of DDR I/O (Data block) 3'b111: 34 Ω 3'b110: 40 Ω 3'b101: 48 Ω 3'b100: 60 Ω 3'b011: 60 Ω 3'b010: 80 Ω 3'b001: 120 Ω 3'b000: 240 Ω
11	-	-	Rsvd	Reserved
[10:8]	RONMD_DQ	RW	7	The driving impedance setting of DDR I/O (Data block) 3'b111: 34 Ω 3'b110: 40 Ω 3'b101: 48 Ω 3'b100: 60 Ω 3'b011: 60 Ω 3'b010: 80 Ω 3'b001: 120 Ω 3'b000: 240 Ω
7	-	-	Rsvd	Reserved
[6:4]	RONMD_CK	RW	7	The driving impedance setting of DDR I/O (Data block) 3'b111: 34 Ω 3'b110: 40 Ω 3'b101: 48 Ω 3'b100: 60 Ω 3'b011: 60 Ω 3'b010: 80 Ω 3'b001: 120 Ω 3'b000: 240 Ω
3	-	-	Rsvd	Reserved

Vangotech

[2:0]	RONMD_CMD	RW	7	The driving impedance setting of DDR I/O (CMDADDR block) 3'b111: 34 Ω 3'b110: 40 Ω 3'b101: 48 Ω 3'b100: 60 Ω 3'b011: 60 Ω 3'b010: 80 Ω 3'b001: 120 Ω 3'b000: 240 Ω

4.5.6.79. DDR ELASTIC FIFO Control Register (EFIFOCR, Offset = 0x138)

Users should write to this register before register 0x04 bit [0] "MemCmd_intiial" is set to '1'.

Table 4-123 DDR ELASTIC FIFO Control Register (EFIFOCR, Offset = 0x138)

Bit	Name	Type	Default Value	Description
[31:9]	-	-	Rsvd	Reserved
8	efifo_rst	RW	0	Writing to this bit will trigger elastic FIFO reset, this bit will be self-cleared. 1: Reset elastic FIFO, users can reset elastic FIFO only when there is no DDR command at the whole controller pipe line stage. 0: No effect Please set this bit to '0' in the initial sequence.
[7:5]	-	-	Rsvd	Reserved
4	auto_rst_srf	RW	0	Automatically reset elastic FIFO when self-refresh state is entered 1: Memory controller will automatically reset elastic FIFO when the self-refresh mode is entered. 0: Memory controller will not automatically reset elastic FIFO when the self-refresh mode is entered.
[3:2]	read_sync_pos	RW	0	Read data elastic FIFO synchronization position for POSTCTLCLK to MClk The valid values are 0, 1, and 2. Setting to 1 can work in most cases, while users can use post-simulation or STA report to decide the proper read synchronization position setting to reduce latency.
[1:0]	cmd_sync_pos	RW	0	Command elastic FIFO synchronization position for MClk to POSTCTLCLK The valid values are 0, 1, and 2.

Vangotech

				Setting to 1 can work in most cases, while users can use post-simulation or STA report to decide the proper command synchronization position setting to reduce latency.

4.5.6.80. Byte n Read Deskew Register (RBnDSKW, Offset = 0x160+4*n)

n means byte index. Byte 0 register offset equals to 0x160. Byte 1 register offset equals to 0x164. Please refer to the DDR PHY application note for more detailed information. This register will be valid when `define FTDDR3030_PERBIT_DESKEW_YES is configured.

Table 4-124 Byte n Read Deskew Register (RBnDSKW, Offset = 0x160+4 * n, n = 0 ~ 7)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:28]	BYTEn_DQ7BITDLSEL_RD	RW	4	Read data de-skew control of byte *n* bit 7
27	-	-	Rsvd	Reserved
[26:24]	BYTEn_DQ6BITDLSEL_RD	RW	4	Read data de-skew control of byte *n* bit 6
23	-	-	Rsvd	Reserved
[22:20]	BYTEn_DQ5BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 5
19	-	-	Rsvd	Reserved
[18:16]	BYTEn_DQ4BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 4
15	-	-	Rsvd	Reserved
[14:12]	BYTEn_DQ3BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 3
11	-	-	Rsvd	Reserved
[10:8]	BYTEn_DQ2BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 2
7	-	-	Rsvd	Reserved
[6:4]	BYTEn_DQ1BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 1
3	-	-	Rsvd	Reserved
[2:0]	BYTEn_DQ0BITDLSEL_RD	RW	4	Read data de-skew control of byte n bit 0

4.5.6.81. Byte n Write Deskew Register (WBnDSKW, Offset = 0x180+4*n)

n means byte index. Byte 0 register offset equals to 0x180. Byte 1 register offset equals to 0x184. Please refer to the DDR PHY application note for more detailed information. This register will be valid when `define FTDDR3030_PERBIT_DESKEW_YES is configured.

Table 4-125 Byte n Write Deskew Register (WBnDSKW, Offset = 0x180+4*n, n = 0 ~ 7)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:28]	wrdllsel_byte*n*_bit7	RW	3	Write data de-skew control of byte *n* bit 7
27	-	-	Rsvd	Reserved
[26:24]	wrdllsel_byte*n*_bit6	RW	3	Write data de-skew control of byte *n* bit 6

23	-	-	Rsvd	Reserved
[22:20]	wrdllsel_byte*n*_bit5	RW	3	Write data de-skew control of byte *n* bit 5
19	-	-	Rsvd	Reserved
[18:16]	wrdllsel_byte*n*_bit4	RW	3	Write data de-skew control of byte *n* bit 4
15	-	-	Rsvd	Reserved
[14:12]	wrdllsel_byte*n*_bit3	RW	3	Write data de-skew control of byte *n* bit 3
11	-	-	Rsvd	Reserved
[10:8]	wrdllsel_byten_bit2	RW	3	Write data de-skew control of byte n bit 2
7	-	-	Rsvd	Reserved
[6:4]	wrdllsel_byten_bit1	RW	3	Write data de-skew control of byte n bit 1
3	-	-	Rsvd	Reserved
[2:0]	wrdllsel_byten_bit0	RW	3	Write data de-skew control of byte n bit 0

4.5.6.82. Write Data Mask Deskew Register (WDMDSKW, Offset = 0x1A0)

Please refer to the DDR PHY application note for more detailed information. This register will be valid when `define FTDDR3030_PERBIT_DESKEW_YES is configured.

Table 4-126 Write Data Mask Deskew Register (WDMDSKW, Offset = 1A0)

Bit	**Name**	**Type**	**Default Value**	**Description**
31	-	-	Rsvd	Reserved
[30:28]	BYTE7_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 7
27	-	-	Rsvd	Reserved
[26:24]	BYTE6_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 6
23	-	-	Rsvd	Reserved
[22:20]	BYTE5_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 5
19	-	-	Rsvd	Reserved
[18:16]	BYTE4_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 4
15	-	-	Rsvd	Reserved
[14:12]	BYTE3_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 3
11	-	-	Rsvd	Reserved
[10:8]	BYTE2_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 2
7	-	-	Rsvd	Reserved
[6:4]	BYTE1_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 1

Vangotech

3	-	-	Rsvd	Reserved
[2:0]	BYTE0_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 0

4.5.6.83. Byte 8 Read Deskew Register (RB8DSKW, Offset = 0x1A4)

Please refer to the DDR PHY application note for more detailed information. This register will be valid when `define FTDDR3030_PERBIT_DESKEW_YES and FTDDR3030_ECC_YES are configured. This register is for ECC byte.

Table 4-127 Byte 8 Read Deskew Register (RB8DSKW, Offset = 0x1A4)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:28]	BYTE8_DQ7BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 7
27	-	-	Rsvd	Reserved
[26:24]	BYTE8_DQ6BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 6
23	-	-	Rsvd	Reserved
[22:20]	BYTE8_DQ5BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 5
19	-	-	Rsvd	Reserved
[18:16]	BYTE8_DQ4BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 4
15	-	-	Rsvd	Reserved
[14:12]	BYTE8_DQ3BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 3
11	-	-	Rsvd	Reserved
[10:8]	BYTE8_DQ2BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 2
7	-	-	Rsvd	Reserved
[6:4]	BYTE8_DQ1BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 1
3	-	-	Rsvd	Reserved
[2:0]	BYTE8_DQ0BITDLSEL_RD	RW	3	Read data de-skew control of byte 8 bit 0

4.5.6.84. Byte 8 Write Deskew Register (WB8DSKW, Offset = 0x1A8)

Please refer to the DDR PHY application note for more detailed information. This register will be valid when `define FTDDR3030_PERBIT_DESKEW_YES and FTDDR3030_ECC_YES is configured. This register is for ECC byte.

Table 4-128 Byte 8 Write Deskew Register (WB8DSKW, Offset = 0x1A8)

Bit	Name	Type	Default Value	Description
31	-	-	Rsvd	Reserved
[30:28]	wrdllsel_byte8_bit7	RW	3	Write data de-skew control of byte 8 bit 7
27	-	-	Rsvd	Reserved
[26:24]	wrdllsel_byte8_bit6	RW	3	Write data de-skew control of byte 8 bit 6
23	-	-	Rsvd	Reserved

Vangotech

[22:20]	wrdllsel_byte8_bit5	RW	3	Write data de-skew control of byte 8 bit 5
19	-	-	Rsvd	Reserved
[18:16]	wrdllsel_byte8_bit4	RW	3	Write data de-skew control of byte 8 bit 4
15	-	-	Rsvd	Reserved
[14:12]	wrdllsel_byte8_bit3	RW	3	Write data de-skew control of byte 8 bit 3
11	-	-	Rsvd	Reserved
[10:8]	wrdllsel_byte8_bit2	RW	3	Write data de-skew control of byte 8 bit 2
7	-	-	Rsvd	Reserved
[6:4]	wrdllsel_byte8_bit1	RW	3	Write data de-skew control of byte 8 bit 1
3	-	-	Rsvd	Reserved
[2:0]	wrdllsel_byte8_bit0	RW	3	Write data de-skew control of byte 8 bit 0

4.5.6.85. Byte 8 PHY Control Register (B8_PHYCR, Offset = 0x1AC)

This register is valid when FTDDR3030_ECC_YES is configured.

Table 4-129 Byte8 PHY Control Register (B8_PHYCR, Offset = 0x1AC)

Bit	Name	Type	Default Value	Description
31	rlevel_byte0_pass	RO	0	Read-leveling status for byte 8 1'b1 = Read-leveling is passed. 1'b0 = Read-leveling is failed or read-leveling hardware is disabled.
[30:28]	BYTE8_DQDMBITDLSEL_WR	R/W	3	Write data mask de-skew control of byte 8 This field is valid when FTDDR3030_PERBIT_DESKEW_YES is configured.
[27:24]	BYTE1_DLSEL_WR	RW	8	Delay control for byte 1 to add a delay in DQS or DQ Please refer to Register Offset 0x78 for details.
[23:20]	DLSEL_RD_FALL_byte8	RW	7	Delay setting of byte 8 to adjust the falling-edge delay of DQS or DQ for read Please refer to Register Offset 0x130 for details.
[19:16]	BYTE8_DLSEL_RD	RW	7	Delay value control to add a delay into read DQS or read DQ Please refer to Register Offset 0x24 for details.
15	wlevel_byte8_pass	RO	0	Byte 8 of the write-leveling status register 1'b1 = Write-leveling passed 1'b0 = Write-leveling failed or write-leveling

Vangotech

				hardware disabled
[14:8]	BYTE8_PDLSET	RW	0	Programmable DLL delay for byte 8 wlevel_byte8_hw_disable = '0': Hardware tuning value wlevel_byte8_hw_disable = '1': Software programmed value Please refer to Register Offset 0x64 for details.
7	dqs_sel_wr_byte8	RW	0	Add a delay in DQ or DQS, the delay value is controlled by BYTE8_DLSEL_WR. 1'b1 = Write DQ is fixed, only delays DQS. 1'b0 = Write DQS is fixed, only delays DQ. Please refer to Register Offset 0x6C for details. For 502 PHY, this bit is not used.
6	dqs_sel_byte8	RW	0	Add a delay in DQ or DQS, the delay value is controlled by BYTE8_DLSEL_RD. 1'b1 = Read DQ is fixed, only delays DQS. 1'b0 = Read DQS is fixed, only delays DQ. Please refer to Register Offset 0x6C for details. For 502 PHY, this bit is not used.
5	wlevel_byte8_hw_disable	RW	1	Byte 8 of the write-leveling hardware disable 1'b1 = Write-leveling by software scan (Hardware disabled) 1'b0 = Write-leveling by hardware Please refer to Register Offset 0x60 for details.
4	rlevel_byte8_hw_disable	RW	1	Read-leveling hardware disable for byte 8 1'b1 = Read-leveling by software (Hardware disabled) 1'b0 = Read-leveling by hardware Please refer to Register Offset 0x70 for details. This bit cannot be independently used. When setting byte 8 Read-leveling by hardware, users have to set at least 1 byten (n = 0~7) Read-leveling by hardware.
[3:0]	BYTE8_MSDLY	RW	1	Programmable read DQS filtering window for byte 8 rlevel_byte8_hw_disable = '0', hardware

Vangotech

				tuning value rlevel_byte8_hw_disable = '1', software programmed value Please refer to Register Offset 0x74 for details.

4.5.6.86. BIST Control Register (BIST_CTRL, Offset = 0x1B0)

To start the BIST test sequence, please follow the steps below:

(1) Set 0x1B4 "bist_start_addr"

(2) Set 0x1B8 "bist_size"

(3) Set 0x1B0 bit[2:1] pattern_sel and set bit[0] "bist_en" = 1

Please note that this BIST is a simple DDR test and cannot be used as KGD or mass production test.

Table 4-130 BIST Control Register (BISTCR, Offset = 0x01B0)

Bit	Name	Type	Default Value	Description
[31:4]	-	-	Rsvd	Reserved
7	bist_rank_decode	RW	0	0 = Select Rank 0 to do BIST 1 = Select Rank 1 to do BIST
[6:4]	bist_rank_type	RW	0	Rank type for the BIST mode Please refer to the 0x10 B[6:4] setting.
3	bist_wr_only	RW	0	1 = For BIST logic issue write commands only. There is no read and compare action. This is often used for filling initial data to DDR for the ECC operation. 0 = Normal BIST operation, including write, read, and comparison
[2:1]	pattern_sel	RW	0	pattern_sel is the BIST test pattern select. All bytes (Byte1, byte2, …, byte7 are the same with byte0.) 0 = 0x00 is used for BIST test data byte0 content. 1 = 0xff is used for BIST test data byte0 content. 2 = 0x5a, 0xa5, 0x5a, 0xa5, … repeated for BIST test data byte0 content 3 = Mixed data content The data byte0 will repeat the six data contents every 1K byte. (1) 128 bytes 0x00 (2) 128 bytes 0xff

Vangotech

				(3) 128 bytes 0x00, 0xff, 0x00, 0xff, …repeated (4) 128 bytes 0x5a, 0xa5, 0x5a, 0xa5, …repeated (5) 256 bytes data increment 0x00, 0x01, 0x02, …, 0xfe, 0xff (6) 256 bytes data decrement 0xff, 0xfe, 0xfd, …, 0x01, 0x00
0	bist_en	RW	0	0 = Disable BIST test after the DDR initial sequence 1 = Enable BIST test after the DDR initial sequence

4.5.6.87. BIST Start Address Register (BIST_START_ADDR, Offset = 0x1B4)

The BIST start address is considered to be DDR offset address. It is not necessary to consider the base address defined in control register 0x10 "RNK0_BASE".

For example, the 1G bytes address ranges from 0x00000000 to 0x3fffffff for bist_start_addr.

Table 4-131 BIST Start Address Register (BIST_START_ADDR, Offset = 0x01B4)

Bit	Name	Type	Default Value	Description
[31:0]	bist_start_addr	RW	0	32-bit BIST start address BIST start address should be aligned with these size boundaries. Examples: When memory width = 8 bits, the start address should align with 1K bytes boundaries. When memory width = 16 bits, the start address should align with 2K bytes boundaries. When memory width = 32 bits, the start address should align with 4K bytes boundaries. When memory width = 64 bits, the start address should align with 8K bytes boundaries.

Please note the following:

If FTDDR3030_LOGIC_ADDR_YES is configured, the actual 33-bit BIST start address is

{bist_start_addr[31:0], 1'b0}.

If FTDDR3030_LOGIC_ADDR_NO is configured, the actual 32-bit BIST start address is

bist_start_addr[31:0].

4.5.6.88. BIST Test Size Register (BIST_SIZE, Offset = 0x1B8)

Table 4-132 BIST Test Size Register (BIST_Test_Size, Offset = 0x01B8)

Bit	Name	Type	Default Value	Description
[31:0]	bist_size	RW	0	(1) The bist_size (In terms of bytes) should be less than the total useful DDR size. (2) The bist_size (In terms of bytes) should be the multiples of the allowable smallest size. When memory width = 8 bits, the allowable smallest size =1K bytes. When memory width = 16 bits, the allowable smallest size = 2K bytes. When memory width = 32 bits, the allowable smallest size = 4K bytes. When memory width = 64 bits, the allowable smallest size = 8K bytes.

Please note the following:

If FTDDR3030_LOGIC_ADDR_YES is configured, the actual 33-bit BIST size is {bist_size[31:0], 1'b0}.

If FTDDR3030_LOGIC_ADDR_NO is configured, the actual 32-bit BIST size is bist_size[31:0].

4.5.6.89. Reserved Register (Reserved, Offset = 0x1BC)

4.5.6.90. BIST Result Register (BIST_RESULT, Offset = 0x1C0)

Users can check the BIST test status, "bist_pass", only when "bist_end" = 1. It is meaningless to check "bist_pass" when "bist_end" = 0. If 0x1B0 bit[3] "bist_wr_only" = 1, bist_pass is meaningless because there is no comparison action.

Table 4-133 BIST Result Register (BIST_RESULT, Offset = 0x01C0)

Bit	Name	Type	Default Value	Description
[31:2]	-	-	Rsvd	Reserved
1	bist_pass	RO	1	When "bist_end" =1: 0 = BIST test is failed. 1 = BIST test is passed.
0	bist_end	RO	0	When "bist_end" =0: 0 = BIST test is not enabled or BIST test is not finished. 1 = BIST test is finished.

4.5.6.91. BIST Fail Address Register (BIST_FAIL_ADDR, Offset = 0x1C4)

Table 4-134 BIST Fail Address Register (BIST_FAIL_ADDR, Offset = 0x01C4)

Bit	Name	Type	Default Value	Description

[31:0]	bist_fail_addr	RO	0	When the BIST test is failed, the first wrong address will be latched in this register.

Please note the following:

If FTDDR3030_LOGIC_ADDR_YES is configured, the actual 33-bit BIST fail address is

{bist_fail_addr[31:0], 1'b0}.

If FTDDR3030_LOGIC_ADDR_NO is configured, the actual 32-bit BIST fail address is

bist_fail_addr[31:0].

4.5.6.92. BIST Fail Data Low Register (BIST_FAIL_DATA_LOW, Offset = 0x1C8)

Table 4-135 BIST Fail Data Low Register (BIST_FAIL_DATA_LOW, Offset = 0x01C8)

Bit	Name	Type	Default Value	Description
[31:0]	bist_fail_data_low	RO	0	When the BIST test is failed, the first wrong data will be latched in this register.

4.5.6.93. BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x1CC)

Table 4-136 BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x01CC)

Bit	Name	Type	Default Value	Description
[31:0]	bist_fail_data_high	RO	0	When the BIST test is failed, the first wrong data will be latched in this register. For memory width = 64 bits, the 64-bit failed data consists of {bist_fail_data_high, bist_fail_data_low}.

4.5.6.94. BIST Expect Data Low Register (BIST_EXP_DATA_LOW, Offset = 0x1D0)

Table 4-137 BIST Expect Data Low Register (BIST_EXP_DATA_LOW, Offset = 0x01D0)

Bit	Name	Type	Default Value	Description
[31:0]	bist_exp_data_low	RO	0	When the BIST test is failed, the related golden data is latched for the user diagnosis.

Vangotech

4.5.6.95. BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x1D4)

Table 4-138 BIST Fail Data High Register (BIST_FAIL_DATA_HIGH, Offset = 0x01D4)

Bit	Name	Type	Default Value	Description
[31:0]	bist_exp_data_high	RO	0	When the BIST test is failed, the related golden data is latched for the user diagnosis. For memory width = 64 bits, the 64-bit golden data consists of {bist_exp_data_high, bist_exp_data_low}.

4.5.6.96. Reserved Register (Reserved, Offset = 0x1D8)

4.5.6.97. Reserved Register (Reserved, Offset = 0x1DC)

4.5.6.98. ECC Control Register (ECC_CTRLR, Offset = 0x1E0)

Users can set "chx_rmw_lock_en = 1" to issue the lock command between read and write commands during RMW (Read-Modify-Write) operation. If users can guarantee that multiple channels will not write the same address as other channels, chx_rmw_lock_en can be 0 to improve DDR access utilization.

Table 4-139 ECC Control Register (ECC_CTRLR, Offset = 0x01E0)

Bit	Name	Type	Default Value	Description
[31:16]	-	-	Rsvd	Reserved
15	ch7_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write command during RMW operation
14	ch6_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
13	ch5_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
12	ch4_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation

Vangotech

11	ch3_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
10	ch2_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
9	ch1_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
8	ch0_rmw_lock_en	RW	0	0 = Disable lock read to write commands during RMW operation 1 = Enable lock read to write commands during RMW operation
[7:5]	-	-	Rsvd	Reserved
4	rmw_rdata_ecc_disable	RW	0	0 = Enable read data ECC correction during RMW operation 1 = Disable read data ECC correction during RMW operation
3	ecc_int_mask	RW	0	0 = Enable ECC interrupt 1 = Disable ECC interrupt
[2:1]	force_error	RW	0	Insert error at the ECC encoder output It is used for the test purpose. 0 = Do not insert error 1 = Insert single-bit error 2 = Insert double-bit error 3 = Insert triple-bit error
0	ecc_enable	RW	0	0 = ECC disable 1 = ECC enable

4.5.6.99. ECC Interrupt Control Register (ECC_INTCR, Offset = 0x1E4)

The ECC engine can correct one-bit error and detect two or more bit error. If two or more bit error occurs, the read data cannot be corrected, and the corresponding ECC interrupt will be asserted.

Writing '1' can clear the interrupt and the status at 0x1E4, 0x1E8, 0x1F0, 0x1F4, 0x1F8, and 0x1FC.

Table 4-140 ECC Interrupt Control Register (ECC_INTCR, Offset = 0x01E4)

Bit	Name	Type	Default Value	Description
[31:16]	-	-	Rsvd	Reserved

Vangotech

0	ecc_int	W1C	0	ECC interrupt Writing '1' will clear the interrupt and error status. 0 = CH0 ECC interrupt is not asserted. 1 = CH0 ECC interrupt is asserted.

4.5.6.100. ECC Error Channel Register (ECC_CHR, Offset = 0x1E8)

After ECC check, there are four types of ECC status:

0 = No ECC error

1 = Single-bit error, and the error can be corrected.

2 = Double or more bit errors, and the read data is not corrected.

3 = Overall parity bit error (P7 for 64bits and P6 for 32bits)

If ECC errors occur and the errors cannot be corrected (Status = 2), the associated channel will be recorded in 0x1E8, and the system address will be recorded in 0x1EC.

Table 4-141 ECC Error Channel Register (ECC_CHR, Offset = 0x01E8)

Bit	Name	Type	Default Value	Description
[31:4]	-	-	Rsvd	Reserved
3	-	-	Rsvd	Reserved
[2:0]	ecc_err_channel	RO	0	The first occurred ECC error channel will be recorded when status = 2. Writing Register Offset 0x1E4 bit[0] = '1' will clear this register to zero.

4.5.6.101. ECC Error Address Register (ECC_EAR, Offset = 0x1EC)

Table 4-142 ECC Error Address Register (ECC_EAR, Offset = 0x01EC)

Bit	Name	Type	Default Value	Description
[31:0]	ecc_err_addr	RO	0	The first occurred ECC error address will be recorded when ECC check status = 2. Writing Register Offset 0x1E4 bit[0] = '1' will clear this register to zero.

Please note the following:

If FTDDR3030_LOGIC_ADDR_YES is configured, the actual 33-bit ECC error address is {ecc_err_addr[31:0], 1'b0}.

Vangotech

If FTDDR3030_LOGIC_ADDR_NO is configured, the actual 32-bit ECC error address is ecc_err_addr[31:0].

4.5.6.102. ECC Error Bit Position Low Byte Register (ECC_EBPLR, Offset = 0x1F0)

The ECC engine can record the first ECC read data one-bit error position. When first one-bit ECC error occurs, the corresponding ECC error bit position is recorded. If the one-bit error occurs at parity bit, it will not be shown in this register because this register indicates data bit error. If two or more bit error occurs, this register also cannot show any information.

Table 4-143 ECC Error Bit Position Low Byte Register (ECC_EBPLR, Offset = 0x01F0)

Bit	Name	Type	Default Value	Description
[31:0]	ecc_error_bit_pos_low	RO	0	32-bit error bit position in low bytes (Bytes 3, 2, 1, 0) The read data error bit will be '1' in the related bit position if ECC error occurs. For example, 32'h0000_0001 indicates that there are errors in the memory bus bit[0]. Writing 0x1E4 bit[0] = 1 will clear this register to zero.

4.5.6.103. ECC Error Bit Position High Byte Register (ECC_EBPHR, Offset = 0x1F4)

Table 4-144 ECC Error Bit Position High Byte Register (ECC_EBPHR, Offset = 0x01F4)

Bit	Name	Type	Default Value	Description
[31:0]	ecc_error_bit_pos_high	RO	0	32bit error bit position in high bytes (bytes7, 6, 5, 4) The read data error bit will be '1' in the related bit position if ECC error occurs. For example, 32'h0000_0001 indicates that there are errors in the memory bus bit[32]. Writing 0x1E4 bit[0] = 1 will clear this register to zero.

4.5.6.104. ECC One-bit error Count Register (ECC_ERR1_COUNT, Offset = 0x1F8)

When the ECC check logic status = 1, it indicates "Single bit read data error and the read data is corrected". This register records the single error bits count until the software clears this register.

Table 4-145 ECC One-bit Error Count Register (ECC_ERR1_COUNT, Offset = 0x01F8)

Bit	Name	Type	Default Value	Description
[31:0]	ecc_one_bit_err_count	RO	0	One-bit error counts When the counts are equal or larger than 32'hffff_fff8, the counts will be fixed to prevent overflow. Writing 0x1E4 bit[0] = 1 will clear this count to zero.

Notes:

1. For IP version (2.15.0), this counter records one-bit data error and parity error, but overall parity error is not included.

2. For IP version (2.16.0) and later, this counter records one-bit data error, parity error, and overall parity error.

4.5.6.105. ECC Two-bit Error Count Register (ECC_ERR2_COUNT, Offset = 0x1FC)

When the ECC check logic status = 2, it indicates "Double or more errors and the read data is not corrected". This register records the two or more error bit count until the software clears this register.

Table 4-146 ECC Two-bit Error Count Register (ECC_ERR2_COUNT, Offset = 0x01FC)

Bit	Name	Type	Default Value	Description
[31:0]	ecc_two_bit_err_count	RO	0	Two-bit error counts When the counts are equal or larger than 32'hffff_fff8, the counts will be fixed to prevent overflow. Writing 0x1E4 bit[0] = 1 will clear this count to zero0.

4.5.7. Memory Address Table (MA Table)

The DDR controller supports various types of DDR2/DDR3 SDRAMs. Users must refer to the MA tables for the external rank0 configuration register, RNK_TYPE.

Each rank must choose the correct type of the MA tables. FTDDR3030 supports three types of MA tables, which can be configured through the AMTSEL bit of the configuration register (Offset 0x00).

• When AMTSEL = 2'b00, the system address will map with RA, BA, and CA.

• When AMTSEL = 2'b01, the system address will map with BA, RA, and CA.

Vangotech

• When AMTSEL = 2'b10, the system address will map with RA, CA, BA, and CA.

This field works only when the hardware configuration of FTDDR3030_BANK_SWIZZLE_YES is enabled.

For a 64-bit memory interface, the bank address mapping is inserted between the column addresses.

ADDR is mapped to RA, CA, BA, and CA.

• Bits[9:8] of the system address are configured as the bank addresses for four banks of DDRx SDRAM.

• Bits[10:8] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM.

For a 32-bit memory interface, the bank address mapping is inserted between the column addresses.

ADDR is mapped to RA, CA, BA, and CA.

• Bits[8:7] of the system address are configured as the bank addresses for four banks of DDRx SDRAM.

• Bits[9:7] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM.

For a 16-bit memory interface, the bank address mapping is inserted between the column addresses.

ADDR is mapped to RA, CA, BA, and CA.

• Bits[7:6] of the system address are configured as the bank addresses for four banks of DDRx SDRAM.

• Bits[8:6] of the system address are configured as the bank addresses for eight banks of DDRx SDRAM.

Note: If the hardware configuration `define FTDDR3030_BANKNUM_4 is configured, BA (Bank Address) will only support 2 bits.

4.5.7.1. AHB/AXI Address to Memory Address Mapping

The DDR address should be an offset, "addr_offset", from the AHB/AXI original address, "addr_org", minus rankn_base.

If the memory address access region is rank0_base ≤ addr_org < rank0_base + rank0_size,

addr_offset = {(addr_org - rank0_base), 24'h0}.

If the memory address access region is rank1_base ≤ addr_org < rank1_base + rank1_size,

addr_offset = {(addr_org - rank1_base), 24'h0}.

Please note that the AHB/AXI address for FTDDR3030, which is outside valid rank0 and rank1 region

(rank1_en = 1), is not allowed.

Please refer to Table 2-141 and Table 2-142, and the address mapping should be shifted if the DDR memory width is larger than 8 bits. The actual address used for address mapping table is "addr_ma".

Table 4-147 AHB/AXI Address for DDR Address Mapping when FTDDR3030_LOGIC_ADDR_YES is Configured

Memory Width	**Address for DDR Address Mapping (addr_ma)**
8	addr_ma[32:0] = addr_offset[32:0]
16	addr_ma[31:0] = addr_offset[32:1]
32	addr_ma[30:0] = addr_offset[32:2]
64	addr_ma[29:0] = addr_offset[32:3]

Table 4-148 AHB/AXI Address for DDR Address Mapping when FTDDR3030_LOGIC_ADDR_NO is Configured

Memory Width	**Address for DDR Address Mapping (addr_ma)**
8	addr_ma[31:0] = addr_offset[31:0]
16	addr_ma[30:0] = addr_offset[31:1]
32	addr_ma[29:0] = addr_offset[31:2]
64	addr_ma[28:0] = addr_offset[31:3]

4.5.7.2. MA Table1

MA table1 is enabled when FTDDR3030_LOGIC_ADDR_YES and FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5 (DDR2/DDR3/LPDDR2) are configured.

Table 4-149 MA Table1, AMTSEL = 0

(1) MA_table1, AMTSEL = 0				
FTDDR3030_LOGIC_ADDR_YES				
FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5				
Rank Type	**Row x Column x Bank**	**Row**	**Bank**	**Column**
0	13x10x3	[25:13]	[12:10]	[9:0]
1	14x10x3	[26:13]	[12:10]	[9:0]
2	15x10x3	[27:13]	[12:10]	[9:0]
3	16x10x3	[28:13]	[12:10]	[9:0]
4	16x11x3	[29:14]	[13:11]	[10:0]
5	14x9x3	[25:12]	[11:19]	[8:0]
6	14x11x3	[27:14]	[13:11]	[10:0]
7	15x11x3	[28:14]	[13:11]	[10:0]

Note: rank_type = 4, 6 and 7, addr_ma[10] is mapped to column[11], and the column address[10] = 0 along with read/write command.

Table 4-150 MA Table1, AMTSEL = 1

(1) MA_table1, AMTSEL = 1
FTDDR3030_LOGIC_ADDR_YES

Vangotech

FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5				
Rank Type	**Row x Column x Bank**	**Bank**	**Row**	**Column**
0	13x10x3	[25:23]	[22:10]	[9:0]
1	14x10x3	[26:24]	[23:10]	[9:0]
2	15x10x3	[27:25]	[24:10]	[9:0]
3	16x10x3	[28:26]	[25:10]	[9:0]
4	16x11x3	[29:27]	[26:11]	[10:0]
5	14x9x3	[25:23]	[22:9]	[8:0]
6	14x11x3	[27:25]	[24:11]	[10:0]
7	15x11x3	[28:26]	[25:11]	[10:0]

Note: rank_type = 4, 6 and 7, addr_ma[10] is mapped to column[11], and the column address[10] = 0 along with read/write command.

Table 4-151 MA Table1, AMTSEL = 2

(2) MA_table1, AMTSEL = 2				
FTDDR3030_LOGIC_ADDR_YES				
FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4, 5				
Rank Type	**Row x Column x Bank**	**Row**	**Column**	**Bank**
0	13x10x3	[25:13]	[12:8]	[7:5]
1	14x10x3	[26:13]	[12:8]	[7:5]
2	15x10x3	[27:13]	[12:8]	[7:5]
3	16x10x3	[28:13]	[12:8]	[7:5]
4	16x11x3	[29:14]	[13:8]	[7:5]
5	14x9x3	[25:12]	[11:8]	[7:5]
6	14x11x3	[27:14]	[13:8]	[7:5]
7	15x11x3	[28:14]	[13:8]	[7:5]

Note: rank_type = 4, 6 and 7, addr_ma[10] is mapped to column[11], and the column address[10] = 0 along with read/write command.

4.5.7.3. MA Table2

MA table2 is enabled when FTDDR3030_LOGIC_ADDR_NO and FTDDR3030_DDR_SUPPORT_TYPE_1 (DDR1/LPDDR1) are configured.

Table 4-152 MA Table2, AMTSEL = 0

(2) MA_table2, AMTSEL = 0
FTDDR3030_LOGIC_ADDR_NO
FTDDR3030_DDR_SUPPORT_TYPE_1

Vangotech

Rank Type	Row x Column x Bank	Row	Bank	Column
0	12x8x2	[21:10]	[9:8]	[7:0]
1	12x9x2	[22:11]	[10:9]	[8:0]
2	13x9x2	[23:11]	[10:9]	[8:0]
3	12x10x2	[23:12]	[11:10]	[9:0]
4	13x10x2	[24:12]	[11:10]	[9:0]
5	14x10x2	[25:12]	[11:10]	[9:0]

Note: rank_type = 0 is not supported at memory width = 8 bits.

Table 4-153 MA Table2, AMTSEL = 1

(2) MA_table2, AMTSEL = 1				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_1				
Rank Type	**Row x Column x Bank**	**Bank**	**Row**	**Column**
0	12x8x2	[21:20]	[19:8]	[7:0]
1	12x9x2	[22:21]	[20:9]	[8:0]
2	13x9x2	[23:22]	[21:9]	[8:0]
3	12x10x2	[23:22]	[21:10]	[9:0]
4	13x10x2	[24:23]	[22:10]	[9:0]
5	14x10x2	[25:24]	[23:10]	[9:0]

Note: rank_type = 0 is not supported at memory width = 8 bits.

Table 4-154 MA Table2, AMTSEL = 2

(2) MA_table2, AMTSEL = 2				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_1				
Rank Type	**Row x Column x Bank**	**Row**	**Column**	**Bank**
0	12x8x2	[21:10]	[9:7]	[6:5]
1	12x9x2	[22:11]	[10:7]	[6:5]
2	13x9x2	[23:11]	[10:7]	[6:5]
3	12x10x2	[23:12]	[11:7]	[6:5]
4	13x10x2	[24:12]	[11:7]	[6:5]
5	14x10x2	[25:12]	[11:7]	[6:5]

Notes:

(1) rank_type = 0 is not supported at memory width = 8 bits.

(2) AMTSEL = 2 is not supported at memory width = 8 bits.

Vangotech

4.5.7.4. MA Table3

MA table3 is enabled when FTDDR3030_LOGIC_ADDR_NO and FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4 (DDR2/DDR3) are configured.

Table 4-155 MA Table3, AMTSEL = 0

(2) MA_table3, AMTSEL = 0				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4				
Rank Type	**Row x Column x Bank**	**Row**	**Bank**	**Column**
0	13x9x2	[23:11]	[10:9]	[8:0]
1	13x10x2	[24:12]	[11:10]	[9:0]
2	14x10x2	[25:12]	[11:10]	[9:0]
3	12x10x3	[24:13]	[12:10]	[9:0]
4	13x10x3	[25:13]	[12:10]	[9:0]
5	14x10x3	[26:13]	[12:10]	[9:0]
6	15x10x3	[27:13]	[12:10]	[9:0]
7	16x10x3	[28:13]	[12:10]	[9:0]

Table 4-156 MA Table3, AMTSEL = 1

(2) MA_table3, AMTSEL = 1				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4				
Rank Type	**Row x Column x Bank**	**Bank**	**Row**	**Column**
0	13x9x2	[23:22]	[21:9]	[8:0]
1	13x10x2	[24:23]	[22:10]	[9:0]
2	14x10x2	[25:24]	[23:10]	[9:0]
3	12x10x3	[24:22]	[21:10]	[9:0]
4	13x10x3	[25:23]	[22:10]	[9:0]
5	14x10x3	[26:24]	[23:10]	[9:0]
6	15x10x3	[27:25]	[24:10]	[9:0]
7	16x10x3	[28:26]	[25:10]	[9:0]

Table 4-157 MA Table3, AMTSEL = 2

(2) MA_table3, AMTSEL = 2					
FTDDR3030_LOGIC_ADDR_NO					
FTDDR3030_DDR_SUPPORT_TYPE_2, 3, 4					
Rank Type	**Row x Column x Bank**	**Row**	**Column**	**Bank**	**Column**
0	13x9x2	[23:11]	[10:7]	[6:5]	[4:0]
1	13x10x2	[24:12]	[11:7]	[6:5]	[4:0]

Vangotech

2	14x10x2	[25:12]	[11:7]	[6:5]	[4:0]
3	12x10x3	[24:13]	[12:8]	[7:5]	[4:0]
4	13x10x3	[25:13]	[12:8]	[7:5]	[4:0]
5	14x10x3	[26:13]	[12:8]	[7:5]	[4:0]
6	15x10x3	[27:13]	[12:8]	[7:5]	[4:0]
7	16x10x3	[28:13]	[12:8]	[7:5]	[4:0]

Note: AMTSEL = 2 is not supported at memory width = 8 bits.

4.5.7.5. MA Table4

MA table4 is enabled when FTDDR3030_LOGIC_ADDR_NO and FTDDR3030_DDR_SUPPORT_TYPE_5 (DDR2/DDR3/LPDDR2) are configured.

Table 4-158 MA Table4, AMTSEL = 0

(2) MA_table4, AMTSEL = 0				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_5				
Rank Type	**Row x Column x Bank**	**Row**	**Bank**	**Column**
0	13x9x2	[23:11]	[10:9]	[8:0]
1	13x9x3	[24:12]	[11:9]	[8:0]
2	14x9x3	[25:12]	[11:9]	[8:0]
3	12x10x3	[24:13]	[12:10]	[9:0]
4	13x10x3	[25:13]	[12:10]	[9:0]
5	14x10x3	[26:13]	[12:10]	[9:0]
6	15x10x3	[27:13]	[12:10]	[9:0]
7	13x10x2	[24:12]	[11:10]	[9:0]

Note: rank_type = 1 and 2 are not supported at memory width = 8 bits and 16 bits.

Table 4-159 MA Table4, AMTSEL = 1

(2) MA_table4, AMTSEL = 1				
FTDDR3030_LOGIC_ADDR_NO				
FTDDR3030_DDR_SUPPORT_TYPE_5				
Rank Type	**Row x Column x Bank**	**Bank**	**Row**	**Column**
0	13x9x2	[23:22]	[21:9]	[8:0]
1	13x9x3	[24:22]	[21:9]	[8:0]
2	14x9x3	[25:23]	[22:9]	[8:0]
3	12x10x3	[24:22]	[21:10]	[9:0]
4	13x10x3	[25:23]	[22:10]	[9:0]
5	14x10x3	[26:24]	[23:10]	[9:0]

Vangotech

6	15x10x3	[27:25]	[24:10]	[9:0]
7	13x10x2	[24:23]	[22:10]	[9:0]

Note: rank_type = 1 and 2 are not supported at memory width = 8 bits and 16 bits.

Table 4-160 MA Table4, AMTSEL = 2

(2) MA_table4, AMTSEL = 2					
FTDDR3030_LOGIC_ADDR_NO					
FTDDR3030_DDR_SUPPORT_TYPE_5					
Rank Type	**Row x Column x Bank**	**Row**	**Column**	**Bank**	**Column**
0	13x9x2	[23:11]	[10:7]	[6:5]	[4:0]
1	13x9x3	[24:12]	[11:8]	[7:5]	[4:0]
2	14x9x3	[25:12]	[11:8]	[7:5]	[4:0]
3	12x10x3	[24:13]	[12:8]	[7:5]	[4:0]
4	13x10x3	[25:13]	[12:8]	[7:5]	[4:0]
5	14x10x3	[26:13]	[12:8]	[7:5]	[4:0]
6	15x10x3	[27:13]	[12:8]	[7:5]	[4:0]
7	13x10x2	[24:12]	[11:7]	[6:5]	[4:0]

Notes:

(1) rank_type = 1 and 2 are not supported at memory width = 8 bits and 16 bits.

(2) AMTSEL = 2 is not supported at memory width = 8 bits.

Vangotech

4.6. SCU

4.6.1. Overview

FTSCU100, a system control unit, is designed to provide a power and clock management function for System-on-a-Chip (SoC) to power on the chip, start-up the power, boot-up, frequency change, soft-off (Dormant or deep sleep), and power-off (Shunt-down) the sequences.

FTSCU100 supports the following functionalities:

- Power-on sequence
- Power-control and start-up sequences
- Crystal and PLL controls
- Dynamic frequency change
- CPU speed up or slow down for multiple clusters frequency dynamic change
- BUS speed up or slow down for frequency dynamic change
- System/Peripheral clock control and generation
- Reset signals generation
- Sleep and wake-up controls
- RTC and alarm wake-up
- Operation mode and state machine control
- General-purpose peripheral control register
- Retention memory control
- Extend register interface

The power-on sequence is a power turn-on procedure when the backup battery is attached and a power-on event is active by pressing the power button. In the power start-up sequence, FTSCU100provides an optional function to enable the power according to a programmed sequence.

When the system power is turned on, FTSCU100 will receive the initial values from the strapping pins and then program PLL/DLL to generate the expected clock for the system bus and processor by using the system boot-up sequence. After finishing the PLL/DLL locking procedure, FTSCU100 will de-assert the reset signal in sequence. When SoC completes the boot-up sequence and enters the working state, CPU will start fetching the code to execute the program.

At the working state, FTSCU100 provides several clock management functions to adjust the system operation speed and bus frequency according to the performance and the power requirement. These functions include the frequency change, CPU speed-up/-down, BUS speed-up/-down, and clock gating to provide a safe controlling procedure for the dynamic frequency change. For the power saving, the clocks of the unused blocks can be gated through the register programming at the working state.

To enter the soft-off or power-off state, controlling the system through the software programming, FTSCU100 will trigger a sequence to shut down the system either to keep or drop the memory content according to the application requirement. The retention memory block is used to store the system parameters when the system enters the soft-off state. Some control signals, kept at the soft-off state, can be implemented by the General-Purpose Output (GPO) pins.

Additionally, some applications should not be considered and implemented in FTSCU100 reserves the extensible customized interface for the embedded system integration.

Vangotech

4.6.2. Features

- Supports APB clock rate of up to 100 MHz
- Compliant to AMBA 3.0 APB protocol for register file interface
- Supports four system operating states, working state, sleep state, dormant state, and power-off state, to realize minimum power consumption
- Provides two to eight power control pins and power check in power on/off sequence
- Supports programmable power start-up sequence
- Includes two crystal control signals to stabilize clock for oscillator pad
- PLL/DLL control and clock locking time scheduling
- Provides RTC function and separate RTC power domains to minimize power consumption
- Supports frequency change sequence function to perform dynamic frequency scaling
- Speed-up/Speed-down swiftly bus/processor without resetting PLL
- Programs clock-gating bit by software for devices to reduce power consumption
- Compliant with low-power interface in AMBA Specification, Rev. 3.0
- Provides AXI clock gating to perform AXI low-power function
- Supports sleep mode and wakeup event
- Provides several boot-up events by software programming to boot up system through RTC alarm, other IPs in same power domains to use power button or external reset
- Provides 16 GPO output pins to hold or program desired value by internal register or external GPIO in dormant mode
- Supports 32 strapping pins for hard-wired control settings in boot-up sequence
- Provides retention registers to backup system parameters
- Provides extensible register interface to extend customized registers
- Supports re-boot function at working state
- Supports RTC clock trimming and BCD decode/encode for time/date
- Supports external RTC clock trimming function for temperature compensation
- Supports multi-cluster and multi-core CPU speed and sleep functions
- Supports up to 4 clusters and up to 8 cores for each cluster
- Supports Wait For Event (WFE) and Wait For Interrupt (WFI) selected for CPU idle check
- Supports multi-DDR controller for entering self-refresh before power mode commands
- Programmable software reset function to tune software reset active interval

4.6.3. Operations and Functional Descriptions

4.6.3.1. Block Diagram

Figure 4-19. Functional Block Diagram depicts the designs, including the clock control module (CCM), power control module (PCM), and real time clock module (RTC) to manage the clock and power of a SoC. The clock control module is to implement the dynamic frequency management for the system bus and processor, while CPU and the bus speed can be scaled to reflect the changes of the performance requirements. The power control module handles the power turn-on sequence for the external Low-DropOut (LDO) regulator or the DC/DC converter in the boot-up sequence. The real time clock module is used to monitor the enable event by using the power button or the Real-Time Clock (RTC) alarm

Vangotech

and keep RTC active with very low power consumption.

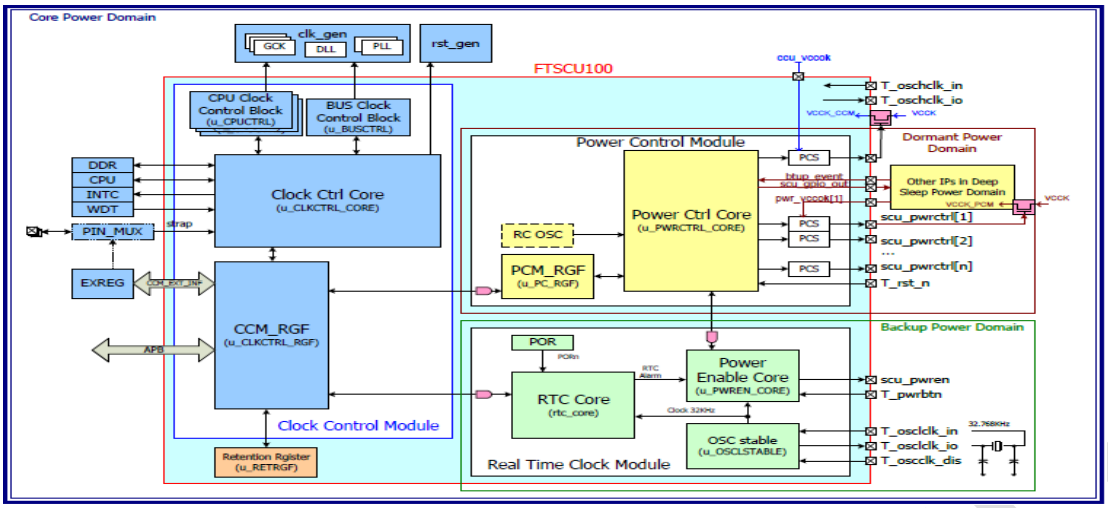


Figure 4-19. Functional Block Diagram

Users can partition one, two, or three power domains to achieve power saving depending on the system requirement. In Figure 4-19, the chip is an example of implementing three power domains: backup, dormant, and core power domains. The detailed information of each module shown in Figure 4-19 is described in the following subsections.

4.6.3.2. Real-Time Clock (RTC) Module

The Real-Time Clock (RTC) module usually is able to separate as an independent power domain, called the backup power domain, and supplied with a backup battery. It includes four main blocks: The RTC core, the power enable core, the Power-On Reset (POR) and the OSC stable block. The RTC core is a real time clock with the calendar function and provides a time-off day alarm to wake up the system when the system is at the power-off state. The power enable core is used to monitor the power button, and trigger an enable event when the button is pressed. The OSC stable block is to stabilize the oscillator pad (OSC PAD) when the backup battery is attached to the RTC module.

The design of the backup power domain uses the high VT cells in the SoC implementation. Because the high VT cell has less leakage power than the low VT cell, it can reduce the static power loss. Because the high VT cells are supplied by the 3.3-V battery power, extra regulator will not be needed to adjust the voltage level for reducing the excessive power consumption from additional components. However, the level shifter and isolation cells are necessary and should be placed between this power domain and other system power domains. All blocks should be in the same power domains.

4.6.3.3. Power Control Module (PCM)

Power Control Module (PCM) is designed to control the external regulator or DC/DC converter and provide a power management machine to handle the start-up and shutdown sequences of the supply power. The power control core can power the start-up sequence according to a programmable sequence when the system receives a boot-up event at the working state. FTSCU100 provides one to eight power control pins to control the internal power switch or the external regulator. This start-up sequence comprises six timeslots to enable each DC-DC converter, LDO voltage regulator, and current limit switch associated with one timeslot. In order to minimize the supply-in-rush current at the power-up time, the start-up of these power control pins should be staggered in time by using this feature.

When power is enabled, the power control core will check if the voltage is ready and releases the isolation

cell used to isolate PCM and the core power domains; and then the system will issue the boot-up sequence of the clock control unit for entering the working state. CPU can issue a power-off or soft-off command through register programming. The power control core will execute the shutdown sequence of the supply power. At the soft-off state, the PCM power is supplied to monitor the boot-up events that come from PCM or the backup power domains; such as the RTC alarm. If users need to boot-up from the external I/O, users can directly connect them to the input port. All boot-up events can be masked by the control registers, except the hardware and watchdog reset signals.

4.6.3.4. Clock Control Module (CCM)

The clock control module handles the boot-up sequence and the bus control sequence. The boot-up sequence is used to control the PLL/DLL settings, enable, and gate the clock for the locking time after powering the start-up sequence. The sequence includes the controlling crystal for the PLL reference clock and the latched strapping value for the hard-wired control setting when the system is powered up from the power-off or dormant state to the working state. All reset signals are sequentially generated when the system boot-up is finished. At the working state, CPU expects to change the operating speed by using Frequency Change Sequence (FCS), that dynamic changes PLL M/N setting value without power off. The boot-up sequence will handle the PLL disabling, setting, and procedures. The bus control sequence, handling the bus behavior, includes the power-down bus, power-down CPU, power-down DDR/SDRAM, speed-up/speed-down processor/bus, and the sleep clock. When CPU issues a command to transit the operation speed or state, the bus control gating sequence should be the power-down request to wait for the acknowledge signals. For example, when a CPU speed is commanded, the bus control sequence will issue the CPU power-down without the power-down DDR and switch to the expected clock and then triggers CPU to exit from the power-down mode. The dynamic frequency scaling can be implemented through both sequences in this unit.

All clock controls are implemented in this module, FTSCU100 provide up to 64-bit clock gating registers in each bus type, which includes APB, AHB, and AXI. If users need more gating bits or other control registers, they can also be implemented through the extend register interface.

4.6.3.5. Power Domain Partition

V8130L with RTC and default power domain, and default domain can be off by soft-off command of SCU. The block diagram of SCU can be referenced by Fig 6-2 about V8130L power domain partition of SCU.

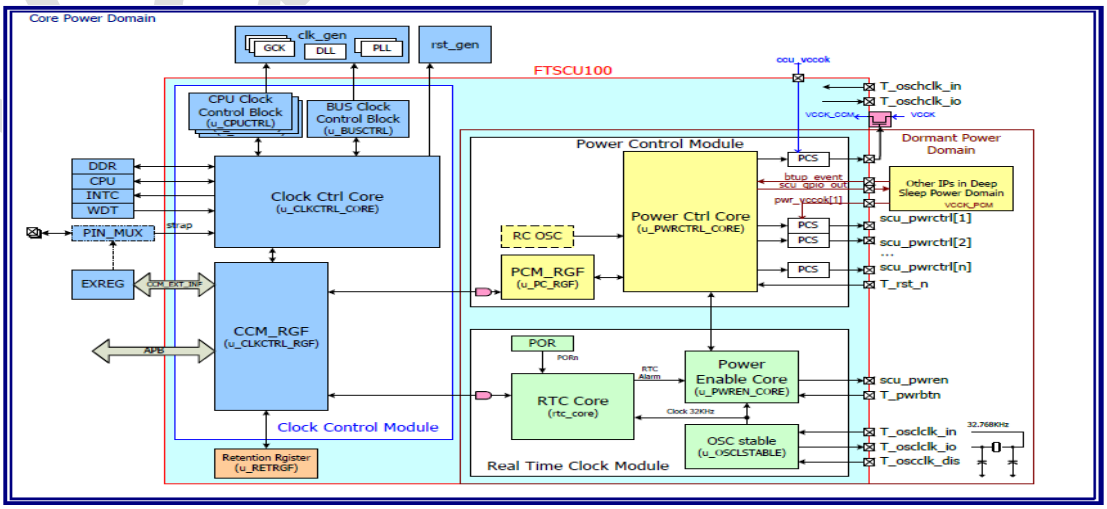


Figure 4-20. V8130L power domain block diagram

The power state of V8130L can be described as Figure 4-21 Power States and Transitions.

There are 14 boot-up event pins for application. The pin list can be refer the 0x2800_0000. User can switch system to G0/G1/G2 power mode and the mapping table are listed in Figure 4-22.

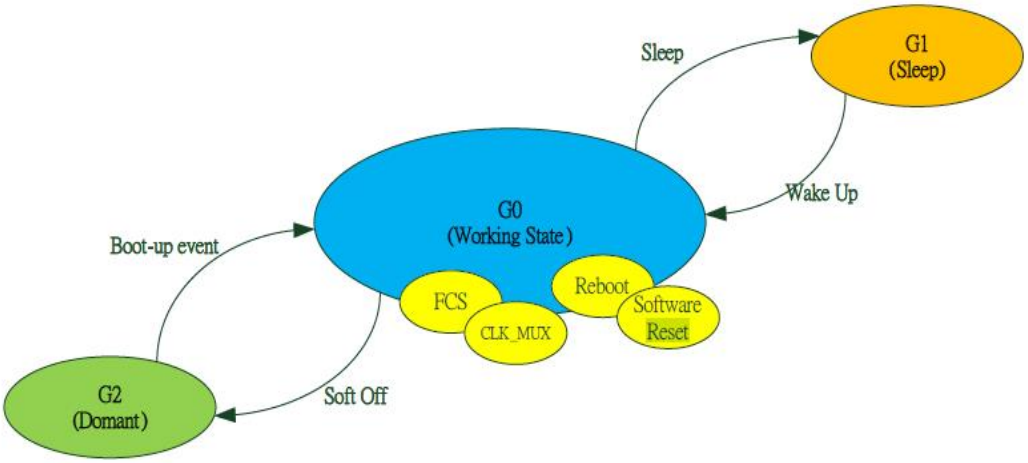


Figure 4-21. Power States and Transitions

	RTC power domain	Default power domain
Working State(G0)	ON	ON
Sleep State(G1)	ON	ON (Clock Gated)
Domant State(G2)	ON	OFF

Figure 4-22. Power mode mapping table

4.6.3.6. SCU Boot Flow

The boot sequence will follow the flow chart in each boot mode. The XTAL of 32KHz will be disabled and replaced by the internal OSC to speed up the boot sequence. The stable waiting time in Figure 4-23 Figure 4-23. SCU Boot **Flow Chart** can be parallel on boot-up. CPU is clocked by the internal OSC (120 MHz) and doesn't need to be blocked by PLL stable waiting.

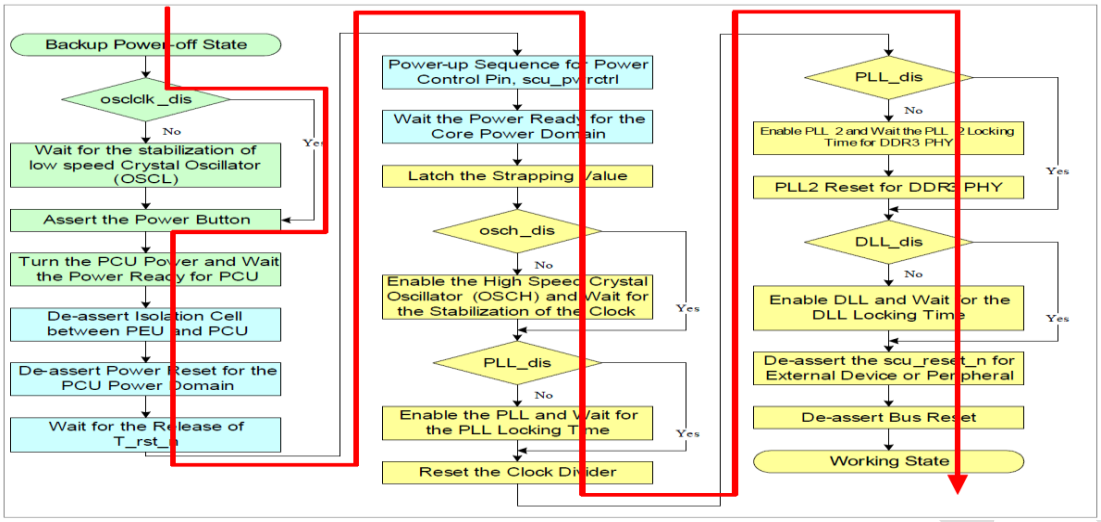


Figure 4-23. SCU Boot Flow Chart

4.6.4. Register List

Module Name	Base Address
SCU	0x2800_0000

Register Name	Offset	Description
BTUP_STS	0x000	Boot-up status register
BTUP_CTRL	0x004	Boot-up control register
PWR_CTRL	0x008	Power control register
CHIPID	0x010	Chip ID register
VERID	0x014	Status control unit version register
STRAP	0x018	Strap value register
OSC_CTRL	0x01C	OSC control register
PWR_MOD	0x020	Power mode register
INT_STS	0x024	Interrupt status register
INT_EN	0x028	Interrupt enable register
SWRST_CTRL	0x02C	Software reset control register
PLL_CTRL	0x030	PLL0 control register
PLL4_CTRL	0x040	PLL4 control register
DLL_CTRL	0x044	DLL control register
AHBCLKG	0x050	AHB clock control register
SLP_AHBCLKG	0x058	AHB clock sleep control register
APBCLKG	0x060~0x064	APB clock control register
SLP_APBCLKG	0x068~x06C	APB clock sleep control register
AXICLKG	0x080	AXI clock gated register
SLP_AXICLKG	0x088	AXI clock sleep control register
SLP_WAKUP_ST	0x0C0	Sleep wakeup events status
SLP_WAKUP_EN	0x0C4	Sleep wakeup events enable

Vangotech

BUS_CLKCTRL_ST	0x0F0	Bus Clock Control Status Register
IDLE_CTRL	0x128	IDLE control register
RTC_TIME1	0x200	RTC timer register 1
RTC_TIME2	0x204	RTC timer register 2
RTC_ALM1	0x208	RTC alarm time register 1
RTC_ALM2	0x20C	RTC alarm time register 2
RTC_CTRL	0x210	RTC control register
RTC_TRIM	0x214	RTC tick trim control register
BCD_RTC_TIME1	0x220	RTC timer register 1 for BCD
BCD_RTC_TIME2	0x224	RTC timer register 2 for BCD
BCD_RTC_ALM1	0x228	RTC alarm time register 1 for BCD
BCD_RTC_ALM2	0x22C	RTC alarm time register 2 for BCD

4.6.5. Register Description

4.6.5.1. Boot-up Status Register

Table 4-161 Boot-up Status Register (BTUP_STS, Offset: 0x000)

Bit	Symbol	Access	Description	Default
31	EXT_BT14_ST	R/WC	Wake up from X_UART5_TX status	0x0
30	EXT_BT13_ST	R/WC	Wake up from X_UART5_RX status	0x0
29	EXT_BT12_ST	R/WC	Wake up from X_UART5_RX_h status	0x0
28	EXT_BT11_ST	R/WC	Wake up from X_UART6_TX status	0x0
27	EXT_BT10_ST	R/WC	Wake up from X_UART6_RX status	0x0
26	EXT_BT9_ST	R/WC	Wake up from X_UART9_TX status	0x0
25	EXT_BT8_ST	R/WC	Wake up from X_UART7_TX status	0x0
24	EXT_BT7_ST	R/WC	Wake up from X_UART7_RX status	0x0
23	EXT_BT6_ST	R/WC	Wake up from X_UART9_RX status	0x0
22	EXT_BT5_ST	R/WC	Wake up from X_UART8_TX status	0x0
21	EXT_BT4_ST	R/WC	Wake up from X_UART8_RX status	0x0
20	EXT_BT3_ST	R/WC	Wake up from X_UART8_RX_h status	0x0
19	EXT_BT2_ST	R/WC	Wake up from X_SPI_DCX1 status	0x0
18	EXT_BT1_ST	R/WC	Wake up from X_SPI_DCX2 status	0x0
17	RTC_BTUP_STS	R/WC	1: Boot up by RTC alarm (Internal or external) events 0: No RTC alarm boot-up event	0x0
16	PWRBTN_STS	R/WC	Boot up by the power button 1: Boot up from the power button 0: No power button even	0x0
15:12	-	-	Reserved	-
11	PMR	R/WC	Boot up from the power-off mode 1: Boot up from the power-off mode 0: No boot-up from the power-off mode	0x0

Vangotech

10	SMR	R/WC	Boot up from the dormant mode 1: Boot-up from the dormant mode 0: No boot-up from the dormant mode	0x0
9	WDR	R/WC	Re-boot by the watchdog reset 1: Watchdog reset will occur. 0: Watchdog reset will not occur.	0x0
8	HWR	R/WC	Re-boot by hardware reset 1: Hardware reset will occur. 0: Hardware reset will not occur.	0x0
7:0	-	-	Reserved	-

4.6.5.2. Boot-up Control Register

Table 4-162 Boot-up Control Register (BTUP_CTRL, Offset: 0x004)

Bit	Symbol	Access	Description	Default
31	EXT_BT14_EN	R/W	Wake up from X_UART5_TX enable	0x1
30	EXT_BT13_EN	R/W	Wake up from X_UART5_RX enable	0x1
29	EXT_BT12_EN	R/W	Wake up from X_UART5_RX_h enable	0x1
28	EXT_BT11_EN	R/W	Wake up from X_UART6_TX enable	0x1
27	EXT_BT10_EN	R/W	Wake up from X_UART6_RX enable	0x1
26	EXT_BT9_EN	R/W	Wake up from X_UART9_TX enable	0x1
25	EXT_BT8_EN	R/W	Wake up from X_UART7_TX enable	0x1
24	EXT_BT7_EN	R/W	Wake up from X_UART7_RX enable	0x1
23	EXT_BT6_EN	R/W	Wake up from X_UART9_RX enable	0x1
22	EXT_BT5_EN	R/W	Wake up from X_UART8_TX enable	0x1
21	EXT_BT4_EN	R/W	Wake up from X_UART8_RX enable	0x1
20	EXT_BT3_EN	R/W	Wake up from X_UART8_RX_h enable	0x1
19	EXT_BT2_EN	R/W	Wake up from X_SPI_DCX1 enable	0x1
18	EXT_BT1_EN	R/W	Wake up from X_SPI_DCX2 enable	0x1
17	RTC_BTUP_EN	R/W	This bit allows the RTC alarm signal to contribute to the boot-up of the power control unit. 1: RTC alarm output is enabled; and transition from low to high on the RTC alarm output will cause a boot-up event to the power control unit. 0: RTC alarm output is disabled; and transition on the RTC alarm output will not cause a wakeup event to the power control unit.	0x1
16	-	-	-	0x1
15:5	GPO_OUT[15:5]	R/W	General-Purpose Output pins	0x0
4:3	GPO[4:3]	R/W	Dcsr[1:0] of X_DDRCK/X_PSW0 Output driving capability 00: 4 mA	0x0

Vangotech

			01: 8 mA 10: 12 mA 11: 16 mA	
2	GPO[2]	R/W	WAKEUP_IN_ENABLE: (Users need to enable before entering the wake-up mode.) 1: Disable the wakeup by external pad 0: Enable the wakeup by external pad	0x1
1	GPO[1]	R/W	RTC_CLK_SEL: 1: Clock from 32KHz XTAL PAD 0: Clock from the internal 32KHz RCOSC	0x0
0	GPO[0]	R/W	DSRONB for DDR	0x1

4.6.5.3. Power Control Register

Table 4-163. Power Control Register (PWR_CTRL, Offset: 0x008)

Bit	Symbol	Access	Description	Default
[31:28]	PWRCTRL_DCSR	R/W	Power pins I/O driving capability For standard I/O, it is allowed to adjust the I/O driving capability. The register provides to set the programmable pins by software. Bit[31]: Slew Rate Bit[30:29]: Output Driving Capability 00 – 2 mA 01 – 4 mA 10 – 6 mA 11 – 8 mA Bit[28]: No support	0x1
27	pwrbtn_offchk	R/W	Power button release check in the power off procedure 1: Check the power button release when entering the power-off or dormant mode. 0: Do not check the power button status when entering the power-off or dormant mode.	0x0
[26:25]	-	-	reserved	-
24	PWRCTRL_UPDATE	R/WS	Power control pins update command The bit is set to directly set the power control pins value by the PWRUP_CTRL, and it will be cleared after finishing the command.	0x0

Vangotech

[23:16]	debounce_mode	R/W	The register used to configure the debounce time for the power button.	0x0A
[15:10]	-	-	-	
[9:8]	PWRUP_CTRL	R/W	Power control pins at the working state [9]: DDR_CKE power on [8]: Default domain on	0x1
[7:2]	-	-	-	
[1:0]	PWRDN_CTRL	R/W	Power control pins at soft-off state [1]: DDR_CKE power down [0]: Default domain down	0x00

4.6.5.4. Power Start-up Sequence Control Register

Table 4-164. *Power Control Register (*PWRUP_SEQ*, Offset: 0x00C)*

Bit	Symbol	Access	Description	Default
[31:16]	Rev	-	-	-
[15:8]	PWRUP_ENSLOT2	R/W	Power Start-up Sequence 2	8'h3
[7:0]	PWRUP_ENSLOT1	R/W	Power Start-up Sequence 1	8'h1

4.6.5.5. Chip ID Register

Table 4-165. Chip ID Register (CHIPID, Offset: 0x010)

Bit	Symbol	Access	Description	Default
[31:0]	CHIP ID	RO or R/PW	Chip ID	0x188B_0000

CHIPID Protected Registers Lock/Unlock

The CHIPID register is protected to avoid unexpected programming. The unlock procedure is shown in Figure 4-24. The unlock ID is the same as CHIPID, and writes the value to the version tag register (0x14). It will unlock the write protected and allow one write operation to the chip ID register (Offset 0x10), and the register will resume the lock state after the write operation. When CHPID is programmed, the unlock key will also be updated to new CHIPID. Users should use new CHIPID to unlock the protected register in the next write procedure.

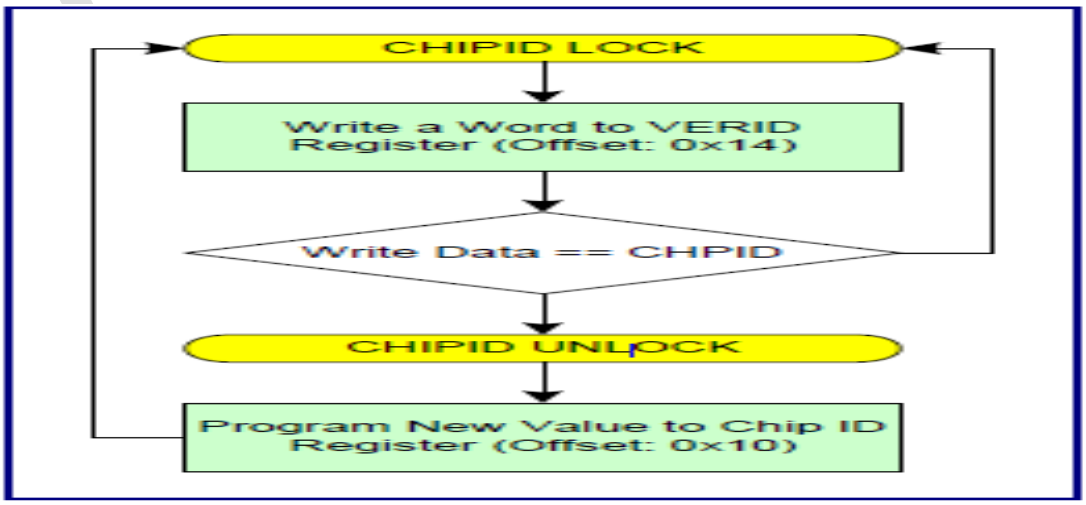


Figure 4-24. CHIPID Protected Register Lock and Unlock

4.6.5.6. System Control Unit Version Register

Table 4-166. System Control Unit Version Register (VERID, Offset: 0x014)

Bit	Symbol	Access	Description	Default
[31:0]	IP VERSION	RO	IP version	0x0003_0100

4.6.5.7. Strap Value Register

Table 4-167. Strap Value Register (STRAP, Offset: 0x018)

Bit	Symbol	Access	Description	Default
17	BOOT_MSG_DIS	RO	1: Disable boot message display 0: Enable boot message display	STRAP
16:15	CPU_JTAG_SEL[1:0]	RO	00: CA7 Debug Only 01: Reserved 10: CA7 daisy chain 11: Disable JTAG function	STRAP
14:13	BOOT_CK_SEL[1:0]	RO	00: From XTAL 01: Internal OSC 30 MHz 10: Internal OSC 60 MHz 11: Internal OSC 120 MHz	STRAP
12	-	-	Reserved.	STRAP
11	EFUSE_CK_SEL	RO	0: eFuse clock from internal OSC 1: eFuse clock from external XTAL	STRAP
10:9	BOOT_SPL_SEL[1:0]	RO	00: SPL is from SPI NOR Flash. 01: SPL is from SPI NAND Flash. 10: SPL is from SPI UART Booting 11: SPL is from eMMC	STRAP
8	BOOT_IPL_SEL	RO	0x0: Boot from ROM (Default) 0x1: Boot from SPI XIP mode	STRAP
-	-	-	-	-
2	DLL_DIS	RO	DLL disable strapping	0x0
1	PLL_DIS	RO	PLL disable strapping	0x0
0	OSCH_DIS	RO	Higher-frequency OSC PAD disable	0x0

4.6.5.8. OSC Control Register

Table 4-168. OSC Control Register (OSC_CTRL, Offset: 0x01C)

Bit	Symbol	Access	Description	Default
[31:4]	-	-	Reserved	0x0
3	OSCHSTABLE	RO	OSCH oscillator stable flag 1: The OSCH oscillator is stable. 0: The OSCH oscillator is not stable.	0x1
2	-	-	-	-
1	OSCHENB	RO	OSCH PAD EB pin	STRAP

Vangotech

			The values depend on strapping DLL_DIS.	
0	OSCHEN	RW	OSCH oscillator enable bit 1: Enable the OSCH oscillator 0: Disable the OSCH oscillator	0x1

4.6.5.9. Power Mode Register

Table 4-169. Power Mode Register, (PWR_MOD, Offset: 0x020)

Bit	Symbol	Access	Description	Default
31	SELFR_CMD_OFF	RW	DDR/SDR self-refresh command off If this bit is set, FTSCU100 will NOT issue the self-refresh command to DDR/SDR in all power modes and frequency change sequences. If system does not have DDR/SDR, please always set to 1'b1.	0x0
30	FCS_PLL2_RSTn	RW	PLL2 resets in the frequency change sequence If the PLL2 output frequency is not satisfied, PLL2 should be reset in the FCS mode. 0: Reset PLL2 1: Keep PLL2 active in FCS	0x0
29	FCS_DLL_RSTn	RW	DLL resets in the frequency change sequence. If the DLL output frequency range is not satisfied, DLL should be reset in the FCS mode. 0: Reset DLL 1: Keep DLL active in FCS	0x0
28	FCS_PLL_RSTn	RW	PLL resets in the frequency change sequence. If the PLL output frequency is not satisfied, PLL should be reset in the FCS mode. 0: Reset PLL	0x0
27:11	-	-	Reserved	-
10	SW_RST	RW	Software reset command The bit is set to issue a reset signal that the interval cab be adjusted by programming SWRST_CTRL (offset 0x02C), and be cleared by hardware after the reset is done.	0x0
9	CURR_RESTORE	RW	Restore current setting frequency in programming register All PLL setting values, including CLKIN_MUX, and the power mode setting values, FCLK_MUX, BUS_MUX, and REMAP will be restored to the PLL control (PLL_CTRL) and power mode (PWR_MOD) registers.	0x0
8	PLL_UPDATE	R/WS	PLL update command PLL setting value is directly updated by the PLL programming register.	0x0

Vangotech

7	REBOOT	R/WS	Reboot command The bit is set to issue a reset to initiate the CPU and bus state. In the combined command, the reset will be de-asserted after another operation sequence is done. However, it is prohibited to perform the sleep command.	0x0
6	FCS	R/WS	Frequency Change Sequence (FCS) The bit is set to enter the frequency change sequence, and will be automatically cleared after finishing FCS. If users want to set PLL by strapping value in next boot-up sequence, the bit should be cleared when setting the soft-off mode command.	0x0
5:4	-	-	Reserved	-
3	SLEEP	R/WS	Sleep-mode sequence This bit will be automatically cleared after waking up. The command does not execute with the soft-off or power-off command.	0x0
2	-	-	Reserved	-
1	SOFTOFF	R/WS	Soft-off command to enter dormant (deep-sleep) mode When the bit is set, the system enters the dormant state, and will be reset when the system comes back from the working state.	0x0
0	PWROFF	R/WS	Power-off mode When the bit is set, the system enters the power-off state, and will be reset when the system is powered on.	0x0

4.6.5.10. Interrupt Status Register

Table 4-170. Interrupt Status Register (INT_STS, Offset: 0x024)

Bits	Field	Type	Description	Default
31:29	-	-	Reserved	0x0
28	INT_PWRSTATE_CHG	R/ WC	Power domain state change status The status indicates that one power state has been changed, and it can be used to check if the power is turned on.	0x0
[27:19]	-	-	Reserved	-
18	INT_RTC_SEC	R/WC	RTC second out interrupt status The field is the value when SCU_RTC_ON is defined.	0x0
17	INT_RTC_PER	R/WC	RTC periodic interrupt status The field is the value when SCU_RTC_ON is defined.	0x0
16	INT_RTC_ALAR	R/WC	RTC alarm interrupt status	0x0

	M		The field is the value when SCU_RTC_ON is defined.	
[15:12]	-	-	Reserved	-
11	INT_REMAP_CHG	R/W	Remap is changed status.	0x0
[10:9]	-	-	Reserved	-
8	INT_PLL_UPDATE	R/WC	PLL update finish interrupt status	0x0
7	-	-	Reserved	
6	INT_FCS	R/WC	FCS command finish interrupt status	0x0
5:4			Reserved	
3	INT_WAKEUP	R/WC	Wake up event status in the sleep mode	0x0
2:0	-	-	Reserved	-

4.6.5.11. Interrupt Enable Register

Table 4-171. Interrupt Enable Register (INT_EN, Offset: 0x028)

Bit	Symbol	Access	Description	Default
31:29	-	-	Reserved	0x0
28	PWRSTATE_CHG_EINT	R/ W	Power domain state change interrupt enable	0x0
27:19	-	-	Reserved	0x0
18	RTC_SEC_EINT	R/W	RTC second out interrupt enable The field is the value when SCU_RTC_ON is defined.	0x0
17	RTC_PER_EINT	R/W	RTC periodic interrupt enable The field is the value when SCU_RTC_ON is defined.	0x0
16	RTC_ALARM_EINT	R/W	RTC alarm interrupt enable The field is the value when SCU_RTC_ON is defined.	0x0
[15:12]	-	-	Reserved	0x0
11	REMAP_CHG_EINT	R/W	REMAP change command enable The bit will be cleared by APB reset, i.e. the remap with reboot command will be disabled, the interrupt never be triggered when issuing the remap function through reboot command.	0x0
[10:9]	-	-	Reserved	0x0
8	PLL_UPDATE_EINT	R/W	PLL update finish interrupt enable	0x1
7	-	-	Reserved	0x0

Vangotech

6	FCS_EINT	R/W	FCS command finish interrupt enable	0x1
5:4	-	-	-	-
3	WAKEUP_EINT	R/W	Wake-up event interrupt enable	0x1
2:0	-	-	Reserved	0x0

4.6.5.12. Software Reset Control Register

Table 4-172. Software Reset Control Register (SWRST_CTRL, Offset: 0x02C)

Bit	Symbol	Access	Description	Default
[31:16]	SWRST_WAIT	R/W	Software reset delay cycles of APB bus clock	0X0
[15:0]	SWRST_ACTIVE	R/W	Software reset active cycles of APB bus clock	0xF

4.6.5.13. PLL0 Control Register

Table 4-173. PLL0 Control Register (PLL_CRTL, Offset: 0x030)

Bit	Symbol	Access	Description	Default
31:24	PLLNS	RW	This field sets the N value of the embedded PLL to control the PLL frequency output.	0x20
23:16	PLLMS	RW	This field sets the M value of the embedded PLL to control the PLL frequency output.	0x1
15:8	PLLFRANG	RW	This field indicates that the output frequency range of the embedded PLL controls the PLL frequency output.	0x3
7:4	CLKIN_MUX	RW	System clock source Mux selection This field changes the system clock source in the FCS mode. When the system clock has many clock sources and users want to change the Mux selection of the clock source without resetting PLL/DLL in the FCS mode, users should set the FCS_PLL_RSTn or FCS_DLL_RSTn pin to disable or un-reset PLL or DLL. 000: 800MHz from PLL0 001: 400MHz from PLL0 010: 200MHz from PLL0 011: 100MHz from PLL0 100: depend on the setting of the bootstrap BOOT_CK_SEL[1:0] Note: The setting will execute after FCS mechanism is complete.	0x4

Vangotech

3:2	-	-	Reserved	0x0
1	PLLSTABLE	RO	PLL locking and stability status 1: PLL is stable. 0: PLL is not stable.	0x0
0	PLLEN	R/W	PLL enable control (PLL0 and PLL4 enable are controlled by this bit!!) The PLL cannot directly be turned ON/OFF when the bit is set. The procedure should accompany a FCS command to stop or enable PLL. (Default enable and disable by user) 1: Enable PLL 0: Disable PLL	0x1

4.6.5.14. PLL4 Control Register

Table 4-174. PLL4 Control Register (PLL4_CTRL, Offset: 0x040)

Bit	Symbol	Access	Description	Default
31:24	PLL4NS	R/W	This field sets the N value of the embedded PLL4 to control the PLL4 frequency output.	0x20
7:0	-	-	Reserved	-

4.6.5.15. DLL Control Register

Table 4-175. DLL Control Register (DLL_CTRL, Offset: 0x044)

Bit	Symbol	Access	Description	Default
31:24	-	-	Reserved	0x0
15:8	DLLFRANG	R/W	This field sets the output frequency range of the embedded DLL to control the DLL frequency output.	0x7
7:2	-	-	Reserved	0x0
1	DLLSTABLE	RO	DLL locking and stability status 1: DLL is stable. 0: DLL is not stable.	0x0
0	DLLEN	R/W	DLL enable control The DLL cannot be directly turned ON/OFF when the bit is set. The procedure should accompany a FCS command to stop or enable DLL. 1: Enable DLL 0: Disable DLL	0x1

4.6.5.16. AHB Clock Control Register

Table 4-176. AHB Clock Control Register0 (AHBCLKG0, Offset: 0x050)

Bit	Symbol	Access	Description	Default
3	spi2ahb_hclk_en	R/W	Hclk enable of SPI2AHB	0x1

Vangotech

2	crypt_hclk_en	R/W	Hclk enable of crypt engine	0x1
1	emmc1_hclk_en	R/W	Hclk enable of eMMC1	0x1
0	emmc0_hclk_en	R/W	Hclk enable of eMMC0	0x1

4.6.5.17. AHB Clock Sleep Control Register

Table 4-177. AHB Clock Sleep Control Register0 (SLP_AHBCLKG0, Offset: 0x058)

Bit	Symbol	Access	Description	Default
3	spi2ahb_hclk_en	R/W	Hclk enable of SPI2AHB	0x1
2	crypt_hclk_en	R/W	Hclk enable of crypt engine	0x1
1	emmc1_hclk_en	R/W	Hclk enable of eMMC1	0x1
0	emmc0_hclk_en	R/W	Hclk enable of eMMC0	0x1

4.6.5.18. APB Clock Control Register

Table 4-178. APB Clock Control Register (APBCLKG0, Offset: 0x060)

Bit	Symbol	Access	Description	Default
31	pwm0_pclk_en	R/W	Pclk enable of PWM TIMER 0	0x1
30	uart4_pclk_en	R/W	Pclk enable of UART_4	0x1
29	uart3_pclk_en	R/W	Pclk enable of UART_3	0x1
28	uart2_pclk_en	R/W	Pclk enable of UART_2	0x1
27	uart1_pclk_en	R/W	Pclk enable of UART_1	0x1
26	uart0_pclk_en	R/W	Pclk enable of UART	0x1
25	ssp3_spi_pclk_en	R/W	Pclk enable of SSP3_SPI	0x1
24	ssp2_spi_pclk_en	R/W	Pclk enable of SSP2_SPI	0x1
23	ssp1_spi_pclk_en	R/W	Pclk enable of SSP1_SPI	0x1
22	ssp0_spi_pclk_en	R/W	Pclk enable of SSP0_SPI	0x1
21	can0_pclk_en	R/W	Pclk enable of CAN0	0x1
20	dmac1_pclk_en	R/W	Pclk enable of DMAC1	0x1
19	wdt3_pclk_en	R/W	Pclk enable of WDT3	0x1
18	wdt2_pclk_en	R/W	Pclk enable of WDT2	0x1
17	tdc_pclk_en	R/W	Pclk enable of TDC	0x1
16	adc_pclk_en	R/W	Pclk enable of ADC	0x1
15	mii_pclk_en	R/W	Pclk enable of MII	0x1
14	rmii_pclk_en	R/W	Pclk enable of RMII	0x1
13	rgmii_pclk_en	R/W	Pclk enable of RGMII	0x1
12	gpio_pclk_en	R/W	Pclk enable of GPIO	0x1
11	emc_sram_pclk_en	R/W	Pclk enable of EMC_SRAM	0x1
10	emc_rom_pclk_en	R/W	Pclk enable of EMC_ROM	0x1
9	ddr_pclk_en	R/W	Pclk enable of DDR	0x1
8	tzc_bus_pclk_en	R/W	Pclk enable of TZC for bus trust	0x1
7	tzc_lcdc_pclk_en	R/W	Pclk enable of TZC for LCDC trust	0x1

vangotech

6	tzc_ca7_pclk_en	R/W	Pclk enable of TZC for CA7 bus trust	0x1
5	-	-	-	0x1
4	dmac0_pclk_en	R/W	Pclk enable of DMAC0	0x1
3	trng_pclk_en	R/W	Pclk enable of TRNG	0x1
2	efuse_pclk_en	R/W	Pclk enable of efuse	0x1
1	wdt1_pclk_en	R/W	Pclk enable of WDT1	0x1
0	wdt0_pclk_en	R/W	Pclk enable of WDT0	0x1

Table 4-179. APB Clock Control Register (APBCLKG1, Offset: 0x0000_0064)

Bit	Symbol	Access	Description	Default
31:25	-	-	-	-
24	ssp7_spi_pclk_en	R/W	Pclk enable of SSP7_SPI	0x1
23	-	-	-	-
22	ssp1_i2s_pclk_en	R/W	Pclk enable of SSP1_I2S	0x1
21	-	R/W	-	0x1
20	i2c4_pclk_en	R/W	Pclk enable of I2C4	0x1
19	i2c3_pclk_en	R/W	Pclk enable of I2C3	0x1
18	pwm3_pclk_en	R/W	Pclk enable of PWM TIMER 3	0x1
17	pwm2_pclk_en	R/W	Pclk enable of PWM TIMER 2	0x1
16	uart9_pclk_en	R/W	Pclk enable of UART9	0x1
15	uart8_pclk_en	R/W	Pclk enable of UART8	0x1
14	uart7_pclk_en	R/W	Pclk enable of UART7	0x1
13	uart6_pclk_en	R/W	Pclk enable of UART6	0x1
12	uart5_pclk_en	R/W	Pclk enable of UART5	0x1
11	ssp6_spi_pclk_en	R/W	Pclk enable of SSP6_SPI	0x1
10	ssp5_spi_pclk_en	R/W	Pclk enable of SSP5_SPI	0x1
9	ssp4_spi_pclk_en	R/W	Pclk enable of SSP4_SPI	0x1
8	can1_pclk_en	R/W	Pclk enable of CAN1	0x1
7	lcdc_pclk_en	R/W	Pclk enable of LCDC controller	0x1
6	kbc_pclk_en	R/W	Pclk enable of KBC	0x1
5	gpio1_pclk_en	R/W	Pclk enable of GPIO1	0x1
4	ssp0_i2s_pclk_en	R/W	Pclk enable of SSP0_I2S	0x1
3	i2c2_pclk_en	R/W	Pclk enable of I2C2	0x1
2	i2c1_pclk_en	R/W	Pclk enable of I2C1	0x1
1	i2c0_pclk_en	R/W	Pclk enable of I2C0	0x1
0	pwm1_pclk_en	R/W	Pclk enable of PWM TIMER 1	0x1

4.6.5.19. APB Clock Sleep Control Register

Table 4-180. APB Clock Sleep Control Register (SLP_APBCLKG0, Offset: 0x068)

Bit	Symbol	Access	Description	Default

Vangotech

31	pwm0_pclk_en	R/W	Pclk enable of PWM TIMER 0	0x1
30	uart4_pclk_en	R/W	Pclk emable of UART_4	0x1
29	uart3_pclk_en	R/W	Pclk emable of UART_3	0x1
28	uart2_pclk_en	R/W	Pclk emable of UART_2	0x1
27	uart1_pclk_en	R/W	Pclk emable of UART_1	0x1
26	uart0_pclk_en	R/W	Pclk emable of UART	0x1
25	ssp3_spi_pclk_en	R/W	Pclk enable of SSP3_SPI	0x1
24	ssp2_spi_pclk_en	R/W	Pclk enable of SSP2_SPI	0x1
23	ssp1_spi_pclk_en	R/W	Pclk enable of SSP1_SPI	0x1
22	ssp0_spi_pclk_en	R/W	Pclk enable of SSP0_SPI	0x1
21	can0_pclk_en	R/W	Pclk enable of CAN0	0x1
20	dmac1_pclk_en	R/W	Pclk enable of DMAC1	0x1
19	wdt3_pclk_en	R/W	Pclk enable of WDT3	0x1
18	wdt2_pclk_en	R/W	Pclk enable of WDT2	0x1
17	tdc_pclk_en	R/W	Pclk enable of TDC	0x1
16	adc_pclk_en	R/W	Pclk enable of ADC	0x1
15	mii_pclk_en	R/W	Pclk enable of MII	0x1
14	rmii_pclk_en	R/W	Pclk enable of RMII	0x1
13	rgmii_pclk_en	R/W	Pclk enable of RGMII	0x1
12	gpio_pclk_en	R/W	Pclk enable of GPIO	0x1
11	emc_sram_pclk_en	R/W	Pclk enable of EMC_SRAM	0x1
10	emc_rom_pclk_en	R/W	Pclk enable of EMC_ROM	0x1
9	ddr_pclk_en	R/W	Pclk enable of DDR	0x1
8	tzc_bus_pclk_en	R/W	Pclk enable of TZC for bus trust	0x1
7	tzc_lcdc_pclk_en	R/W	Pclk enable of TZC for LCDC trust	0x1
6	tzc_ca7_pclk_en	R/W	Pclk enable of TZC for CA7 bus trust	0x1
5	-	-	-	-
4	dmac0_pclk_en	R/W	Pclk enable of DMAC0	0x1
3	trng_pclk_en	R/W	Pclk enable of TRNG	0x1
2	efuse_pclk_en	R/W	Pclk enable of efuse	0x1
1	wdt1_pclk_en	R/W	Pclk enable of WDT1	0x1
0	wdt0_pclk_en	R/W	Pclk enable of WDT0	0x1

Table 4-181. APB Clock Sleep Control Register (SLP_APBCLKG1, Offset: 0x06C)

Bit	Symbol	Access	Description	Default
24	ssp7_spi_pclk_en	R/W	Pclk enable of SSP7_SPI	0x1
23	-	-	-	-
22	ssp1_i2s_pclk_en	R/W	Pclk enable of SSP1_I2S	0x1
21	-	R/W	-	0x1
20	i2c4_pclk_en	R/W	Pclk enable of I2C4	0x1
19	i2c3_pclk_en	R/W	Pclk enable of I2C3	0x1

Vangotech

18	pwm3_pclk_en	R/W	Pclk enable of PWM TIMER 3	0x1
17	pwm2_pclk_en	R/W	Pclk enable of PWM TIMER 2	0x1
16	uart9_pclk_en	R/W	Pclk enable of UART9	0x1
15	uart8_pclk_en	R/W	Pclk enable of UART8	0x1
14	uart7_pclk_en	R/W	Pclk enable of UART7	0x1
13	uart6_pclk_en	R/W	Pclk enable of UART6	0x1
12	uart5_pclk_en	R/W	Pclk enable of UART5	0x1
11	ssp6_spi_pclk_en	R/W	Pclk enable of SSP6_SPI	0x1
10	ssp5_spi_pclk_en	R/W	Pclk enable of SSP5_SPI	0x1
9	ssp4_spi_pclk_en	R/W	Pclk enable of SSP4_SPI	0x1
8	can1_pclk_en	R/W	Pclk enable of CAN1	0x1
7	lcdc_pclk_en	R/W	Pclk enable of LCDC controller	0x1
6	kbc_pclk_en	R/W	Pclk enable of KBC	0x1
5	gpio1_pclk_en	R/W	Pclk enable of GPIO1	0x1
4	ssp0_i2s_pclk_en	R/W	Pclk enable of SSP0_I2S	0x1
3	i2c2_pclk_en	R/W	Pclk enable of I2C2	0x1
2	i2c1_pclk_en	R/W	Pclk enable of I2C1	0x1
1	i2c0_pclk_en	R/W	Pclk enable of I2C0	0x1
0	pwm1_pclk_en	R/W	Pclk enable of PWM TIMER 1	0x1

4.6.5.20. AXI Clock Control Register

Table 4-182. AXI Clock Control Register0 (AXICLKG, Offset: 0x080)

Bit	Symbol	Access	Description	Default
11	u2_otg2_aclk_en	R/W	Aclk enable of U2_2	0x1
10	lc_aclk_en	R/W	Aclk enable of LCDC controller	0x1
9	mii_aclk_en	R/W	Aclk enable of RMII_1	0x1
8	rmii_aclk_en	R/W	Aclk enable of RMII	0x1
7	rgmii_aclk_en	R/W	Aclk enable of RGMII_1	0x1
6	u2_otg1_aclk_en	R/W	Aclk enable of U2_1	0x1
5	u2_otg0_aclk_en	R/W	Aclk enable of U2	0x1
4	dmac1_aclk_en	R/W	Aclk enable of DMAC1	0x1
3	dmac0_aclk_en	R/W	Aclk enable of DMAC0	0x1
2	ddr0_ch2_aclk_en	R/W	Aclk enable of DDR CH2	0x1
1	ddr0_ch1_aclk_en	R/W	Aclk enable of DDR CH1	0x1
0	ddr0_ch0_aclk_en	R/W	Aclk enable of DDR CH0	0x1

4.6.5.21. AXI Clock Sleep Control Register

Table 4-183. AXI Clock Sleep Control Register0 (AXICLK_SLPCTRL, Offset: 0x088)

Bit	Symbol	Access	Description	Default
11	u2_otg2_aclk_en	R/W	Aclk enable of U2_2	0x1

Vangotech

10	lc_aclk_en	R/W	Aclk enable of LCDC controller	0x1
9	mii_aclk_en	R/W	Aclk enable of RMII_1	0x1
8	rmii_aclk_en	R/W	Aclk enable of RMII	0x1
7	rgmii_aclk_en	R/W	Aclk enable of RGMII_1	0x1
6	u2_otg1_aclk_en	R/W	Aclk enable of U2_1	0x1
5	u2_otg0_aclk_en	R/W	Aclk enable of U2	0x1
4	dmac1_aclk_en	R/W	Aclk enable of DMAC1	0x1
3	dmac0_aclk_en	R/W	Aclk enable of DMAC0	0x1
2	ddr0_ch2_aclk_en	R/W	Aclk enable of DDR CH2	0x1
1	ddr0_ch1_aclk_en	R/W	Aclk enable of DDR CH1	0x1
0	ddr0_ch0_aclk_en	R/W	Aclk enable of DDR CH0	0x1

4.6.5.22. Multi-Core WFI/WFE Control Register

Table 4-184. Multi-Core WFI/WFE Control Register (0x0B4)

Bit	Symbol	Access	Description	Default
7:0	CPU0_WFI_MSK	R/W	Specify checking by standby_wfi or standby_wfe when executing the SCU commands for each core of Cluster 0 CPU. Bit0: Core 0 of Cluster 0 CPU Bit1: Core 1 of Cluster 0 CPU Bit2: Core 2 of Cluster 0 CPU Bit3: Core 3 of Cluster 0 CPU The programmable bit content is 1: Checked by standby_wfe 0: Checked by standby_wfi (If users want to trigger by WFE, it needs to enable EVENTI_EN bit on 0x2800_8090[28].)	0x0

4.6.5.23. Wakeup Event Status Register

Table 4-185. Sleep Wakeup Status Register (SLP_WAKUP_ST, Offset: 0x0C0)

Bit	Symbol	Access	Description	Default
14	EXT_BT14_ST	R/W	Wake up from X_UART5_TX events	0x0
13	EXT_BT13_ST	R/W	Wake up from X_UART5_RX events	0x0
12	EXT_BT12_ST	R/W	Wake up from X_UART5_RX_h events	0x0
11	EXT_BT11_ST	R/W	Wake up from X_UART6_TX events	0x0
10	EXT_BT10_ST	R/W	Wake up from X_UART6_RX events	0x0
9	EXT_BT9_ST	R/W	Wake up from X_UART9_TX events	0x0
8	EXT_BT8_ST	R/W	Wake up from X_UART7_TX events	0x0
7	EXT_BT7_ST	R/W	Wake up from X_UART7_RX events	0x0
6	EXT_BT6_ST	R/W	Wake up from X_UART9_RX events	0x0
5	EXT_BT5_ST	R/W	Wake up from X_UART8_TX events	0x0

Vangotech

4	EXT_BT4_ST	R/W	Wake up from X_UART8_RX events	0x0
3	EXT_BT3_ST	R/W	Wake up from X_UART8_RX_h events	0x0
2	EXT_BT2_ST	R/W	Wake up from X_SPI_DCX1 events	0x0
1	EXT_BT1_ST	R/W	Wake up from X_SPI_DCX2 events	0x0
0	SLP_WAKE_ST0	R/W1C	Wakeup from sleep by the boot-up events	0x0

4.6.5.24. Wakeup Event Status Register

Table 4-186. Sleep Wakeup Enable Register (SLP_WAKUP_EN, Offset: 0x0C4)

Bit	Symbol	Access	Description	Default
14	EXT_BT14_EN	R/W	Wake up enable from X_UART5_TX events	0x0
13	EXT_BT13_EN	R/W	Wake up enable from X_UART5_RX events	0x0
12	EXT_BT12_EN	R/W	Wake up enable from X_UART5_RX_h events	0x0
11	EXT_BT11_EN	R/W	Wake up enable from X_UART6_TX events	0x0
10	EXT_BT10_EN	R/W	Wake up enable from X_UART6_RX events	0x0
9	EXT_BT9_EN	R/W	Wake up enable from X_UART9_TX events	0x0
8	EXT_BT8_EN	R/W	Wake up enable from X_UART7_TX events	0x0
7	EXT_BT7_EN	R/W	Wake up enable from X_UART7_RX events	0x0
6	EXT_BT6_SEN	R/W	Wake up enable from X_UART9_RX events	0x0
5	EXT_BT5_EN	R/W	Wake up enable from X_UART8_TX events	0x0
4	EXT_BT4_EN	R/W	Wake up enable from X_UART8_RX events	0x0
3	EXT_BT3_EN	R/W	Wake up enable from X_UART8_RX_h events	0x0
2	EXT_BT2_EN	R/W	Wake up enable from X_SPI_DCX1 events	0x0
1	EXT_BT1_EN	R/W	Wake up enable from X_SPI_DCX2 events	0x0
0	SLP_WAKE_ST0	R/W1C	Wakeup enable for the boot-up events	0x0

4.6.5.25. IDLE Control Register

Table 4-187. IDLE Control Register(IDLE_CTRL, Offset: 0x128)

Bit	Symbol	Access	Description	Default
12	IDLE_CHK_EN	RW	When executing FCS, BUS SPEED or CPU SPEED command in PWR_MOD(0x20) register, to return to normal state if SCU can't receive the IDLE bit from CPU in certain waiting cycles 0x1: Enable 0x0: Disable	1'h0
11:0	IDLE_CHK_CNT	RW	To define the number of waiting cycles if SCU can't receive the IDLE bit from CPU	12'hFFF

Vangotech

4.6.5.26. Real-Time Clock (RTC) Function

FTSCU100 contains a Real-Time Clock (RTC) that can be controlled by using the hardware parameter, SCU_RTC_ON. RTC maintains the current date and time, and is able to generate the alarm and periodic interrupt or boot-up event by using the alarm in the soft-off mode.

RTC is usually partitioned to the 3.3-V power domain with the real time clock unit that is supplied by the backup power to keep running when the normal power sources are unavailable. RTC needs an accurate 32.768 KHz clock that is provided from the external crystal oscillator.

4.6.5.27. RTC Time Register

Register Description:

The current time can be read from two time registers defined as below. These time registers are updated every second. In the APB protocol, it only simultaneously issues a single register read and executes at an arbitrary, so it does not guarantee the accurate time reading, because the second register is possible to be changed after the first register reading. Three possible methods are recommended for reliable reading of the time registers:

- Read after interrupt: The INT_RTC_SEC interrupt indicates that the second counter has just been incremented, and the RTC registers will not change before the next second. A register read will immediately be executed after issuing an INT_RTC_SEC interrupt; therefore, it is considered as an accurate time reading.
- Two consecutive reads: If two consecutive reads within a short time (Less than 1 second apart) returns the same result, this is considered as the accurate reading. If the two results are different, the procedure should be repeated.
- Using the Time Lock function: FTSCU100 implements a lock function in RTC. Users can lock the current time before reading the register. To enable bit 2 of the RTC control register (0x210), the current time should be locked for an accurate reading of the time1 register and can be released by one of the following three conditions:

  Time2 register is read.

  Time1 or Time2 register is written.

  RTC is disabled.

If users want to set new RTC time, the current time register has to be immediately written and then the RTC_EN bit of the RTC control register should be set. When the setting values are reliably written to the RTC counter, the RTCEN_STS bit will be set and the RTC counter will start running. Before writing the RTC time register, RTC should be stopped to prevent glitches. The procedure is as follows:

1. Clear the RTC_EN bit of the RTC control register to stop the RTC counter
2. Check and read the RTCEN_STS bit of the RTC control register to confirm if RTC is disabled.
3. Set new time to registers Time1 and Time2
4. Set the RTC_EN bit and wait until the RTCEN_STS bit is changed to '1' and the RTC counter will be re-enabled.

In the time setting procedure, the written values of the current time register will guarantee a valid time for each field by the software programming. For example, the content in second and minute must be between 0 and 59.

Table 4-188. RTC Time1 Register (RTC_TIME1, Offset: 0x200)

Bit	Symbol	Access	Description	Default
[31:27]	-	-	Reserved	-
[26:24]	RTC_WEEKDAY	R/W	Current weekday The valid value is ranging from 0 to 6. 0: Sunday 1: Monday 2: Tuesday 3: Wednesday 4: Thursday 5: Friday 6: Saturday	0x0
[23:22]	-	-	Reserved	-
[21:16]	RTC_HOUR	R/W	Current hour The valid value is ranging from 0 to 23.	0x0
15	-	-	Reserved	-
[14:8]	RTC_MIN	R/W	Current minute The valid value is ranging from 0 to 59.	0x0
7	-	-	Reserved	-
[6:0]	RTC_SEC	R/W	Current second The valid value is ranging from 0 to 59.	0x0

Table 4-189. RTC Time2 Register (RTC_TIME2, Offset: 0x204)

Bit	Symbol	Access	Description	Default
[31:24]	RTC_CENTURY	R/W	Current century The valid value is ranging from 0 to 99.	0x14
[23:16]	RTC_YEAR	R/W	Current year The valid value is ranging from 0 to 99.	0xA
[15:13]	-	-	Reserved	-
[12:8]	RTC_MONTH	R/W	Current month The valid value is ranging from 1 to 12.	0x1
[7:6]	-	-	Reserved	-
[5:0]	RTC_DATE	R/W	Current day The valid value is ranging from 1 to 31.	0x1

4.6.5.28. RTC Alarm Time Register

Register Description:

Two alarm registers are used to set the time-of-day/date alarm time. The alarm can be programmed by the alarm enable (ALARM_EN) of the control register (Offset 0x210) to activate the wake event on an alarm match condition when the chip is in the soft-off mode. The alarm also can issue an interrupt when the system is at the working state. There are six comparing fields for the RTC alarm, which are described in Table 4-190 and Table 4-191. Users can individually disable the time match checking in each field. For

Vangotech

example, ALM_WEEKDAY is set to '1', and the alarm will not check if the field matches the current time. For setting the alarm, the procedure is similar to that of being used when setting the RTC current time. Users should disable the alarm, clear the ALARM_EN bit of the RTC control register, and prevent the accidental alarm event from being triggered:

1. Clear the ALARM_EN bit of the RTC control register to disable the RTC alarm
2. Check and read the RTC_ALM_STS bit of the RTC control register to confirm if RTC alarm is disabled.
3. Set new time to registers, Alarm Time1 and Alarm Time2
4. Set the ALARM_EN bit and wait until the RTC_ALM_STS bit is changed to '1' and the setting of the RTC alarm is complete.

Table 4-190. RTC Alarm Time1 Register (RTC_ALM1, Offset: 0x208)

Bit	Symbol	Access	Description	Default
[31:27]	-	-	Reserved	-
[26:24]	ALM_WEEKDAY	R/W	Alarm weekday The valid value is ranging from 0 to 6. When all bits are set to '1', this field will be "don't care" in the alarm matching.	0x7
[23:22]	-	-	Reserved	-
[21:16]	ALM_HOUR	R/W	Alarm hour The valid value is ranging from 0 to 23. When all bits are set to '1', this field will be "don't care" in the alarm matching.	0x1F
15	-	-	Reserved	-
[14:8]	ALM_MIN	R/W	Alarm minute The valid value is ranging from 0 to 59. When all bits are set to '1', this field will be "don't care" in the alarm matching.	0x3F
7	-	-	Reserved	-
[6:0]	ALM_SEC	R/W	Alarm second The valid value is ranging from 0 to 59. When all bits are set to '1', this field will be "don't care" in the alarm matching.	0x3F

Table 4-191. RTC Alarm Time2 Register (RTC_ALM2, Offset: 0x20C)

Bit	Symbol	Access	Description	Default
[31:13]	-	-	Reserved	-
[12:8]	ALM_MONTH	R/W	Alarm Month, the valid value is ranging from 1 to 12. When all bits are set to '0', the field is "don't care" in alarm matching.	0x0
[7:6]	-	-	Reserved	-
[5:0]	ALM_DATE	R/W	Alarm Date, the valid value is ranging from 1 to 31. When all bits are set to '0', the field is "don't	0x0

Vangotech

			care" in alarm matching.	

4.6.5.29. RTC Control Register

Register Description:

The internal RTC of FTSCU100 provides extra two interrupts, INT_RTC_SEC and INT_RTC_PER, to trigger the periodic events. The INT_RTC_SEC interrupt will raise an approximate 30.5-μs pulse on every second when the SECOUT_EN bit of the RTC control register is set. Another periodic interrupt, INT_RTC_PER, is configurable with a frequency determined by the PERINT_SEL field of the RTC control register as defined in Table 4-192.

Table 4-192. RTC Control Register (RTC_CTRL, Offset: 0x210)

Bit	Symbol	Access	Description	Default
31	OSC_PAD_FEB	R/W	OSCL crystal pad FEB control	0x0
[30:24]	-	-	Reserved	-
[23:16]	RTC_GPO	R/W	RTC GPO output value The bit number is defined by the parameter, SCU_RTCGPO_N.	0x0
[15:12	-	-	Reserved	-
11	RTC_CLK_READY	R	RTC clock stable status	0x0
9	RTC_ALM_STS	RO	RTC alarm status The bit will be set when RTC accepts the command and is cleared when RTC_ALARM_EN is disabled.	0x0
8	RTCEN_STS	RO	RTC enable status The bit will be set when RTC accepts the command and is cleared when RTC_EN is disabled.	0x0
7	SECOUT_EN	R/W	Enable an event in each second	0x0
[6:4]	PERINT_SEL	R/W	Periodic interrupt output signal select 3'b111: Each second 3'b110: Each minute and triggered at 0 second of every minute 3'b101: Each hour and triggered at 0 second of every hour. 3'b100: Each day and triggered at 0 second of every day 3'b011: Each month and triggered at 0 second of every month Others: Disable	0x0
3	-	-	Reserved	-
2	LOCK_EN	R/W	Lock the Time2 register after reading the Time1 register, the lock will be released by	0x0

Vangotech

			the following conditions: 1. Time2 is read. 2. Time1 or Time2 is written, 3. RTC is disabled.	
1	ALARM_EN	R/W	RTC alarm enable	0x0
0	RTC_EN	R/W	RTC enable The RTC clock should be confirmed stable through monitoring bit 2 of the OSC control register (0x1C) before turning on RTC. If SCU_RTCDEF_ON is configured, RTC will be turned on after power start-up. Otherwise, RTC is OFF, and will start through software programming.	0x0

4.6.5.30. RTC Tick Trim Control Register

Register Description:

RTC includes a tick generation block to provide precise tick for counting the time in each second. The tick can be trimmed by the RTC tick trim control register to adjust the counting value of the tick generation block and compensate the frequency drift caused by the crystal inaccuracies. The tick trim is fractional divider, the TRIM_P field is the fixed point part, and the TRIM_F field is the floating point part in Table 4-193. The tick frequency equation is shown below.

Tick = TRTC_OSC/ (P + (F/1000)) KHz where, P is TRIM_P and F is TRIM_F.

TRTC_OSC is the period of RTC clock of RTC, and it is usually from an external clock source of 32.768 KHz crystal. The tick should be tuned to 100 Hz when the SCU_SYSTICK_ON parameter is defined. Otherwise, the tick must to set 1 Hz.

The tick trim control register is initiated by the configuration parameter, SCU_RTCTRIM_DEF, and the write access is protected by the security function.

Table 4-193. RTC Tick Trim Control Register (RTC_TRIM, Offset: 0x214)

Bit	Symbol	Access	Description	Default
[31:16]	TRIM_P	R/PW	RTC tick dividing factor	0x0000
[15:0]	TRIM_F	R/PW	RTC tick fractional factor The register is allowed to program a value between 0 and 999.	0x8000

RTC Clock Trimming Protected Registers Lock/Unlock

The RTC trimming register does not allow to arbitrarily changing, it is protected to avoid the unexpected programming. The un-lock procedure is shown in Figure 4-25. The unlock ID is the same CHIPID, and writes the value to the version tag register (0x14), it will unlock the write protected and allow once write operation to the RTC clock trimming register (0x214), the register will resume the lock state after the write operation. When CHPID is programmed, the unlock key will also be updated to new CHIPID. Users

Vangotech

should use new CHIPID to unlock the protected register in the next write procedure.

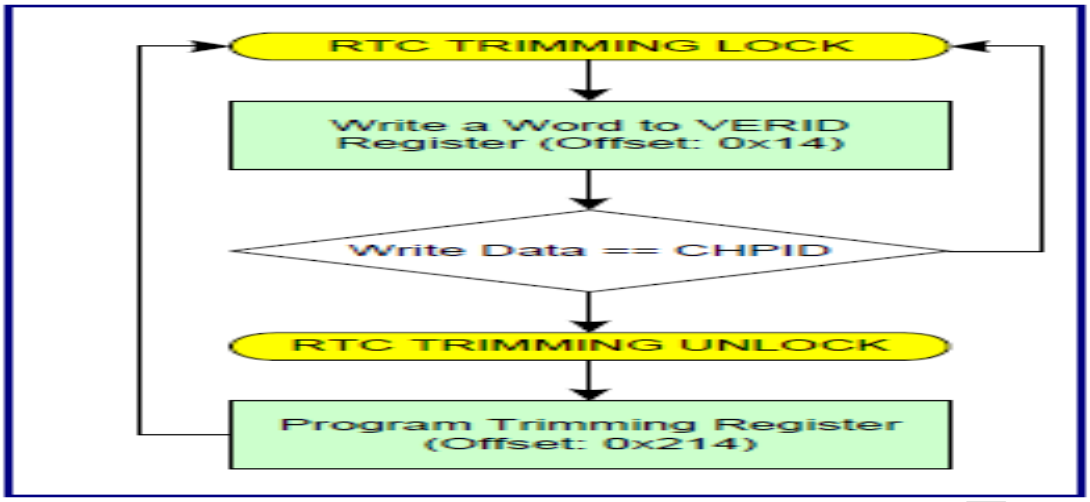


Figure 4-25. RTC Trimming Protected Register Lock and Unlock

4.6.5.31. RTC Time BCD Register

Register Description:

When the parameter, SCU_RTCBCD_ON, is defined, both the decoding and encoding circuits for BCD are included to set or read the RTC current time/date.

Table 4-194. RTC Time1 BCD Register (BCD_RTC_TIME1, Offset: 0x220)

Bit	Symbol	Access	Description	Default
[31:27]	-	-	Reserved	-
[26:24]	BCD_RTC_WEEKDAY	R/W	Current weekday The valid value is ranging from 0 to 6. 0: Sunday 1: Monday 2: Tuesday 3: Wednesday 4: Thursday 5: Friday 6: Saturday The valid programmed value is ranging from 0 to 6.	0x0
[23:20]	BCD_RTC_HOUR_DEC	R/W	Current hour in decimal and unit values The decimal valid value is ranging from 0 to 2, and the unit valid value is ranging from 0 to 9. The legal programmed value is ranging from 0 to 0x23.	0x0
[19:16]	BCD_RTC_HOUR	R/W		0x0
[15:12]	BCD_RTC_MIN_DEC	R/W	Current minute in decimal and unit values The decimal valid value is ranging from	0x0
11:8]	BCD_RTC_MIN	R/W		0x0

			0 to 5, and the unit valid value is ranging from 0 to 9. The legal programmed value is ranging from 0 to 0x59.	
[7:4]	BCD_RTC_SEC_DEC	R/W	Current second in decimal and unit values The decimal valid value is ranging from 0 to 5, and the unit valid value is ranging from 0 to 9. The legal programmed value is ranging from 0 to 0x59.	0x0
[3:0]	BCD_RTC_SEC	R/W		0x0

Table 4-195. RTC Time2 BCD Register (BCD_RTC_TIME2, Offset: 0x224)

Bit	Symbol	Access	Description	Default
[31:28]	BCD_RTC_CENTURY_DEC	R/W	Current century in decimal and unit values The decimal valid value is ranging from 0 to 9, and the unit valid value is ranging from 0 to 9.	0x2
[27:24]	BCD_RTC_CENTURY	R/W		0x0
[23:20]	BCD_RTC_YEAR_DEC	R/W	Current year in decimal and unit values The decimal valid value is ranging from 0 to 9, and the unit valid value is ranging from 0 to 9.	0x1
[19:16]	BCD_RTC_YEAR	R/W		0x0
15:12]	BCD_RTC_MONTH_DEC	R/W	Current month in decimal and unit values The decimal valid value is ranging from 0 to 1, and the unit valid value is ranging from 1 to 9. The legal programmed value is ranging from 0 to 0x59.	0x0
[11:8]	BCD_RTC_MONTH	R/W		0x1
[7:4]	BCD_RTC_DATE_DEC	R/W	Current day in decimal and unit values The decimal valid value is ranging from 0 to 3, and the unit valid value is ranging from 1 to 9.	0x0
[3:0]	BCD_RTC_DATE	R/W		0x1

4.6.5.32. RTC Alarm BCD Register

Register Description:

As the same as BCD_RTC_TIME1 and BCD_RTC_TIME2, both BCD_RTC_ALM1 and BCD_RTC_ALM2 are used to set the RTC alarm by BCD encoding for the RTC alarm.

Table 4-196. RTC Alarm Time1 BCD Register (BCD_RTC_ALM1, Offset: 0x228)

Bit	Symbol	Access	Description	Default
[31:27]	-	-	Reserved	-
[26:24]	BCD_ALM_WEEKDAY	R/W	Alarm weekday setting by BCD The valid value is ranging from 0 to 6, and set 7 to disable the weekday setting for the RTC alarm.	0x7
[23:20]	BCD_ALM_HOUR_DEC	R/W	Alarm hour setting by BCD in decimal and unit values The decimal valid value is ranging from 0 to 2, and the unit valid value is ranging from 0 to 9. The legal programmed value is up to 0x23, and set to 0x31 to disable the hour setting for the RTC alarm.	0x3
[19:16]	BCD_ALM_HOUR	R/W		0x1
[15:12]	BCD_ALM_MIN_DEC	R/W	Alarm minute setting by BCD in decimal and unit values The decimal valid value is ranging from 0 to 5, and the unit valid value is ranging from 0 to 9. The legal programmed value is up to 0x59, and set to 0x63 to disable the minute setting for the RTC alarm.	0x6
11:8]	BCD_ALM_MIN	R/W		0x3
[7:4]	BCD_ALM_SEC_DEC	R/W	Alarm second setting by BCD in decimal values The decimal valid value is ranging from 0 to 5, and the unit valid value is ranging from 0 to 9. The legal programmed value is up to 0x59, and set to 0x63 to disable the second setting for the RTC alarm.	0x6
[3:0]	BCD_ALM_SEC	R/W		0x3

Table 4-197. RTC Alarm Time2 BCD Register (BCD_RTC_ALM2, Offset: 0x22C)

Bit	Symbol	Access	Description	Default
[31:16]	-	-	Reserved	-
[15:12]	BCD_ALM_MONTH_DEC	R/W	Alarm month setting by BCD in decimal and unit values The decimal valid value is ranging from 0 to 1, and the valid value is ranging from 0 to 9. The legal programmed value is from 0x1 to 0x12, and set to 0x0 to	0x0
[11:8]	BCD_ALM_MONTH	R/W		0x0

Vangotech

			disable the month setting for the RTC alarm.	
[7:4]	BCD_ALM_DATE_DEC	R/W	Alarm date setting by BCD in decimal & unit values The decimal valid value is ranging from 0 to 3, and the valid value is ranging from 0 to 9. The legal programmed value is from 0x1 to 0x31, and set to 0x0 to disable the date setting for the RTC alarm.	0x0
[3:0]	ALM_DATE	R/W		0x0

4.6.6. Extension Register Description

Table 4-198. Extension register offsets for V8130L.

Register Name	Offset	Description	Default Value
	0x0000_8000	CA7 configuration 1 register	0x9F000000
	0x0000_8004	CA7 configuration 2 register	0xE401E000
	0x0000_8008	CA7 configuration 3 register	0x10029000
	0x0000_800C	CA7 configuration 4 register	0x01FFFF00
	0x0000_8010	CA7 configuration 5 register	0x02010000
	0x0000_8014	CA7 configuration 6 register	0x00180010
	0x0000_801C	CA7's SPI Host magic number	0x20200110
	0x0000_8020	PLL lock status	0x00000013
	0x0000_8024	DDR CTRL register 0	0x72078082
	0x0000_8028	DDR CTRL register 1	0x03030803
	0x0000_802C	DDR CTRL register 2	0x030303E0
	0x0000_8030	SBS_OTG_u0 CTRL register	0x000F0182
	0x0000_8034	SBS_OTG_u1 CTRL register	0x000F0182
FTSCU100_OFFSET_PLL1CR	0x0000_8040	PLL1 control register	0x00350121
	0x0000_8044	PLL1 lock timer register	0x000005DC
FTSCU100_OFFSET_PLL2CR	0x0000_8048	PLL2 control register	0x00330121
	0x0000_804C	PLL2 lock timer register	0x000005DC
FTSCU100_OFFSET_PLL3CR	0x0000_8050	PLL3 control register	0x040D525C
	0x0000_8054	PLL3 lock timer register	0x00002328
FTSCU100_OFFSET_PLL5CR	0x0000_8060	PLL5 control register	0x020C625C
	0x0000_8064	PLL5 lock timer register	0x00002328
	0x0000_8068	DLL0 control register	0x00000000
	0x0000_806C	DLL1 control register	0x00000000
	0x0000_8070	Clock Enable 0 register	0x00000000
	0x0000_8074	Clock Enable 1 register	0x00000000

Vangotech

	0x0000_8078	Clock Enable 2 register	0x00000000
FTSCU100_OFFSET_CLKMUX	0x0000_807C	Clock Mux Selection register	0x00000020
FTSCU100_OFFSET_CLKDIV1	0x0000_8080	Clock Divider 1 register	0x12515C53
FTSCU100_OFFSET_CLKDIV2	0x0000_8084	Clock Divider 2 register	0xC5330C31
FTSCU100_OFFSET_CLKDIV3	0x0000_8088	Clock Divider 3 register	0x34D310C1
FTSCU100_OFFSET_CLKDIV4	0x0000_808C	Clock Divider 4 register	0x51154141
	0x0000_8090	Miscellaneous	0x00000000
	0x0000_8094	Software Reset Mask 0 register	0x00000000
	0x0000_8098	Software Reset Mask 1 register	0x00000000
	0x0000_809C	Software Reset Mask 2 register	0x00000000
	0x0000_80A0	Software Reset Mask 3 register	0x00000000
	0x0000_80A4	Software Reset register	0x00000000
FTSCU100_OFFSET_PLL6CR	0x0000_80C8	PLL6 control register	0x00330121
	0x0000_80CC	PLL6 lock timer register	0x000005DC
	0x0000_80D0	Clock Divider 5 register	0x099E79E7
	0x0000_80D4	Efuse read password	0x20200110
	0x0000_80D8	WDT tickle register	0x00000000
	0x0000_80DC	Efuse feature read	0x00000000
	0x0000_80E0	eMMC control register	0x00000000
	0x0000_80E4	SBS_OTG_u2 CTRL register	0x00F0C182
	0x0000_80E8	Clock Divider 6 register	0x03111144
	0x0000_80EC	Gpio0_bps_en	0x00000000
	0x0000_80F0	Gpio0_bps_in	0x00000000
	0x0000_80F4	Gpio1_bps_en	0x00000000
	0x0000_80F8	Gpio1_bps_in	0x00000000
	0x0000_8100	RGMII GTX delay chain selection	0x00000000
	0x0000_8104	RGMII RX delay chain selection	0x00000000
	0x0000_8108	TZC configuration	0x0000000C
	0x0000_810C	SPI address cycle	0x00000000
	0x0000_8110	SSP_SPI MUX selection	0x00000000
FTSCU100_OFFSET_PLL7CR	0x0000_8114	PLL7 control register	0x00240120
	0x0000_8118	PLL7 lock timer register	0x000005DC
	0x0000_8500	SPI_CS_N IO control register	0x00000040
	0x0000_8504	SPI_CS_CLK IO control register	0x00000040
	0x0000_8508	SPI_DO IO control register	0x00000040
	0x0000_850C	SPI_DI IO control register	0x00000040
	0x0000_8510	SPI_WP_N IO control register	0x00000040
	0x0000_8514	SPI_HOLD_N IO control register	0x00000040
	0x0000_8518	EMMC0_DATA_7 IO control register	0x000000C8
	0x0000_851C	EMMC0_DATA_6 IO control	0x000000C8

Vangotech

		register	
	0x0000_8520	EMMC0_DATA_5 IO control register	0x000000C8
	0x0000_8524	EMMC0_DATA_4 IO control register	0x000000C8
	0x0000_8528	EMMC0_DATA_3 IO control register	0x000000C8
	0x0000_852C	EMMC0_DATA_2 IO control register	0x000000C8
	0x0000_8530	EMMC0_DATA_1 IO control register	0x000000C8
	0x0000_8534	EMMC0_DATA_0 IO control register	0x000000C8
	0x0000_8538	EMMC0_RSTN IO control register	0x000000C8
	0x0000_853C	EMMC0_CMD IO control register	0x000000C0
	0x0000_8540	EMMC0_CLK IO control register	0x000000C0
	0x0000_8544	USB0_DRVBUS IO control register	0x00000040
	0x0000_8548	USB0_VBUS IO control register	0x00000000
	0x0000_854C	USB1_DRVBUS IO control register	0x00000040
	0x0000_8550	USB1_VBUS IO control register	0x00000000
	0x0000_8554	USB2_DRVBUS IO control register	0x00000040
	0x0000_8558	USB2_VBUS IO control register	0x00000000
	0x0000_855C	JTAG_TRST_N IO control register	0x00000050
	0x0000_8560	JTAG_TDI IO control register	0x00000040
	0x0000_8564	JTAG_SWDITMS IO control register	0x00000040
	0x0000_8568	JTAG_SWCLKTCK IO control register	0x00000040
	0x0000_856C	JTAG_TDO IO control register	0x00000040
	0x0000_8570	HSPI_CLK IO control register	0x00000040
	0x0000_8574	HSPI_CS IO control register	0x00000040
	0x0000_8578	HSPI_DI IO control register	0x00000040
	0x0000_857C	HSPI_DO IO control register	0x00000040
	0x0000_8580	I2C0_CLK IO control register	0x00000040
	0x0000_8584	I2C0_DATA IO control register	0x00000040
	0x0000_8588	I2C1_CLK IO control register	0x00000040

Vangotech

	0x0000_858C	I2C1_DATA IO control register	0x00000040
	0x0000_8590	I2C2_CLK IO control register	0x00000040
	0x0000_8594	I2C2_DATA IO control register	0x00000040
	0x0000_8598	I2C3_CLK IO control register	0x00000040
	0x0000_859C	I2C3_DATA IO control register	0x00000040
	0x0000_85A0	SPI0_CLK IO control register	0x00000040
	0x0000_85A4	SPI0_CS IO control register	0x00000040
	0x0000_85A8	SPI0_DI IO control register	0x00000040
	0x0000_85AC	SPI0_DO IO control register	0x00000040
	0x0000_85B0	SPI1_CLK IO control register	0x00000040
	0x0000_85B4	SPI1_CS IO control register	0x00000040
	0x0000_85B8	SPI1_DI IO control register	0x00000040
	0x0000_85BC	SPI1_DO IO control register	0x00000040
	0x0000_85C0	SPI2_CLK IO control register	0x00000040
	0x0000_85C4	SPI2_CS IO control register	0x00000040
	0x0000_85C8	SPI2_DI IO control register	0x00000040
	0x0000_85CC	SPI2_DO IO control register	0x00000040
	0x0000_85D0	SPI3_CLK IO control register	0x00000040
	0x0000_85D4	SPI3_CS IO control register	0x00000040
	0x0000_85D8	SPI3_DI IO control register	0x00000040
	0x0000_85DC	SPI3_DO IO control register	0x00000040
	0x0000_85E0	SPI4_CLK IO control register	0x00000040
	0x0000_85E4	SPI4_CS IO control register	0x00000040
	0x0000_85E8	SPI4_DI IO control register	0x00000040
	0x0000_85EC	SPI4_DO IO control register	0x00000040
	0x0000_85F0	SPI5_CLK IO control register	0x00000040
	0x0000_85F4	SPI5_CS IO control register	0x00000040
	0x0000_85F8	SPI5_DI IO control register	0x00000040
	0x0000_85FC	SPI5_DO IO control register	0x00000040
	0x0000_8600	SPI6_CLK IO control register	0x00000040
	0x0000_8604	SPI6_CS IO control register	0x00000040
	0x0000_8608	SPI6_DI IO control register	0x00000040
	0x0000_860C	SPI6_DO IO control register	0x00000040
	0x0000_8610	UART0_TX IO control register	0x00000040
	0x0000_8614	UART0_RX IO control register	0x00000040
	0x0000_8618	UART1_TX IO control register	0x00000040
	0x0000_861C	UART1_RX IO control register	0x00000040
	0x0000_8620	UART2_TX IO control register	0x00000040
	0x0000_8624	UART2_RX IO control register	0x00000040
	0x0000_8628	UART2_NRTS IO control register	0x00000040
	0x0000_862C	UART2_NCTS IO control register	0x00000040

Vangotech

	0x0000_8630	UART3_TX IO control register	0x00000040
	0x0000_8634	UART3_RX IO control register	0x00000040
	0x0000_8638	UART3_NRTS IO control register	0x00000040
	0x0000_863C	UART3_NCTS IO control register	0x00000040
	0x0000_8640	UART4_TX IO control register	0x00000040
	0x0000_8644	UART4_RX IO control register	0x00000040
	0x0000_8648	SPI7_CLK IO control register	0x00000040
	0x0000_864C	SPI7_CS IO control register	0x00000040
	0x0000_8650	SPI7_DI IO control register	0x00000040
	0x0000_8654	SPI7_DO IO control register	0x00000040
	0x0000_8658	EMMC1_DATA_7 IO control register	0x000000C8
	0x0000_865C	EMMC1_DATA_6 IO control register	0x000000C8
	0x0000_8660	EMMC1_DATA_5 IO control register	0x000000C8
	0x0000_8664	EMMC1_DATA_4 IO control register	0x000000C8
	0x0000_8668	EMMC1_DATA_3 IO control register	0x000000C8
	0x0000_866C	EMMC1_DATA_2 IO control register	0x000000C8
	0x0000_8670	EMMC1_DATA_1 IO control register	0x000000C8
	0x0000_8674	EMMC1_DATA_0 IO control register	0x000000C8
	0x0000_8678	EMMC1_RSTN IO control register	0x000000C0
	0x0000_867C	EMMC1_CMD IO control register	0x000000C8
	0x0000_8680	EMMC1_CLK IO control register	0x000000C0
	0x0000_8684	I2S0_FS IO control register	0x00000040
	0x0000_8688	I2S0_RXD IO control register	0x00000040
	0x0000_868C	I2S0_SCLK IO control register	0x00000040
	0x0000_8690	I2S0_TXD IO control register	0x00000040
	0x0000_8694	UART5_TX IO control register	0x00000040
	0x0000_8698	UART5_RX IO control register	0x00000040
	0x0000_869C	UART5_RX_h IO control register	0x00000040
	0x0000_86A0	UART6_TX IO control register	0x00000040
	0x0000_86A4	UART6_RX IO control register	0x00000040
	0x0000_86A8	UART7_TX IO control register	0x00000040
	0x0000_86AC	UART7_RX IO control register	0x00000040

Vangotech

	0x0000_86B0	UART8_TX IO control register	0x00000040
	0x0000_86B4	UART8_RX IO control register	0x00000040
	0x0000_86B8	UART8_RX_h IO control register	0x00000040
	0x0000_86BC	UART9_RX IO control register	0x00000040
	0x0000_86C0	UART9_TX IO control register	0x00000040
	0x0000_86C4	RGMII_RX_CK IO control register	0x00000088
	0x0000_86C8	RGMII_RXD0 IO control register	0x00000080
	0x0000_86CC	RGMII_RXD1 IO control register	0x00000080
	0x0000_86D0	RGMII_RXD2 IO control register	0x00000080
	0x0000_86D4	RGMII_RXD3 IO control register	0x00000080
	0x0000_86D8	RGMII_RXCTL IO control register	0x00000080
	0x0000_86DC	RGMII_GTX_CK IO control register	0x00000080
	0x0000_86E0	RGMII_TXCTL IO control register	0x00000080
	0x0000_86E4	RGMII_TXD0 IO control register	0x00000080
	0x0000_86E8	RGMII_TXD1 IO control register	0x00000080
	0x0000_86EC	RGMII_TXD2 IO control register	0x00000080
	0x0000_86F0	RGMII_TXD3 IO control register	0x00000080
	0x0000_86F4	RGMII_MDC IO control register	0x00000080
	0x0000_86F8	RGMII_MDIO IO control register	0x00000080
	0x0000_86FC	RMII_TXD0 IO control register	0x00000040
	0x0000_8700	RMII_TXD1 IO control register	0x00000040
	0x0000_8704	RMII_TX_EN IO control register	0x00000040
	0x0000_8708	RMII_RX_CRSDV IO control register	0x00000040
	0x0000_870C	RMII_RX_ER IO control register	0x00000040
	0x0000_8710	RMII_RXD0 IO control register	0x00000040
	0x0000_8714	RMII_RXD1 IO control register	0x00000040
	0x0000_8718	RMII_REF_CLK IO control register	0x00000040
	0x0000_871C	RMII_MDC IO control register	0x00000040
	0x0000_8720	RMII_MDIO IO control register	0x00000040
	0x0000_8724	MII_TXD0 IO control register	0x00000040
	0x0000_8728	MII_TXD1 IO control register	0x00000040
	0x0000_872C	MII_TXD2 IO control register	0x00000040
	0x0000_8730	MII_TXD3 IO control register	0x00000040
	0x0000_8734	MII_TX_EN IO control register	0x00000040
	0x0000_8738	MII_RX_DV IO control register	0x00000040

Vangotech

	0x0000_873C	MII_RX_ER IO control register	0x00000040
	0x0000_8740	MII_RXD0 IO control register	0x00000040
	0x0000_8744	MII_RXD1 IO control register	0x00000040
	0x0000_8748	MII_RXD2 IO control register	0x00000040
	0x0000_874C	MII_RXD3 IO control register	0x00000040
	0x0000_8750	MII_TX_ER IO control register	0x00000040
	0x0000_8754	MII_CRS IO control register	0x00000040
	0x0000_8758	MII_COL IO control register	0x00000040
	0x0000_875C	MII_TX_CK IO control register	0x00000040
	0x0000_8760	MII_MDC IO control register	0x00000040
	0x0000_8764	MII_MDIO IO control register	0x00000040
	0x0000_8768	MII_RXCK control register	0x00000040
	0x0000_876C	LC_PCLK IO control register	0x000000C0
	0x0000_8770	LC_VS IO control register	0x000000C0
	0x0000_8774	LC_HS IO control register	0x000000C0
	0x0000_8778	LC_DE IO control register	0x000000C0
	0x0000_877C	LC_DAT0 IO control register	0x000000C0
	0x0000_8780	LC_DAT1 IO control register	0x000000C0
	0x0000_8784	LC_DAT2 IO control register	0x000000C0
	0x0000_8788	LC_DAT3 IO control register	0x000000C0
	0x0000_878C	LC_DAT4 IO control register	0x000000C0
	0x0000_8790	LC_DAT5 IO control register	0x000000C0
	0x0000_8794	LC_DAT6 IO control register	0x000000C0
	0x0000_8798	LC_DAT7 IO control register	0x000000C0
	0x0000_879C	LC_DAT8 IO control register	0x000000C0
	0x0000_87A0	LC_DAT9 IO control register	0x000000C0
	0x0000_87A4	LC_DAT10 IO control register	0x000000C0
	0x0000_87A8	LC_DAT11 IO control register	0x000000C0
	0x0000_87AC	LC_DAT12 IO control register	0x000000C0
	0x0000_87B0	LC_DAT13 IO control register	0x000000C0
	0x0000_87B4	LC_DAT14 IO control register	0x000000C0
	0x0000_87B8	LC_DAT15 IO control register	0x000000C0
	0x0000_87BC	LC_DAT16 IO control register	0x000000C0
	0x0000_87C0	LC_DAT17 IO control register	0x000000C0
	0x0000_87C4	LC_DAT18 IO control register	0x000000C0
	0x0000_87C8	LC_DAT19 IO control register	0x000000C0
	0x0000_87CC	LC_DAT20 IO control register	0x000000C0
	0x0000_87D0	LC_DAT21 IO control register	0x000000C0
	0x0000_87D4	LC_DAT22 IO control register	0x000000C0
	0x0000_87D8	LC_DAT23 IO control register	0x000000C0
	0x0000_87DC	I2S_MCLK IO control register	0x00000040

Vangotech

	0x0000_87E0	SPI_DCX1 IO control register	0x00000040
	0x0000_87E4	SPI_DCX2 IO control register	0x00000040
	0x0000_87E8	DDRCK IO control register	0x00000040
	0x0000_87EC	PSW0 IO control register	0x00000040
	0x0000_87F4	SYS_XTAL IO control register	0x00000001

4.6.6.1. CA7 Configuration 1 Register

Table 4-199. CA7 Configuration 1 Register (Offset: 0x0000_8000)

Bit	Symbol	Access	Description	Default
29	CISBYPASS	RW	Cross trigger channel interface sync bypass	0x0
28:25	CIHSBYPASS[3:0]	RW	Cross trigger channel interface handshake bypass	0xF
24	ACINACTM	RW	In a single cluster system, tie ACINACTM High.	0x1
23:20	-	-	-	-
19:16	CLUSTERID	RW	Value read in the Cluster ID field	0x0
15:0	-	-	-	-

4.6.6.2. CA7 Configuration 2 Register

Table 4-200. CA7 Configuration 2 Register (Offset: 0x0000_8004)

Bit	Symbol	Access	Description	Default
31	TSVALUEB_EN_CA7	RW	Time stamp value enable	0x1
30	CNTVALUEB_EN_CA7	RW	Generic timer counter enable	0x1
29:27	MAXEXTIN[1:0]	RW	Number of external outputs that ASIC supports	0x4
26:25	MAXEXTOUT[2:0]	RW	Number of external inputs that ASIC supports	0x2
24:16	-	-	-	-
15:0	PERIPHBASE	RW	Specifies the GIC base address[30:15]	0xE000

4.6.6.3. CA7 Configuration 3 Register

Table 4-201. CA7 Configuration 3 Register (Offset: 0x0000_8008)

Bit	Symbol	Access	Description	Default
28	DBGROMADDRV	RW	Valid signal for DBGROMADDR	0x1
27:0	DBGROMADDR	RW	Specifies bits [39:12] of the ROM table physical address	0x29000

4.6.6.4. CA7 Configuration 4 Register

Table 4-202. CA7 Configuration 4 Register (Offset: 0x0000_800C)

Bit	Symbol	Access	Description	Default
31:20	-	-	-	-

Vangotech

19:16	SPIDEN	RW	Secure privileged invasive debug enable	0xF
15:12	SPNIDEN	RW	Secure privileged non-invasive debug enable	0xF
11:8	NIDEN	RW	Non-invasive debug enable	0xF
7:4	-	-	-	-
3:0	EDBGRQ	RW	The ORed value of all the debug request signal sources in the user system must be connected to this signal.	0x0

4.6.6.5. CA7 Configuration 5 Register

Table 4-203. CA7 Configuration 5 Register (Offset: 0x0000_8010)

Bit	Symbol	Access	Description	Default
31	-	-	Reserved	-
25	DBGSELFADDRV	RW	Valid signal for DBGSELFADDR	0x1
24:23	-	-	-	-
22:0	DBGSELFADDR	RW	Set this signal to bits[39:17] of the offset address that the debug APB interface of the macro cell is mapped to from the debug ROM physical address. If users want to change this setting, users must do this when the macro cell is at the reset state.	0x10000

4.6.6.6. CA7 Configuration 6 Register

Table 4-204. CA7 Configuration 6 Register (Offset: 0x0000_8014)

Bit	Symbol	Access	Description	Default
31:22	-	-	Reserved	-
21:19	edge_ratio_cnt_pclk	RW	edge ratio of CA7 PCLK bus(N:1)	0x3
18:15	SMPnAMP	RO	CA7 SMP status	0x0
14:11	STANDBYWFI	RO	CA7 STANDBYWFI status	0x0
10:7	STANDBYWFE	RO	CA7 STANDBYWFE status	0x0
6:4	edge_ratio_cnt_aclk	RW	edge ratio of CA7 ACLK bus(N:1)	0x1
3:0	-	-	-	-

4.6.6.7. CA7's SPI Host Magic Number Register

Table 4-205. CA7's SPI Host Magic Number Register (Offset: 0x0000_801C)

Bit	Symbol	Access	Description	Default
31:0	HSPI password	RW	CA7's SPI Host magic number	0x20200110

4.6.6.8. PLL Lock Status Register

Table 4-206. PLL Lock Status Register (Offset: 0x0000_8020)

Vangotech

Bit	Symbol	Access	Description	Default
31:6	-	-	-	-
5	PLL7 Lock status	RO	PLL7 Lock status	0x0
4	PLL6 Lock status	RO	PLL6 Lock status	0x0
3	PLL5 Lock status	RO	PLL5 Lock status	0x0
2	PLL3 Lock status	RO	PLL3 Lock status	0x0
1	PLL2 Lock status	RO	PLL2 Lock status	0x0
0	PLL1 Lock status	RO	PLL1 Lock status 0: Non-stable 1: Stable Note: PLL0/PLL4 is controlled by SCU.	0x0

4.6.6.9. DDR CTRL 0 Register

Table 4-207. DDR CTRL 0 Register (Offset: 0x0000_8024)

Bit	Symbol	Access	Description	Default
30:28	VREF_SELECT	RW	Bias voltage generated by PHY for VREF (When SELFBIAS = '1') '000': 0.55 * VCC15O_DDR '001': 0.525 * VCC15O_DDR '010': 0.475 * VCC15O_DDR '011': 0.45 * VCC15O_DDR '100' ~ '111': 0.5 * VCC15O_DDR (Default)	0X7
27:25	CKTREE_SKEW_4	RW	The control bits of DDR_PHY_4 used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x1
23	ext_hiPrior_ch3	RW	Raise channel 3 priority	0x0
22	ext_hiPrior_ch2	RW	Raise channel 2 priority	0x0
21	ext_hiPrior_ch1	RW	Raise channel 1 priority	0x0
20	ext_hiPrior_ch0	RW	Raise channel 0 priority 0: Disable 1: Raise channel 0 priority	0x0
15	CLOCK_EN_A	RW	Enable the DDR CMD PHY internal clock.	0x1
14:10	CKTREE_DELAY_4	RW	The programmable delay control bits used to balance the DDR-PHY_4 and the chip clock tree of DDR controller	0x0
8	SELFBIAS	RW	Enable self-bias generation 1: Bias from inside (VREF is floating.) 0: Bias from outside (VREF is connected to the outside bias power.)	0x0
7	CONUPDATE	RW	Continuous enable update code of DLL 1: Enable	0x1

Vangotech

			0: Disable	
5	SIO	RW	Set DQS in the single-ended mode or differential mode 1: Set DQS in the single-ended mode 0: Set DQS in the differential mode	0x0
4	IO15V	RW	I/O voltage selection When combo PHY is operating at 1.35 V, 1.5 V, or 1.8 V, please set this pin to logic 1. When combo PHY is operating at 1.2 V, please set this pin to logic '0'. 1: 1.35 V, 1.5 V, or 1.8 V 0: 1.2 V	0x0
3	BYONE	RW	Select the 1:1 mode or 1:2 mode 1: (1:1) mode 0: (1:2) mode	0x0
2	LPDDR2	RW	Enable the LPDDR2 mode	0x0
1	DDR3	RW	Enable DDR3 mode	0x1
0	MDDR1	RW	Enable MDDR (LPDDR) mode	0x0

4.6.6.10. DDR CTRL 1 Register

Table 4-208. DDR CTRL 1 Register (Offset: 0x0000_8028)

Bit	Symbol	Access	Description	Default
31:27	CKTREE_DELAY_1	RW	The programmable delay control bits used to balance the DDR-PHY_1 and the chip clock tree of DDR controller	0x0
26:24	CKTREE_SKEW_1	RW	The control bits of DDR-PHY_1 used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x3
23:19	CKTREE_DELAY_0	RW	The programmable delay control bits used to balance the DDR-PHY_0 and the chip clock tree of DDR controller	0x0
18:16	CKTREE_SKEW_0	RW	The control bits of DDR-PHY_0 used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x3
11:8	DUTYSEL	RW	The CK/CKB duty adjusting bits	0x8
7:3	CMD_CKTREE_DELAY	RW	The programmable delay control bits used to balance the DDR-PHY_CMD and the chip clock tree of DDR controller	0x0
2:0	CMD_CKTREE_SKEW	RW	The skew control of of DDR-PHY_CMD bits used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x3

Vangotech

4.6.6.11. DDR CTRL 2 Register

Table 4-209. DDR CTRL 2 Register (Offset: 0x0000_802C)

Bit	Symbol	Access	Description	Default
31:27	CKTREE_DELAY_3	RW	The programmable delay control bits used to balance the DDR-PHY_3 and the chip clock tree of DDR controller	0x0
26:24	CKTREE_SKEW_3	RW	The control bits of DDR-PHY_3 used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x3
23:19	CKTREE_DELAY_2	RW	The programmable delay control bits used to balance the DDR-PHY_2 and the chip clock tree of DDR controller	0x0
18:16	CKTREE_SKEW_2	RW	The control bits of DDR-PHY_2 used to adjust the skew between POSTPHYCLK2X and POSTCTLCLK	0x3
9:5	CLOCK_EN	RW	Enable the DDR PHY internal clock Bit[9]: DDR-PHY_4 Bit[8]: DDR-PHY_3 Bit[7]: DDR-PHY_2 Bit[6]: DDR-PHY_1 Bit[5]: DDR-PHY_0 '1': In the normal mode, all clock signals in DDR PHY are toggling. '0': In the low-power mode, the unused clock signals are gated.	0x1F
4:0	LEAKTEST	RW	Enter the I/O leakage test mode Bit[4]: DDR-PHY_4 Bit[3]: DDR-PHY_3 Bit[2]: DDR-PHY_2 Bit[1]: DDR-PHY_1 Bit[0]: DDR-PHY_0 '0': Normal operation '1': Power-down mode	0x0

4.6.6.12. SBS_OTG CTRL Register

Table 4-210. SBS_OTG CTRL Register (Offset: 0x0000_8030)

Bit	Symbol	Access	Description	Default
31:9	-	-	Reserved	-
23:21	PREEMPS	RW	3-bit control signal of the pre-emphasis strength 000: None 111: Strongest	0x7

Vangotech

			Default setting = 111	
20	PREENPS_EN	RW	Control pre-emphasis function 0: Disable 1: Enable Default setting = 1	0x1
19	V115_TEMP	RW	SQ reference voltage variation with temperature Default setting: V115_TEMP = 0	0x0
18	TXSR0	RW	High-speed transmitter slew rate adjustment. Default setting: TXSR0 = 0	0x0
17:16	RS[2:1]	RW	PLL loop filter resistor setting used for adjusting BW Default setting: RS[2:1] = [00]	0x0
15:14	PCR[2:1]	RW	High-speed transmitter current level setting Default setting: PCR[2:1] = [11]	0x1
13:12	IS[2:1]	RW	PLL charge pump current setting used for adjusting BW Default setting: IS[2:1] = [00]	0x0
11	HD2	RW	3-bit control signal of the reference voltage in the host disconnect detect circuit {HD2,HD1,HD0} 000: 574 mV 011: 629 mV 100: 503 mV 111: 557 mV	0x0
10	HD1	RW	-	0x0
9	HD0	RW	-	0x0
8	ext_suspendm	RW	This signal sets the transceiver in a mode that draws the minimum power from the supply and shuts down all blocks that are not necessary for the suspend/resume operations. When suspended, the TERMSELECT signal must always be set in the FS mode to ensure that the 1.5-kΩ pull-up resistor on DP remains powered. 0: Draw the suspend current 1: Draw the normal current	0x1
7	u_iddig	RW	This signal indicates that the connected plug is mini-A or mini-B. It must be valid within 20 ms after the u_idpulup signal is set to “1”. 0: Connected to a mini-A plug 1: Connected to a mini-B plug	0x0

Vangotech

6	wakeup	RW	The remote L2 wakeup invoked by application This signal is used only when FOTG210 operates in the device mode.	0x0
5	-	-	-	-
4	oscouten	RW	CLKOSCO clock output selection Toggling the OSCOUTEN signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signal to generate the glitches. 0: CLKOSCO is disabled. 1: CLKOSCO is enabled.	0x0
3	pllaliv	RW	When PLLALIV = '1', the suspended PLL will be alive, while SUSCLK120/SUSCLK480 and CLK48M will generate the clock. In other cases, SUSCLK120/SUSCLK480 will be generated as logic low. Toggling the PLLALIV signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signals to generate glitches.	0x0
2:1	xtlsel	RW	Reference clock frequency selection 00: Select the 12 MHz input reference clock 01: Select the 30 MHz input reference clock 1X: Reserved Reference clock is 12 MHz. Bits should be fixed at '00'.	0x1
0	outclksel	RW	Reference clock selection 0: Reference clock from XTALIN 1: Reference clock from CORECLKIN Reference clock is from XTALIN. This bit should be fixed at '0'.	0x0

4.6.6.13. SBS_OTG_1 CTRL Register

Table 4-211. SBS_OTG_1 CTRL Register (Offset: 0x0000_8034)

Bit	Symbol	Access	Description	Default
31:24	-	-	Reserved	-
23:21	PREEMPS	RW	3-bit control signal of the pre-emphasis strength 000: None 111: Strongest Default setting = 111	0x7

Vangotech

20	PREENPS_EN	RW	Control pre-emphasis function 0: Disable 1: Enable Default setting = 1	0x1
19	V115_TEMP	RW	SQ reference voltage variation with temperature Default setting: V115_TEMP = 0	0x0
18	TXSR0	RW	High-speed transmitter slew rate adjustment. Default setting: TXSR0 = 0	0x0
17:16	RS[2:1]	RW	PLL loop filter resistor setting used for adjusting BW Default setting: RS[2:1] = [00]	0x0
15:14	PCR[2:1]	RW	High-speed transmitter current level setting Default setting: PCR[2:1] = [11]	0x1
13:12	IS[2:1]	RW	PLL charge pump current setting used for adjusting BW Default setting: IS[2:1] = [00]	0x0
11	HD2	RW	3-bit control signal of the reference voltage in the host disconnect detect circuit {HD2,HD1,HD0} 000: 574 mV 011: 629 mV 100: 503 mV 111: 557 mV	0x0
10	HD1	RW	-	0x0
9	HD0	RW	-	0x0
8	ext_suspendm	RW	This signal sets the transceiver in a mode that draws the minimum power from the supply and shuts down all blocks that are not necessary for the suspend/resume operations. When suspended, the TERMSELECT signal must always be set in the FS mode to ensure that the 1.5-kΩ pull-up resistor on DP remains powered. 0: Draw the suspend current 1: Draw the normal current	0x1
7	u_iddig	RW	This signal indicates that the connected plug is mini-A or mini-B. It must be valid within 20 ms after the u_idpulup signal is set to "1". 0: Connected to a mini-A plug 1: Connected to a mini-B plug	0x0
6	wakeup	RW	The remote L2 wakeup invoked by application	0x0

Vangotech

			This signal is used only when FOTG210 operates in the device mode.	
5	-	-	-	-
4	oscouten	RW	CLKOSCO clock output selection Toggling the OSCOUTEN signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signal to generate the glitches. 0: CLKOSCO is disabled. 1: CLKOSCO is enabled.	0x0
3	pllaliv	RW	When PLLALIV = '1', the suspended PLL will be alive, while SUSCLK120/SUSCLK480 and CLK48M will generate the clock. In other cases, SUSCLK120/SUSCLK480 will be generated as logic low. Toggling the PLLALIV signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signals to generate glitches.	0x0
2:1	xtlsel	RW	Reference clock frequency selection 00: Select the 12 MHz input reference clock 01: Select the 30 MHz input reference clock 1X: Reserved Reference clock is 12 MHz. Bits should be fixed at '00'.	0x1
0	outclksel	RW	Reference clock selection 0: Reference clock from XTALIN 1: Reference clock from CORECLKIN Reference clock is from XTALIN. This bit should be fixed at '0'.	0x0

4.6.6.14. PLL1 Control Register

Table 4-212. PLL1 Control Register (Offset: 0x0000_8040)

Bit	Symbol	Access	Description	Default
31:22	-	-	Reserved	-
21:20	FS	RW	PLL1 CKOUT frequency range selection	0x3
19:13	NS	RW	PLL1 N value	0x28
12:8	MS	RW	PLL1 M value	0x1
5:4	CC	RW	PLL1 bandwidth selection	0x2
3:2	-	-	-	-
0	EN	RW	PLL1 enable signal (Default enable and disable by	0x1

Vangotech

			users) 1: Enable PLL 0: Power down PLL	

4.6.6.15. PLL1 Lock Timer Register

Table 4-213. PLL1 Lock Timer Register (Offset: 0x0000_8044)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL1 lock stable release timer	0x000005DC

4.6.6.16. PLL2 Control Register

Table 4-214. PLL2 Control Register (Offset: 0x0000_8048)

Bit	Symbol	Access	Description	Default
31:22	-	-	Reserved	-
21:20	FS	RW	PLL2 CKOUT frequency range selection	0x3
19:13	NS	RW	PLL2 N value	0x18
12:8	MS	RW	PLL2 M value	0x1
7:6	-	-	-	-
5:4	CC	RW	PLL2 loop bandwidth selection	0x2
3:2	-	-	-	-
0	EN	RW	PLL2 enable signal (Default enable and disable by users) 1: Enable PLL 0: Power down PLL	0x1

4.6.6.17. PLL2 Lock Timer Register

Table 4-215. PLL2 Lock Timer Register (Offset: 0x0000_804C)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL2 lock stable release timer	0x000005DC

4.6.6.18. PLL3 Control Register

Table 4-216. PLL3 Control Register (Offset: 0x0000_8050)

Bit	Symbol	Access	Description	Default
31:29	-	-	Reserved	-
28:24	PS	RW	PLL3 P value	0x4
20:12	NS	RW	PLL3 N value	0xD5
10:8	MS	RW	PLL3 M value	0x2
6:4	IS	RW	PLL3 loop bandwidth selection	0x5
3:2	RS	RW	PLL3 loop bandwidth selection	0x3
0	EN	RW	PLL3 enable signal 1: Enable PLL	0x0

Vangotech

			0: Power down PLL	

4.6.6.19. PLL3 Lock Timer Register

Table 4-217. PLL3 Lock Timer Register (Offset: 0x0000_8054)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL3 lock stable release timer	0x00002328

4.6.6.20. PLL5 Control Register

Table 4-218. PLL5 Control Register (Offset: 0x0000_8060)

Bit	Symbol	Access	Description	Default
31:29	-	-	Reserved	-
28:24	PS	RW	PLL5 P value	0x2
20:12	NS	RW	PLL5 N value	0xC6
10:8	MS	RW	PLL5 M value	0x2
6:4	IS	RW	PLL5 loop bandwidth selection	0x5
3:2	RS	RW	PLL5 loop bandwidth selection	0x3
0	EN	RW	PLL5 enable signal 1: Enable PLL 0: Power down PLL	0x0

4.6.6.21. PLL5 Lock Timer Register

Table 4-219. PLL5 Lock Timer Register (Offset: 0x0000_8064)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL5 lock stable release timer	0x00002328

4.6.6.22. DLL0 Control Register

Table 4-220. DLL0 Control Register (Offset: 0x0000_8068)

Bit	Symbol	Access	Description	Default
31:29	-	-	Reserved	-
8	lock_in	RO	Lock status	0x0
7	-	-	-	-
6	pdn	RW	Power-down mode setting, active low.	0x0
5:4	frange	RW	Frequency range setting signal 00: 50 MHz ~ 73 MHz 01: 73 MHz ~ 110 MHz 10: 110 MHz ~ 150 MHz 11: 150 MHz ~ 210 MHz	0x11
3:2	pd_gain	RW	Control the gain of PD PD_GAIN[1:0] = 2'b00, the gain of PD will be the maximum value.	0x10

Vangotech

			PD_GAIN[1:0] = 2'b11, the gain of PD will be the minimum value.	
1:0	afl_gain	RW	Control the gain of AFL AFL_GAIN[1:0] = 2'b00: the gain of AFL will be the maximum value. AFL_GAIN[1:0] = 2'b11: the gain of AFL will be the minimum value.	0x10

4.6.6.23. DLL1 Control Register

Table 4-221. DLL1 Control Register (Offset: 0x0000_806C)

Bit	Symbol	Access	Description	Default
31:29	-	-	Reserved	-
8	lock_in	RO	Lock status	0x0
7	-	-	-	-
6	pdn	RW	Power-down mode setting, active low	0x0
5:4	frange	RW	Frequency range setting signal 00: 50 MHz ~ 73 MHz 01: 73 MHz ~ 110 MHz 10: 110 MHz ~ 150 MHz 11: 150 MHz ~ 210 MHz	0x11
3:2	pd_gain	RW	Control the gain of PD PD_GAIN[1:0] = 2'b00, the gain of PD will be the maximum value. PD_GAIN[1:0] = 2'b11, the gain of PD will be the minimum value.	0x10
1:0	afl_gain	RW	Control the gain of AFL AFL_GAIN[1:0] = 2'b00: the gain of AFL will be the maximum value. AFL_GAIN[1:0] = 2'b11: the gain of AFL will be the minimum value.	0x10

4.6.6.24. Clock Enable 0 Register

Table 4-222. Clock Enable 0 Register (Offset: 0x0000_8070)

Bit	Symbol	Access	Description	Default
31:16	-	-	Reserved	-
15	mac2_ck_src_en	RW	mac2 clock source enable control (MII)	0x0
14	mac1_ck_src_en	RW	mac1 clock source enable control (RMII)	0x0
13	mac0_ck_src_en	RW	mac0 clock source enable control (RGMII)	0x0
12	mac1_refck_out_en	RW	mac1 ref clock output to PHY enable control (RMII)	0x0
11	-	-	-	-

Vangotech

10	otg2_30m_in_en	RW	Otg2 ref clk clock enable control	0x0
9	mac1_refck_mac_en	RW	mac1 ref clock enable control(RMII)	0x0
8	mac_rxck_en	RW	mac rxck clock enable control(RGMII)	0x0
-	-	-	-	-
6	emmc1_ck_en	RW	sdc_1 clock enable control	0x0
5	emmc_ck_en	RW	sdc_0 clock enable control	0x0
4	lc_scaler_en	RW	scaler clock enable control	0x0
3	lc_clk_en	RW	lc_clk clock enable control	0x0
2	ddr3_ctlclk_en	RW	ddr main clock enable control	0x1
1	otg1_30m_in_en	RW	otg1 ref clk clock enable control	0x0
0	otg_30m_in_en	RW	otg ref clk clock enable control	0x0

4.6.6.25. Clock Enable 1 Register

Table 4-223. Clock Enable 1 Register (Offset: 0x0000_8074)

Bit	Symbol	Access	Description	Default
31	wdt3_extck_en	RW	wdt3 extclk clock enable control	0x1
30	wdt2_extck_en	RW	wdt2 extclk clock enable control	0x1
29	wdt1_extck_en	RW	wdt1 extclk clock enable control	0x1
28	wdt_extck_en	RW	wdt extclk clock enable control	0x1
27	uart9_uclk_en	RW	uart9_uclk clock enable control	0x0
26	uart8_uclk_en	RW	uart8_uclk clock enable control	0x0
25	uart7_uclk_en	RW	uart7_uclk clock enable control	0x0
24	uart6_uclk_en	RW	uart6_uclk clock enable control	0x0
23	uart5_uclk_en	RW	uart5_uclk clock enable control	0x0
22	uart4_uclk_en	RW	uart4_uclk clock enable control	0x0
21	uart3_uclk_en	RW	uart3_uclk clock enable control	0x0
20	uart2_uclk_en	RW	uart2_uclk clock enable control	0x0
19	uart1_uclk_en	RW	uart1_uclk clock enable control	0x0
18	uart_uclk_en	RW	uart_uclk clock enable control	0x1
17	tmr3_extck3_en	RW	Extclk3 clock enable control of PWMTIMER_3	0x0
16	tmr3_extck2_en	RW	Extclk2 clock enable control of PWMTIMER_3	0x0
15	tmr3_extck1_en	RW	Extclk1 clock enable control of PWMTIMER_3	0x0
14	tmr3_extck0_en	RW	Extclk0 clock enable control of PWMTIMER_3	0x0
13	tmr2_extck3_en	RW	Extclk3 clock enable control of PWMTIMER_2	0x0
12	tmr2_extck2_en	RW	Extclk2 clock enable control of PWMTIMER_2	0x0
11	tmr2_extck1_en	RW	Extclk1 clock enable control of PWMTIMER_2	0x0
10	tmr2_extck0_en	RW	Extclk0 clock enable control of PWMTIMER_2	0x0
9	tmr1_extck3_en	RW	Extclk3 clock enable control of PWMTIMER_1	0x0
8	tmr1_extck2_en	RW	Extclk2 clock enable control of PWMTIMER_1	0x0
7	tmr1_extck1_en	RW	Extclk1 clock enable control of PWMTIMER_1	0x0

Vangotech

6	tmr1_extck0_en	RW	Extclk0 clock enable control of PWMTIMER_1	0x0
5	tmr_extck3_en	RW	Extclk3 clock enable control of PWMTIMER	0x0
4	tmr_extck2_en	RW	Extclk2 clock enable control of PWMTIMER	0x0
3	tmr_extck1_en	RW	Extclk1 clock enable control of PWMTIMER	0x0
2	tmr_extck0_en	RW	Extclk0 clock enable control of PWMTIMER	0x0
1	canbus1_en	RW	canbus controller 1 clock enable control	0x0
0	canbus_en	RW	canbus controller clock enable control	0x0

4.6.6.26. Clock Enable 2 Register

Table 4-224. Clock Enable 2 Register (Offset: 0x0000_8078)

Bit	Symbol	Access	Description	Default
30	ssp7_ck_en	RW	Sspclk7 clock enable control	0x0
29	ssp6_ck_en	RW	Sspclk6 clock enable control	0x0
28	ssp5_ck_en	RW	Sspclk5 clock enable control	0x0
27	ssp4_ck_en	RW	Sspclk4 clock enable control	0x0
26	ssp3_ck_en	RW	Sspclk3 clock enable control	0x0
25	ssp2_ck_en	RW	Sspclk2 clock enable control	0x0
24	ssp1_ck_en	RW	Sspclk1 clock enable control	0x0
23	ssp_ck_en	RW	Sspclk clock enable control	0x0
22	ssp1_i2s_ck_en	RW	Sspclk1_i2s clock enable control	0x0
21	ssp_i2s_ck_en	RW	Sspclk_i2s clock enable control	0x0
20:10	-	-	Reserved	-
9	audio_clk_en	RW	Audio master clock enable control (12.288 MHz)	0x0
8	tdc_mclk_en	RW	TDC clock enable control	0x0
7	adc_mclk_en	RW	ADC clock enable control	0x0
6:0	-	-	Reserved	-

4.6.6.27. Clock Mux Selection Register

Table 4-225. Clock Mux Selection Register (Offset: 0x0000_807C)

Bit	Symbol	Access	Description	Default
31	peri3_osc_sw	RW	Peripheral OSC switch 3	0x0
30	peri2_osc_sw	RW	Peripheral OSC switch 2	0x0
29	peri1_osc_sw	RW	Peripheral OSC switch 1	0x0
28	uart8_fir_sel	RW	UART8 FIR mode selection	0x0
27	uart7_fir_sel	RW	UART7 FIR mode selection	0x0
26	uart6_fir_sel	RW	UART6 FIR mode selection	0x0
25	uart5_fir_sel	RW	UART5 FIR mode selection	0x0
24	uart4_fir_sel	RW	UART4 FIR mode selection	0x0
23	uart3_fir_sel	RW	UART3 FIR mode selection	0x0
22	uart2_fir_sel	RW	UART2 FIR mode selection	0x0

Vangotech

21	uart1_fir_sel	RW	UART1 FIR mode selection	0x0
20	uart_fir_sel	RW	UART FIR mode selection	0x0
19:17	-	-	-	-
16	peri4_osc_sw	RW	Peripheral OSC switch 4	0x0
14	rmii_refck_inv	RW	Select rmii refck output clock inverse 0: REFCLK 1: Inverse of REFCLK	0x0
13	emmc_osc_sw	RW	Select PLL3 as EMMC clock source 0: Internal 120 MHz OSC 1: PLL3	0x0
12	pll3_ref_sel	RW	pll3 ref clock sel 1: 25 MHz for SDC clock gen 0: 12 MHz for audio clock gen	0x0
11	ssp1_i2s_sel	RW	Select SSP1_I2S clock source for audio 12.288 MHz	0x0
10	ssp_i2s_sel	RW	Select SSP_I2S clock source for audio 12.288 MHz	0x0
9	pll6_osc_sw_1	RW	select PLL6 as clock source switch 1	0x0
8	pll6_osc_sw_2	RW	select PLL6 as clock source switch 2	0x0
7	timer_clk_sw	RW	Timer clock source sw 0: From PLL2 1: From 32.768 KHz	0x0
6	uart_30m_disable	RW	uart clock source selection 0: 30 MHz from 120 MHz (Internal Ring OSC) 1: From uart_fdiv clock output	0x0
5:4	-	-	-	-
3	uart9_fir_sel	RW	UART9 FIR mode selection	0x0
2:0	-	-	-	-

4.6.6.28. Clock Divider 1 Register

Table 4-226. Clock Divider 1 Register (Offset: 0x0000_8080)

Bit	Symbol	Access	Description	Default
27:22	mac_50M_cnt	RW	mac 50 MHz Ref clock divider	0x9
21:20	ssp_i2s_cnt	RW	ssp_i2s clock divider	0x1
19:14	apb_cnt	RW	Non-secure APB pclk divider	0x5
13:12	-	-	-	-
11:6	adc_pll_cnt	RW	ADC mclk divider	0x31
5:0	otg_src_cnt	RW	divider for OTG 30 MHz source clock	0x13

4.6.6.29. Clock Divider 2 Register

Table 4-227. Clock Divider 2 Register (Offset: 0x0000_8084)

vangotech

Bit	Symbol	Access	Description	Default
31:26	mac0_2p5_cnt	RW	2.5 MHz of mac0 divider (RGMII)	0x31
25:20	mac0_25_cnt	RW	25 MHz of mac0 divider (RGMII)	0x13
19:16	mac0_125_cnt	RW	125 MHz of mac0 divider (RGMII)	0x3
13:8	audio_cnt	RW	Audio clock source divider	0xC
7:4	-	RW	Reserved	0x3
3:0	lcd_cnt	RW	LCDC lc_clk divider	0x1

4.6.6.30. Clock Divider 3 Register

Table 4-228. Clock Divider 3 Register (Offset: 0x0000_8088)

Bit	Symbol	Access	Description	Default
31:28	ca7_dbg_cnt	RW	CA7 debug clk divider	0x3
27:22	mac2_25_cnt	RW	25 MHz of mac2 divider (MII)	0X13
21:18	uart_fir_cnt	RW	uart fir mode clock divider	0x4
17:12	lcdc_pll_cnt	RW	PLL source of DDR clock divider	0x31
11:6	uart_cnt	RW	uart clock divider	0x3
5:0	lc_scaler_cnt	RW	LCDC scaler divider	0x1

4.6.6.31. Clock Divider 4 Register

Table 4-229. Clock Divider 4 Register (Offset: 0x0000_808C)

Bit	Symbol	Access	Description	Default
31:30	mac_cnt	RW	mac pre divider	0x1
29:26	fcnt_n	RW	uart fraction divider parameter n	0x4
25:22	fcnt_m	RW	uart fraction divider parameter m	0x4
21:20	can1_cnt	RW	can bus 1 system clock divider	0x1
19:18	can0_cnt	RW	can bus 0 system clock divider	0x1
17:16	spi_cnt	RW	spi clk divider	0x1
15:14	ssp_spi_cnt	RW	ssp clk divider	0x1
11:8	-	-	-	-
7:4	ahb_cnt	RW	ahb clk divider	0x4
3:0	ca7_cnt	RW	Ca7 bus divider	0x1

4.6.6.32. MISC. Register

Table 4-230. MISC. Register (Offset: 0x0000_8090)

Bit	Symbol	Access	Description	Default
31:30	dma_switch	RW	DMA switch mode selection	0x0
29	boot_msg_dis	RO	FW boot message disable	0x0
28	EVENTI_EN	RW	EVENTI_EN 1: Enable WFE trigger in 0: Disable WFE trigger in	0x0

Vangotech

27:26	-	-	-	-
25:16	R0SIZE	RW	TZC secure region configuration	0x0
11	wdtoff_dis_3	RW	wdtoff_dis for WDT3	0x0
10	wdtoff_dis_2	RW	wdtoff_dis for WDT2	0x0
9	wdtoff_dis_1	RW	wdtoff_dis for WDT1	0x0
8	wdtoff_dis_0	RW	wdtoff_dis for WDT0	0x0
7:0	-	-	-	-

4.6.6.33. Software Reset Mask 0 (swrst_msk0) Register

Table 4-231. Software Reset Mask 0 (swrst_msk0) Register (Offset: 0x0000_8094)

Bit	Symbol	Access	Description	Default
31:0	swrst0	RW	[31]: presetn_apbs_u_CA7QCS Note: PCLKDBG & PCLKSYS are reset mask by the same signal. [30]: - [29]: rst_n_u_ADC [28]: Presetn_EMC_ROM [27]: Presetn_EMC_SRAM [26]: Presetn_TZC_lcdc_ddr [25]: Presetn_TZC_bus_ddr [24]: Presetn_TZC_ca7_ddr [23]: - [21]: PRESETn_u_H2P_u2 [17]: Aresetn_TZC_lcdc_ddr_ch3 [16]: Aresetn_TZC_bus_ddr_ch2 [15]: Aresetn_TZC_bus_ddr_ch1 [14]: Aresetn_TZC_ca7_ddr_ch0 [13]: AResetn3_DDR3_TOP [12]: AResetn2_DDR3_TOP [11]: AResetn1_DDR3_TOP [10]: AResetn0_DDR3_TOP [9]: MResetn_DDR3_TOP [8]: PResetn_DDR3_TOP [7]: AResetn_u_LCDC [6]: PRESETn_u_LCDC [5]: TV_RSTn_LCDC [4]: LC_SCALER_RSTn_LCDC [3]: LC_RSTn_LCDC [2]: ARSTn_DMAC_u1 [1]: ARSTn_DMAC_u0 [0]: PRSTn_DMAC_u1	0x0

Vangotech

4.6.6.34. Software Reset Mask 1 (swrst_msk1) Register

Table 4-232. Software Reset Mask 1 (swrst_msk1) Register (Offset: 0x0000_8098)

Bit	Symbol	Access	Description	Default
31:0	Swrst1	RW	[31]: PRSTn_DMAC_u0 [30]: AResetn_EMC_ROM [29]: AResetn_EMC_RAM [28]: AResetn_BP141 [27]: sys_rst_n_u_CAN_1 [26]: sys_rst_n_u_CAN [25]: PRSTn_u_SSP_I2S_u1 [24]: PRSTn_u_SSP_I2S_u0 [23]: PRSTn_u_SSP_SPI_u7 [22]: AResetn_u_MAC_RGMII [21]: PResetn_u_MAC_RGMII [20]: AResetn_u_MAC_RMII [19]: PResetn_u_MAC_RMII [18]: AResetn_u_MAC_MII [17]: PResetn_u_MAC_MII [16]: eFuse_RST_N_EFUSE40LP_CTRL [15]: PRESET_N_EFUSE40LP_CTRL [14]: sys_rstn_SPI2AHBL [13]: PRSTn_u_SSP_u6 [12]: PRSTn_u_SSP_u5 [11]: PRSTn_u_SSP_u4 [10]: PRSTn_u_SSP_u3 [9]: PRSTn_u_SSP_u2 [8]: PRSTn_u_SSP_u1 [7]: PRSTn_u_SSP_u0 [6]: - [5]: BusnRst_u_cryptomaster [4]: hw_rstn_u_SDC_eMMC [3]: hw_rstn_u_SDC_eMMC_1 [2]: sys_rstn_FTSPI020 [1]: prstn_WDT_u3 [0]: prstn_WDT_u2	0x0

4.6.6.35. Software Reset Mask 2 (swrst_msk2) Register

Table 4-233. Software Reset Mask 2 (swrst_msk2) Register (Offset: 0x0000_809C)

Bit	Symbol	Access	Description	Default

Vangotech

31:0	Swrst2	RW	[31]: prstn_WDT_u1 [30]: prstn_WDT_u0 [29]: prstn_TRNG [28]: prstn_GPIO_u0 [27]: prstn_GPIO_u1 [26]: prstn_KBC [25]: prstn_CAN_u0 [24]: prstn_CAN_u1 [23]: PRSTn_u_UART_u8 [22]: PRSTn_u_UART_u7 [21]: PRSTn_u_UART_u6 [20]: PRSTn_u_UART_u5 [19]: PRSTn_u_UART_u4 [18]: PRSTn_u_UART_u3 [17]: PRSTn_u_UART_u2 [16]: PRSTn_u_UART_u1 [15]: PRSTn_u_UART_u0 [14]: prstn_PWMTMR_u3 [13]: prstn_PWMTMR_u2 [12]: prstn_PWMTMR_u1 [11]: prstn_PWMTMR_u0 [10]: prstn_IIC_u4 [9]: prstn_IIC_u3 [8]: prstn_IIC_u2 [7]: prstn_IIC_u1 [6]: prstn_IIC_u0 [5]: prstn_TDC [4]: prstn_ADC [3]: - [2]: - [1]: prstn_H2P_u0 [0]: prstn_H2P_u1	0x0

4.6.6.36. Software Reset Mask 3 (swrst_msk3) Register

Table 4-234. Software Reset Mask 3 (swrst_msk3) Register (Offset: 0x0000_80A0)

Bit	Symbol	Access	Description	Default
18	-	RW	-	0x0
17	nic_0_bus_rstn	RW	NIC_0 bus reset	0x0
16	nic_1_bus_rstn	RW	NIC_1 bus reset	0x0
15:3	-	-	-	-
2	ca7_sysrst	RW	CA7 core reset	0x0

Vangotech

1	PRSTn_u_UART_u9	RW	Presetn of UART9	0x0
0	-	-	-	-

4.6.6.37. Software Reset Register

Table 4-235. Software Reset Register (Offset: 0x0000_80A4)

Bit	Symbol	Access	Description	Default
31:6	-	-	Reserved	-
5	otg2_rst_n	RW	OTG2 reset gated	0x0
4	otg1_rst_n	RW	OTG1 reset gated	0x0
3	otg_rst_n	RW	OTG reset gated	0x0
2	-	RW	-	0x0
1	Dphy_resetn	RW	Reset Dphy after FTDDR3030 init OK	0x0
0	-	-	-	-

4.6.6.38. PLL6 Control Register

Table 4-236. PLL6 Control Register (Offset: 0x0000_80C8)

Bit	Symbol	Access	Description	Default
31:22	-	-	Reserved	-
21:20	FS	RW	PLL6 CKOUT frequency range selection	0x3
19:13	NS	RW	PLL6 N value	0x18
12:8	MS	RW	PLL6 M value	0x1
5:4	CC	RW	PLL6 bandwidth selection	0x2
0	EN	RW	PLL6 enable signal (Default enable and disable by user) 1: Enable PLL 0: Power down PLL	0x1

4.6.6.39. PLL6 Lock Timer Register

Table 4-237. PLL6 Lock Timer Register (Offset: 0x0000_80CC)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL6 lock stable release timer	0x000005DC

4.6.6.40. Clock Divider 5 Register

Table 4-238. Clock Divider 5 Register (Offset: 0x0000_80D0)

Bit	Symbol	Access	Description	Default
31:30	-	-	-	-
29:24	mac2_2p5_cnt	RW	2.5 MHz of mac2 divider(MII)	0x9
23:18	Timer3_cnt	RW	pwm timer clock divider of PWMTMR_u3	0x27
17:12	Timer2_cnt	RW	pwm timer clock divider of PWMTMR_u2	0x27
11:6	Timer1_cnt	RW	pwm timer clock divider of PWMTMR_u1	0x27
5:0	Timer0_cnt	RW	pwm timer clock divider of PWMTMR_u0	0x27

Vangotech

4.6.6.41. Efuse Read Password Register

Table 4-239. Efuse Read Password Register (Offset: 0x0000_80D4)

Bit	Symbol	Access	Description	Default
31:0	efuse_pwd	RW	efuse read password	0x20200110

4.6.6.42. WDT Tickle Feature Register

Table 4-240. WDT Tickle Feature Register (Offset: 0x0000_80D8)

Bit	Symbol	Access	Description	Default
31:24	WDT0tickle	RW	WDT0 tickle register setting	0x0
23:16	WDT1tickle	RW	WDT1 tickle register setting	0x0
15:8	WDT2tickle	RW	WDT2 tickle register setting	0x0
7:0	WDT3tickle	RW	WDT3 tickle register setting	0x0

4.6.6.43. EFUSE Feature Read Register

Table 4-241. EFUSE Feature Read Register (Offset: 0x0000_80DC)

Bit	Symbol	Access	Description	Default
31:7	-	RO	-	0x0
6	emmc_drive_sel	RO	eMMC pad driving strength selection	0x0
5	ef_jtag_dis	RO	JTAG function disable 0: JTAG function is enabled. 1: JTAG function is disabled.	0x0
4	ef_sec_en	RO	Security enable 0: Security is disabled. 1: Security is enabled.	0x0
3	ef_xip_dis	RO	Boot from SPI Flash (XIP mode) disable 0: Allow the bootstrap function "BOOT_IPL_SEL" 1: Disable the bootstrap function "BOOT_IPL_SEL"	0x0
2:1	-	-	-	-
0	ef_hif_dis	RO	Host i/f disable 0: Enable the bootstrap function "HSPI_IF_ON" 1: Disable the bootstrap function "HSPI_IF_ON"	0x0

4.6.6.44. EFUSE Feature Read Register

Table 4-242. EFUSE Feature Read Register (Offset: 0x0000_80E0)

Bit	Symbol	Access	Description	Default

Vangotech

31:6	-	RW	-	0x0
5	io_sd_wpofeMMC_1	RW	io_sd_wp port of eMMC_1	0x0
4	io_sd_wpofeMMC	RW	io_sd_wp port of eMMC	0x0
3	io_sd_cdofeMMC_1	RW	io_sd_cd port of eMMC_1	0x0
2	io_sd_cdofeMMC	RW	io_sd_cd port of eMMC	0x0
1	io_sd_3v3_en_emmc_1	RW	sd 3v3 enable control of emmc_1 0: IO operate on 1.8v 1: IO operate on 3.3v	0x0
0	io_sd_3v3_en_emmc	RW	sd 3v3 enable control of emmc 0: IO operate on 1.8v 1: IO operate on 3.3v	0x0

4.6.6.45. SBS_OTG_2 CTRL Register

Table 4-243. SBS_OTG_2 CTRL Register (Offset: 0x0000_80E4)

Bit	Symbol	Access	Description	Default
31:9	-	-	Reserved	-
23:21	PREEMPS	RW	3-bit control signal of the pre-emphasis strength 000: None 111: Strongest Default setting = 111	0x7
20	PREENPS_EN	RW	Control pre-emphasis function 0: Disable 1: Enable Default setting = 1	0x1
19	V115_TEMP	RW	SQ reference voltage variation with temperature Default setting: V115_TEMP = 0	0x0
18	TXSR0	RW	High-speed transmitter slew rate adjustment. Default setting: TXSR0 = 0	0x0
17:16	RS[2:1]	RW	PLL loop filter resistor setting used for adjusting BW Default setting: RS[2:1] = [00]	0x0
15:14	PCR[2:1]	RW	High-speed transmitter current level setting Default setting: PCR[2:1] = [11]	0x3
13:12	IS[2:1]	RW	PLL charge pump current setting used for adjusting BW Default setting: IS[2:1] = [00]	0x0
11	HD2	RW	3-bit control signal of the reference voltage in the host disconnect detect circuit {HD2,HD1,HD0} 000: 574 mV	0x0

Vangotech

			011: 629 mV 100: 503 mV 111: 557 mV	
10	HD1	RW	-	0x0
9	HD0	RW	-	0x0
8	ext_suspendm	RW	This signal sets the transceiver in a mode that draws the minimum power from the supply and shuts down all blocks that are not necessary for the suspend/resume operations. When suspended, the TERMSELECT signal must always be set in the FS mode to ensure that the 1.5-kΩ pull-up resistor on DP remains powered. 0: Draw the suspend current 1: Draw the normal current	0x1
7	u_iddig	RW	This signal indicates that the connected plug is mini-A or mini-B. It must be valid within 20 ms after the u_idpulup signal is set to "1". 0: Connected to a mini-A plug 1: Connected to a mini-B plug	0x1
6	wakeup	RW	The remote L2 wakeup invoked by application This signal is used only when FOTG210 operates in the device mode.	0x0
5	-	-	-	-
4	oscouten	RW	CLKOSCO clock output selection Toggling the OSCOUTEN signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signal to generate the glitches. 0: CLKOSCO is disabled. 1: CLKOSCO is enabled.	0x0
3	pllaliv	RW	When PLLALIV = '1', the suspended PLL will be alive, while SUSCLK120/SUSCLK480 and CLK48M will generate the clock. In other cases, SUSCLK120/SUSCLK480 will be generated as logic low. Toggling the PLLALIV signal will cause the CLKOSCO, CLK60SYS, CLK48M, or SUSCLK120/SUSCLK480 signals to generate glitches.	0x0
2:1	xtlsel	RW	Reference clock frequency selection 00: Select the 12 MHz input reference clock	0x1

Vangotech

			01: Select the 30 MHz input reference clock 1X: Reserved Reference clock is 12 MHz. Bits should be fixed at '00'.	
0	outclksel	RW	Reference clock selection 0: Reference clock from XTALIN 1: Reference clock from CORECLKIN Reference clock is from XTALIN. This bit should be fixed at '0'.	0x0

4.6.6.46. Clock Divider 6 Register

Table 4-244. Clock Divider 6 Register (Offset: 0x0000_80E8)

Bit	Symbol	Access	Description	Default
31:26	-	-	-	-
25:20	tdc_pll_cnt	RW	TDC clock divider	0x31
19:16	axi_ddr_cnt	RW	DDR AXI clock divider	0x1
15:12	-	-	-	-
11:8	axi_dev_cnt	RW	AXI peripheral device divider	0x1
7:4	mac_ptpcnt	RW	mac ptpcnt divider	0x4
3:0	axi_2_cnt	RW	AXI2 clock divider of nic_1	0x4

4.6.6.47. GPIO0 bps Enable Register

Table 4-245. GPIO0 bps Enable Register (Offset: 0x0000_80EC)

Bit	Symbol	Access	Description	Default
31:0	gpio0_bps_en	RW	The PinBypass register controls the bypass mode.	0x0

4.6.6.48. GPIO0 bps Data Register

Table 4-246. GPIO0 bps Data Register (Offset: 0x0000_80F0)

Bit	Symbol	Access	Description	Default
31:0	gpio0_bps_in	RW	When PinBypass indicates the pin is in the bypass mode, gpio_en will be connected to gpio_bps_en, gpio_in will be connected to gpio_bps_out, and gpio_bps_in will be connected to gpio_out	0x0

4.6.6.49. GPIO1 bps Enable Register

Table 4-247. GPIO1 bps Enable Register (Offset: 0x0000_80F4)

Bit	Symbol	Access	Description	Default
31:0	gpio1_bps_en	RW	The PinBypass register controls the bypass mode	0x0

Vangotech

4.6.6.50. GPIO1 bps Data Register

Table 4-248. GPIO1 bps Data Register (Offset: 0x0000_80F8)

Bit	Symbol	Access	Description	Default
31:0	gpio1_bps_in	RW	When PinBypass indicates the pin is in the bypass mode, gpio_en will be connected to gpio_bps_en, gpio_in will be connected to gpio_bps_out, and gpio_bps_in will be connected to gpio_out	0x0

4.6.6.51. RGMII_gtx_clk_delay Register

Table 4-249. RGMII_gtx_clk_delay Register (Offset: 0x0000_8100)

Bit	Symbol	Access	Description	Default
15:0	gtx_ck_dly	RW	GTX_CK delay chain setting by RGMII [15]: inverse clock in [14]: bypass mode [13]: delay chain shift increment + 0.522 ns [12:0]: delay chain shift increment + 0.174 ns EX: Bit[15:0] = 0x1010_0100_0110_0100 Delay increment = (Bit[15]*0.5T(gtx_ck) + (0.522*Bit[13] + 0.174*($\Sigma Bit[n]$)) ns	0x0000000

4.6.6.52. RGMII_rx_clk_delay Register

Table 4-250. RGMII_rx_clk_delay Register (Offset: 0x0000_8104)

Bit	Symbol	Access	Description	Default
15:0	rx_ck_dly	RW	RX_CK delay chain setting by RGMII [15]: inverse clock in [14]: bypass mode [13]: delay chain shift increment + 0.522 ns [12:0]: delay chain shift increment + 0.174 ns EX: Bit[15:0] = 0x1010_0100_0110_0100 Delay increment = (Bit[15]*0.5T(rx_ck)+ (0.522*Bit[13] + 0.174*($\Sigma Bit[n]$)) ns	0x0000000

4.6.6.53. TZC Configure Register

Table 4-251. TZC Configure Register (Offset: 0x0000_8108)

Bit	Symbol	Access	Description	Default
3	FPID0_CA7	RW	The Fast Path IDentity (FPID) inputs determine whether read transactions take the lower latency path or the normal path through a filter unit(CA7 bus filer)	0x1
2	FPID0_LCDC	RW	The Fast Path IDentity (FPID) inputs determine whether read transactions take the lower latency	0x1

Vangotech

			path or the normal path through a filter unit (LCDC bus filer)	
1	FPID1_BUS	RW	The Fast Path IDentity (FPID) inputs determine whether read transactions take the lower latency path or the normal path through a filter unit (NIC_1 bus filter CH1)	0x0
0	FPID0_BUS	RW	The Fast Path IDentity (FPID) inputs determine whether read transactions take the lower latency path or the normal path through a filter unit (NIC_1 bus filter CH0)	0x0

4.6.6.54. SPI020 Configure Register

Table 4-252. SPI020 Configure Register (Offset: 0x0000_810C)

Bit	Symbol	Access	Description	Default
8	SPI_addr_cyc	RW	SPI Address Cycle Selection for the XIP Port This signal is used to decide if the SPI Flash address byte number is at the XIP port. 1'b0: 3-byte address 1'b1: 4-byte address For the XIP port access, the value cannot be changed during the read data transfer.	0x0

4.6.6.55. SSP_SPI MUX SEL Register

Table 4-253. SSP_SPI MUX SEL Register (Offset: 0x0000_8110)

Bit	Symbol	Access	Description	Default
7	SSP7_SPI_SEL	RW	1 : Slave mode (IO Control Register set pin-mux mode 1) 0 : Master mode(IO Control Register set pin-mux mode 0)	0x0
6	SSP6_SPI_SEL	RW	-	0x0
5	SSP5_SPI_SEL	RW	-	0x0
4	SSP4_SPI_SEL	RW	-	0x0
3	SSP3_SPI_SEL	RW	-	0x0
2	SSP2_SPI_SEL	RW	-	0x0
1	SSP1_SPI_SEL	RW	-	0x0
0	SSP_SPI_SEL	RW	-	0x0

4.6.6.56. PLL7 Control Register

Table 4-254. PLL7 Control Register (Offset: 0x0000_8114)

Bit	Symbol	Access	Description	Default

Vangotech

31:22	-	-	Reserved	-
21:20	FS	RW	PLL7 CKOUT frequency range selection	0x2
19:13	NS	RW	PLL7 N value	0x20
12:8	MS	RW	PLL7 M value	0x1
5:4	CC	RW	PLL7 loop bandwidth selection	0x2
0	EN	RW	PLL7 enable signal 1: Enable PLL 0: Power down PLL	0x0

4.6.6.57. PLL7 Lock Timer Register

Table 4-255. PLL7 Lock Timer Register (Offset: 0x0000_8118)

Bit	Symbol	Access	Description	Default
31:0	Lock timer	RW	PLL7 lock stable release timer	0x000005DC

4.6.6.58. Pin Mux with Mode X

Table 4-256. Pin Mux with Mode X (Offset: 0x0000_8204)

Bit	Symbol	Access	Description	Default
31:19	-	-	-	-
18	Mode_X_UART_8	RW	X_UART8_TX pin-mux by UART8_TX pin X_UART8_RX pin-mux by UART8_RX pin	0x0
17	Mode_X_SPI6_CS1	RW	X_RGMII_TXD[3] pin-mux by SPI6_CS1 pin	0x0
16	Mode_X_SPI6	RW	X_RGMII_TXD[2] pin-mux by SPI6_CS0 pin X_RGMII_TXD[1] pin-mux by SPI6_CLK pin X_RGMII_TXD[0] pin-mux by SPI6_MOSI pin X_RGMII_RX_CK pin-mux by SPI6_MISO pin	0x0
15	Mode_X_UART_9	RW	X_RGMII_RXD[1] pin-mux by UART9_TX pin X_RGMII_RXD[0] pin-mux by UART9_RX pin	0x0
14	Mode_X_UART_6	RW	X_RGMII_RX_CTL pin-mux by UART6_TX pin X_RGMII_RX_MDC pin-mux by UART6_RX pin	0x0
13	Mode_X_GPIO0_30	RW	X_UART7_RX pin-mux by	0x0

Vangotech

			GPIO0_30	
12	Mode_X_GPIO0_29	RW	X_UART7_TX pin-mux by GPIO0_29	0x0
11	Mode_X_GPIO0_26	RW	X_RMII_RXD[1] pin-mux by GPIO0_26	0x0
10	Mode_X_GPIO0_25	RW	X_I2S0_FS pin-mux by GPIO0_25	0x0
9	Mode_X_GPIO0_24	RW	X_MII_TX_EN pin-mux by GPIO0_24	0x0
8	Mode_X_GPIO0_23	RW	X_I2S0_SCLK pin-mux by GPIO0_23	0x0
7	Mode_X_GPIO0_22	RW	X_I2S0_RXD pin-mux by GPIO0_22	0x0
6	Mode_X_GPIO0_21	RW	X_LC_DATA[5] pin-mux by GPIO0_21	0x0
5	Mode_X_GPIO0_20	RW	X_UART2_NCTS pin-mux by GPIO0_20	0x0
4	Mode_X_GPIO0_19	RW	X_UART2_NRTS pin-mux by GPIO0_19	0x0
3	Mode_X_GPIO0_18	RW	X_I2S0_TXD pin-mux by GPIO0_18	0x0
2	Mode_X_GPIO0_17	RW	X_LC_DATA[6] pin-mux by GPIO0_17	0x0
1	Mode_X_GPIO0_3	RW	X_UART5_RX pin-mux by GPIO0_3	0x0
0	Mode_X_GPIO0_2	RW	X_UART5_TX pin-mux by GPIO0_2	0x0

Note: In pin-mux function, as mode X for a pad is selected, mode 0 ~ mode 7 would be ignored. While, please switch mode 0 ~ mode 7 to mode 0, or a mode that is not GPIO.

4.6.6.59. IO Control Register

Table 4-257. *IO Control Register (Offset: 0x0000_8500~0x0000_87F4)*

Bit	Symbol	Access	Description	Default
31:10	-	-	Reserved	-
9	SMT	RW	1: Schmitt-trigger	
8:5	DCSR	RW	[8]: Slew rate 1: Slow [7:6]: Output driving capability for EMMC pads 00: 10 mA at3.3 V , 4 mA at 1.8 V 01: 15 mA at 3.3 V , 6 mA at 1.8 V	Detail see Table 4-198

			10: 7.5 mA at 3.3 V, 3 mA at 1.8 V 11: 5 mA at 3.3 V, 2 mA at 1.8 V [7:6]: Output driving capability for normal pads 00: 4 mA 01: 8 mA 10: 12 mA 11: 16 mA [5]: Reserved	
4	PD	RW	1: 75K at 3.3 V Pull Down	
3	PU	RW	1: 75K at 3.3 V Pull Up	Table 4-198
2:0	SEL	RW	Pin mux mode	Table 4-198

4.6.6.60. SYS_XTAL Control Register

Table 4-258. SYS_XTAL Control Register (Offset: 0x0000_87F4)

Bit	Symbol	Access	Description	Default
3	EB	RW	Configure 0 when operation.	0x0
2:0	E	RW	Enable pin	Table 4-198

Vangotech

4.7. System Boot

4.7.1. Overview

Figure 4-26 shows the system boot process. First load SPL from eMMC, and load Kernel & rootfs from eMMC, finally to start the kernel.

4.7.2. System Boot Diagram

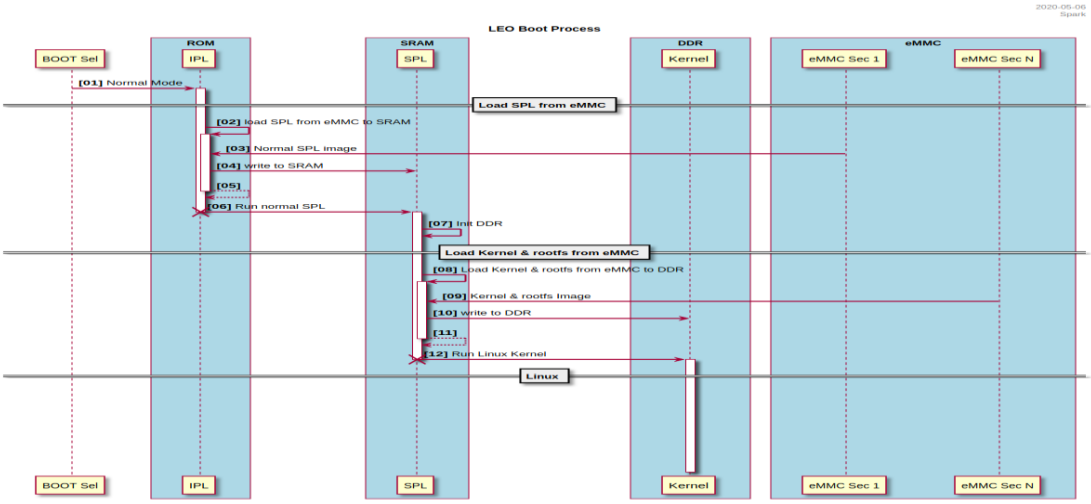


Figure 4-26. System Boot Diagram

4.7.3. Hardware Bootstrap Pin

Bootstrap pin is normal function output pin. During reset period, the system controller will make this function pin as input and get the strap value. After reset, this function pin goes back to its normal function. A pull-up resistor on this function pin will get high (1'b1) strap value. A pull-down resistor on this function pin will get low (1'b0) strap value.

Table 5-2. Hardware Bootstrap Pin

Bootstrap Selection	Original Function Pin	Description
Boot_IPL_SEL	X_USB2_DRVBUS	STRAP[8]: 0x0: Boot from SPI XIP mode 0x1: Boot from ROM (Default)
BOOT_SPL_SEL[1:0]	{X_UART0_TX, X_UART1_TX}	STRAP[10:9]: 00: SPL is from SPI NOR Flash. 01: SPL is from SPI NAND Flash. 10: SPL is from UART booting. 11: SPL is from eMMC.
Efuse_ck_sel	X_USB0_DRVBUS	STRAP[11]: 0: eFuse clock from internal OSC 1: eFuse clock from external XTAL
BOOT_CK_SEL[1:0]	{X_UART2_TX, X_UART3_TX}	STRAP[14:13]: 00: From XTAL 01: Internal OSC 30 MHz

		10: Internal OSC 60 MHz 11: Internal OSC 120 MHz
CPU_JTAG_SEL[1:0]	{ X_SPI0_DO,X_SPI1_DO}	STRAP[16:15]: 00: CA7 debug only 01: Reserved 10: CA7 daisy chain 11: Disable JTAG function
USB_DFU_DISABLE	X_SPI2_DO	STRAP[17]: 0: Enable USB DFU Boot 1: Disable the USB DFU Boot

Note:

STRAP[n] can be read from the SCU register offset 0x18, please refer to section 4.4

Vangotech

4.8. WDT

4.8.1. Overview

The FTWDT011 is an independent watchdog peripheral which offers a clocked by a dedicated clock through the external port and thus stays active event if the main clock fails. It can serve to detect and resolve malfunctions due to software failure, and to trigger system reset or an interrupt when the counter reaches a given timeout value.

4.8.2. Features

The WDT communication controller provides the following features:

- During timeout, outputs are one or a combination of the following signals:
  - System reset
  - System interrupt
  - External interrupt
- 32-bit down counter
- Internal or external clock source selection
- Clocked from external clock port and indent with APB bus clock
- A variable time-out period of reset
- Reset (if watchdog activated) when the down counter value of 0x000 is reached

4.8.3. Block Diagram

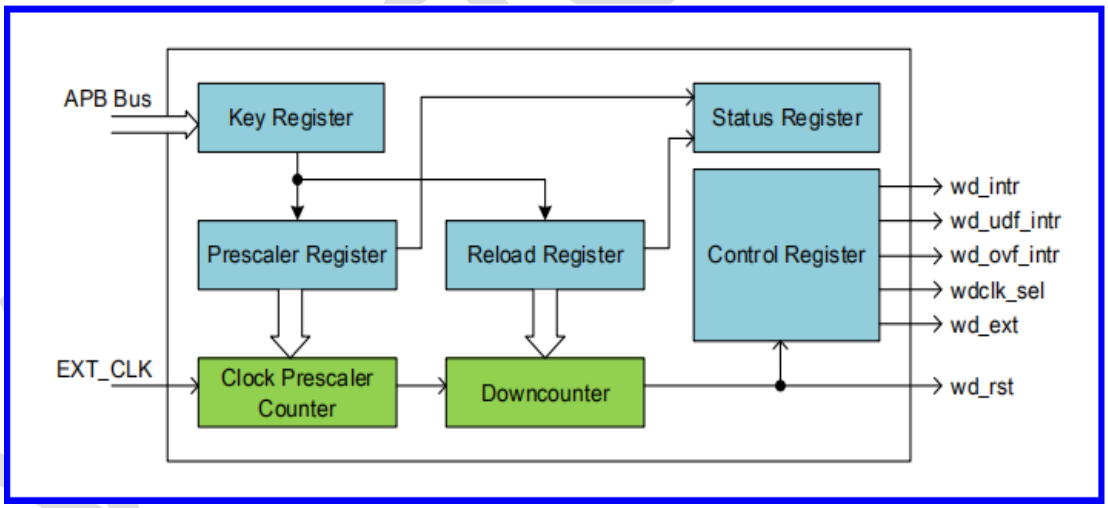


Figure 4-27. Functional Block Diagram

4.8.4. Register List

Module Name	Base Address
WDT_0	0x2810_0000
WDT_1	0x2820_0000
WDT_2	0x5460_0000
WDT_3	0x5470_0000

Base Offset (Hex)	Name	Type	Description	Default Value (Hex)
0x00	IWDT_KR	W	Key Register	0x0
	IWDT_CURR	R	Current Counter value	0xFFFF
0x04	IWDT_PR	R/W	Prescaler Register	0x0
0x08	IWDT_RLR	R/W	Reload Register	0xFFFF
0x0C	IWDT_SR	R	Status Register	0x0
0x10	IWDT_INTSR	R/W/C	Interrupt status	0x0
0x14	IWDT_CR	R/W	Control Register	0x0
0x18	IWDT_INTRLEN	R/W	Interrupt interval length register	0xFF
0x1C	IWDT_UDF	R/W	Watchdog counter under flow value	0x00
0x100	IWD_INTSTS	R	Interrupt status for each watchdog	0x00
0x104	IWDT_FEATURES	R	Features	Hwlinit
0x10C	IWDT_REVISION	R	Revision Tag	0x00010000

4.8.5. Register Description

4.8.5.1. Key Register

Table 4-259. Key Register

Bit	Name	Type	Description	Default Value
31:16			Reserved	
15:0	IWDT_KEY	RW	Key value (write only, read 0000h) These bits must be written by software at regular intervals with the key value 16'HAAAA, otherwise the watchdog generates a resetwhen the counter reaches 0. Writing the key value 16'h5555 to enable access to the IWDG_PR and IWDG_RLR registers. Writing the key value 16'hCCCC starts the watchdog (except if the hardware watchdog option is selected)	0x0

4.8.5.2. Prescaler Register

Table 4-260. Prescaler Register

Bit	Name	Type	Description	Default Value
31:16			Reserved	
2:0	PR	RW	Prescaler divider	0x0

Vangotech

			These bits are write access protected.They are written by software toselect the prescaler divider feeding the counter clock. PVU bit of IWDT_SR must be reset inorder to be able to change the prescaler divider. 000: divider /4 001: divider /8 010: divider /16 011: divider /32 100: divider /64 101: divider /128 110: divider /256 111: divider /256	

4.8.5.3. Reload Register

Table 4-261. Reload Register

Bit	Name	Type	Description	Default Value
31:N			Reserved	
N-1:0	RL	RW	Watchdog counter reload value These bits are written access protected. They are written by software to define the value to be loaded in the watchdog counter each time the value 16'HAAAA is written in the IWDG_KR register. The watchdog counter counts down from this value. The time out period is a function of this value and the clock prescaler. The RVU bit in the IWDT_SR register must be reset in order to be able to change the reload value.	All 1

4.8.5.4. Status Register

Table 4-262. Status Register

Bit	Name	Type	Description	Default Value
31:24			Reserved	
1	RVU	R	Watchdog counter reload value update This bit is set by hardware to indicate that an update of the reload value is ongoing.	0x0
0	PVU	R	Watchdog prescaler value update This bit is set by hardware to indicate that an update of the prescaler value is ongoing.	0x0

4.8.5.5. Interrupt Status Register

Table 4-263. Interrupt Status Register

vangotech

Bit	Name	Type	Description	Default Value
31:1			Reserved	
0	IntrStatus	RWC	The WdStatus register will record whether the Watch Dog timer reaches '0' or not. Write 1 to clear.	0

4.8.5.6. Control Register

Table 4-264. Control Register

Bit	Name	Type	Description	Default Value
31:9			Reserved	
8	IntrStatus	RWC	The WdStatus register will record whether the Watch Dog timer reaches '0' or not. Write 1 to clear.	0
7	WdUDFEn	RW	Watch Dog timer system reset enable bit for underflow function The bit is written access protected. The value 16'h5555 is written in the IWDG_KR register by software before enabling the wd_rst output by software programing. 0: Disable, 1: Enable	0
6	WdUDFIntEn	RW	Watch Dog timer system interrupt enable bit for underflow 0: Disable, 1: Enable	0
5			Reserved	
4	WdClock	RW	Watch Dog timer clock source bit It is a control register for wdclk_sel port. The bit is written access protected. The value 16'h5555 is written in the IWDG_KR register by software before changing the watchdog clock source by software programing. 0: Disable, 1: Enable	0
3	WdExt	RW	Watch Dog timer external signal enable bit The bit is written access protected. The value 16'h5555 is written in the IWDG_KR register by software before enabling the wd_ext output by software programing. 0: Disable, 1: Enable	0
2	WdRSt	RW	Watch Dog timer system reset enable bit The bit is written access protected. The value 16'h5555 is written in the IWDG_KR register by software before enabling the wd_rst output by software programing. 0: Disable, 1: Enable	0
1	WdIntr	RW	Watch Dog timer system interrupt enable bit 0: Disable, 1: Enable	0
0	WdEnable	R	Watch Dog timer enable bit	0

Vangotech

			0: Disable, 1: Enable	

4.8.5.7. Interrupter Length Register

Table 4-265. Interrupt Length Register

Bit	Name	Type	Description	Default Value
31:8			Reserved	
7:0	WdRSt	RW	The WdIntrlen register decides the duration of the assertion of wd_rst, wd_intr, and wd_ext signals. The default value is0xFF, which means that the default assertion durations of wd_rst, wd_intr, and wd_ext are 256 clock cycles. These bits are written access protected. The value 16'h5555 is written in the IWDG_KR register by software before programming WdIntrlen register by software programing.	0xFF

4.8.5.8. Under Flow Value Register

Table 4-266. Reload Register

Bit	Name	Type	Description	Default Value
31:N			Reserved	
N-1:0	UDF	RW	Watchdog counter under flow value These bits are writtenaccess protected. They are written by software todefine the value to be loaded in the watchdog counter each time the value 16'hAAAA is writtenin the IWDG_KR register. The watchdog counter counts down from this value. The timeoutperiod is a function of this value and the clock prescaler. The RVU bit in the IWDT_SR register must be reset in order to be able to change the reload value.	All 0

4.8.5.9. WdRevision Register

Table 4-267. WdRevision Register

Bit	Name	Type	Description	Default Value
31:0	WdRevision	R	Revision number of FTWDT011	0x00010100

4.9. PWMTMR

4.9.1. Overview

PWM timer is an APB device which provides four independent timers used to generate internal interrupts to CPU and to trigger DMA transfers. Each timer uses the internal system clock (PCLK of APB) or external

clock (ext_clk), and supports the PWM (Pulse Width Modulation) function to generate the PWM signals for the motor control or power level control.

For synchronization, it is recommended that the clocks, PCLK and ext_clk, must maintain the relationship: ext_clk cycle time/PCLK cycle time > 3.

4.9.2. Features

- Supports AMBA 2.0 APB bus
- Supports 4 independent 32-bit timers
- Selects internal or external clock source for each timer
- Supports PWM function to generate each timer
- Programmable duty control and polarity of PWM
- Supports auto-reload mode and one-shot pulse mode
- Dead-zone generator on PWM outputs
- Supports timer based on DMA transfer request

4.9.3. Block Diagram

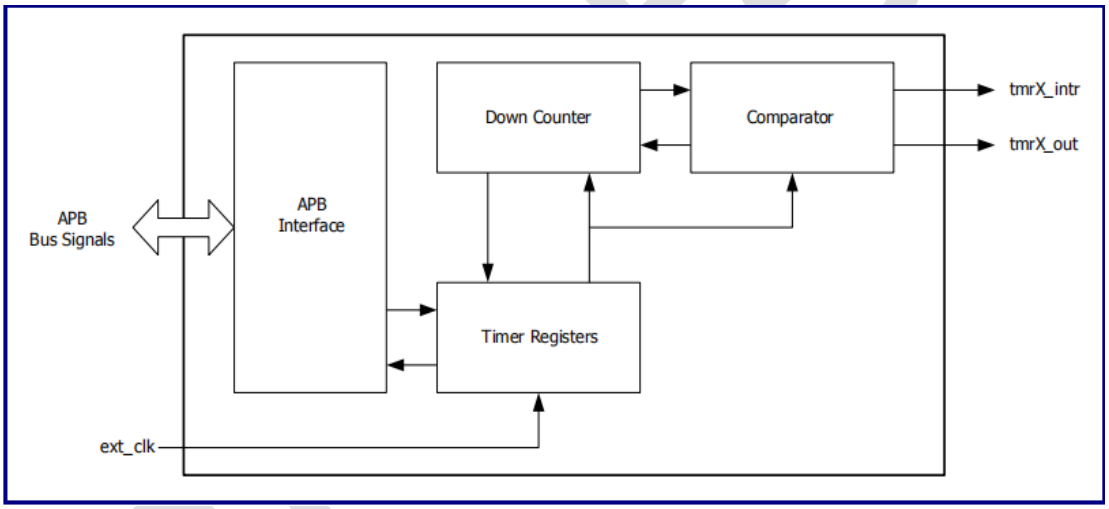


Figure 4-28. PWMTMR Block Diagram

4.9.4. Operations and Functional Descriptions

4.9.4.1. Basic Timer Operation

The procedure of timer operation is depicted in Figure 4-29.

1. Enable the auto-reload feature (TMX_CTRL[TMX_AUTO]). Set TMX_CNTB as 160 (50+110) and set TMX_CMPB as 110. Set the manual update bit (TMX_CTRL[TMX_UPDATE]) and inverter bit on / off (TMX_CTRL[TMX_OUT_INV]). The manual update bit sets the values of the counter and comparator to TMX_CNTB and TMX_CMPB, separately.
2. Start the timer by setting the start bit and manual update bit off.
3. When the counter has the same value as the comparator, the logic level of tmrX_out will be changed from low to high.
4. When the counter reaches '0', the counter will automatically reload MX_CNTB. At the same time, the interrupt request will be generated and tmrX_out is changed from high to low.

5. In ISR (Interrupt Service Routine), TMX_CNTB and TMX_CMPB will be set as 80 (20 + 60) and 60, which are used for the next duration.
6. When the counter has the same value as the comparator, the logic level of tmrX_out will be changed from low to high.
7. When the counter reaches '0', the counter will automatically reload TMX_CNTB. At the same time, the interrupt request will be generated and tmrX_out will be changed from high to low.
8. In ISR, auto-reload and interrupt request will be disabled to stop the timer.
9. When the counter has the same value as the comparator, the logic level of tmrX_out will be changed from low to high.
10. Even when the counter reaches '0', the counter will not be reloaded and the timer stops since the auto-reload feature is disabled.
11. No interrupt request is generated.

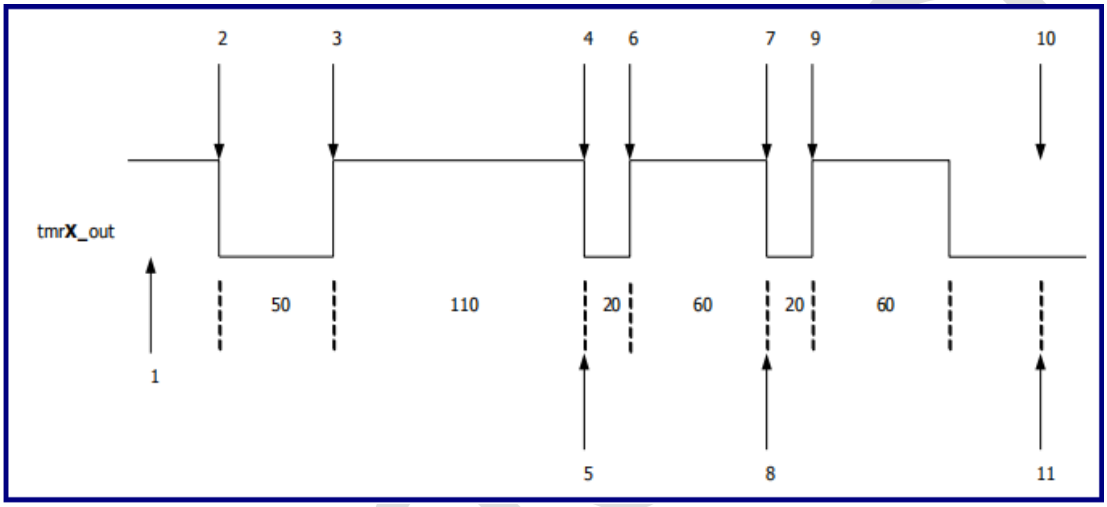


Figure 4-29. Example of Timer Operation

4.9.4.2. Auto-reload and Double Buffering

The timers have a double buffering feature, which can change the reload value for the next timer operation without stopping the current timer operation. Consequently, a current timer operation can be completed successfully even if the new time value is set.

The timer value can be written into TMX_CNTB (Timer count buffer register) and the current counter value of the timer can be read from TMX_CNTO (Timer count observation register). When TMX_CNTB reads, the value read is the reload value for the next timer duration, not the current state of the counter.

The auto-reload feature is used to copy TMX_CNTB into the counter when the counter reaches '0' and the auto-reload feature is enabled. When the counter reaches '0' and auto-reload feature is disabled, the counter will stop operating. This is called "one-shot operation".

Figure 4-30 shows an example of double buffering function.

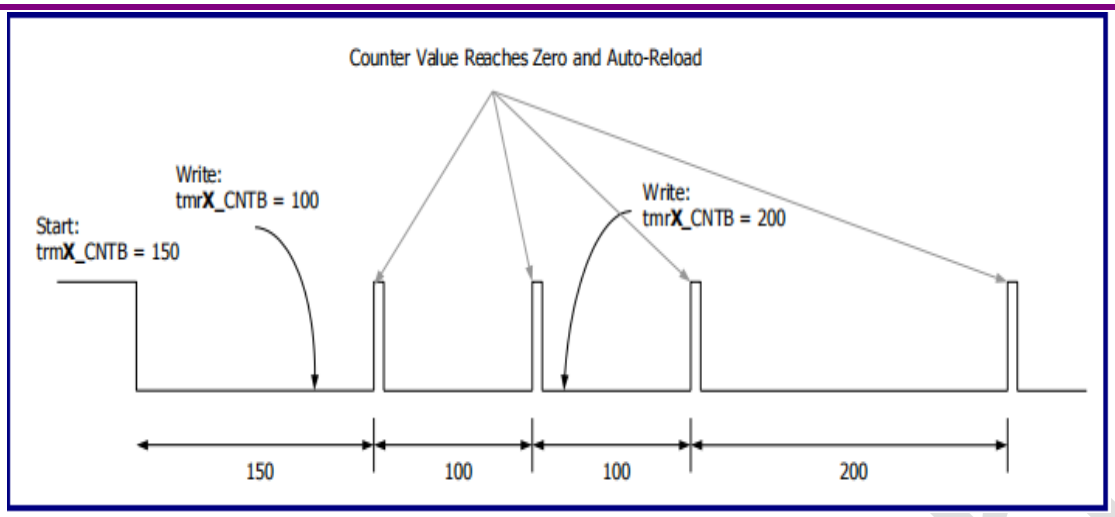


Figure 4-30. Example of Double Buffering Function

4.9.4.3. PWM Function

The PWM function can be implemented by using TMX_CMPB. When the output inverter, TMX_CTRL[TMX_OUT_INV],is disabled and the manual update or auto-reload of the counter is enabled, the timer output, tmrX_out, will go to low level. The output will go to high level when the counter value equals to the compare value. This mechanism implies that the PWM period is determined by TMX_CNTB and the pulse width is determined by TMX_CMPB.

For higher PWM value, users can increase the TMX_CMPB value. For lower PWM value, users can decrease the TMX_CMPB value. However, if the output inverter is enabled, the increment / decrement may be opposite.

The double buffering feature, TMX_CMPB, for a next PWM cycle can be written into any point of the current PWM cycle by ISR.

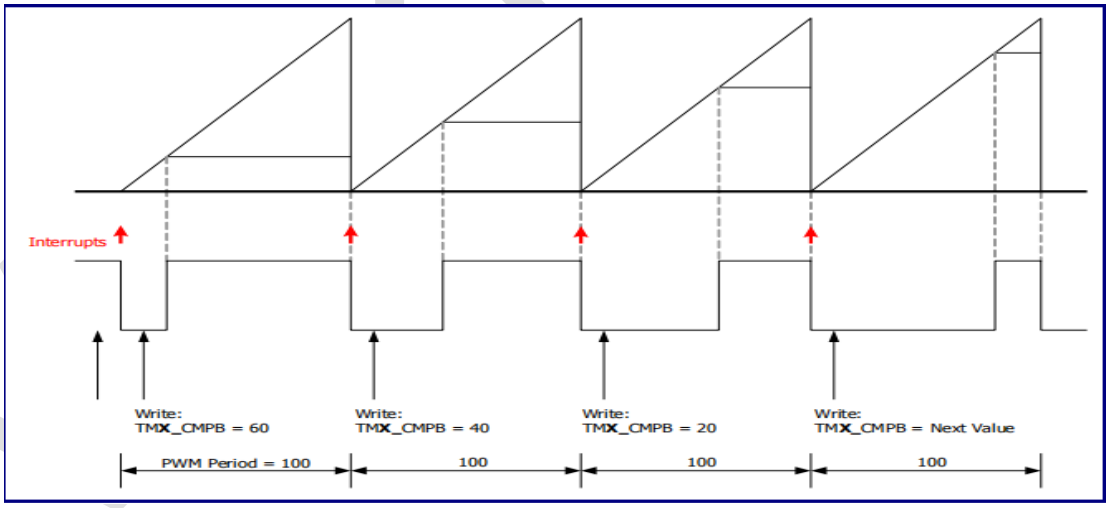


Figure 4-31. Example of PWM Function

4.9.4.4. Dead-zone Generation

The dead-zone is used for PWM to control the power devices. This feature is used to insert the time gap between the turn-off switching device and the turn-on switching device. This time gap prohibits from simultaneously turning on these two switching devices for a short period of time.

Figure 4-32 shows the waveform when the dead-zone feature is enabled. Where tmrX_out is the PWM output, tmrX_out_n will be the inversion of tmrX_out. When the dead-zone feature is enabled, the output waveforms of tmrX_out and tmrX_out_n will be tmrX_out_DZ and tmrX_out_n_DZ. These two signals can

never be simultaneously turned on by the dead-zone interval. For the functional correctness, the dead-zone length must be smaller than the compare counter value; otherwise, pulse with a pulse width of less than the dead-zone will vanish.

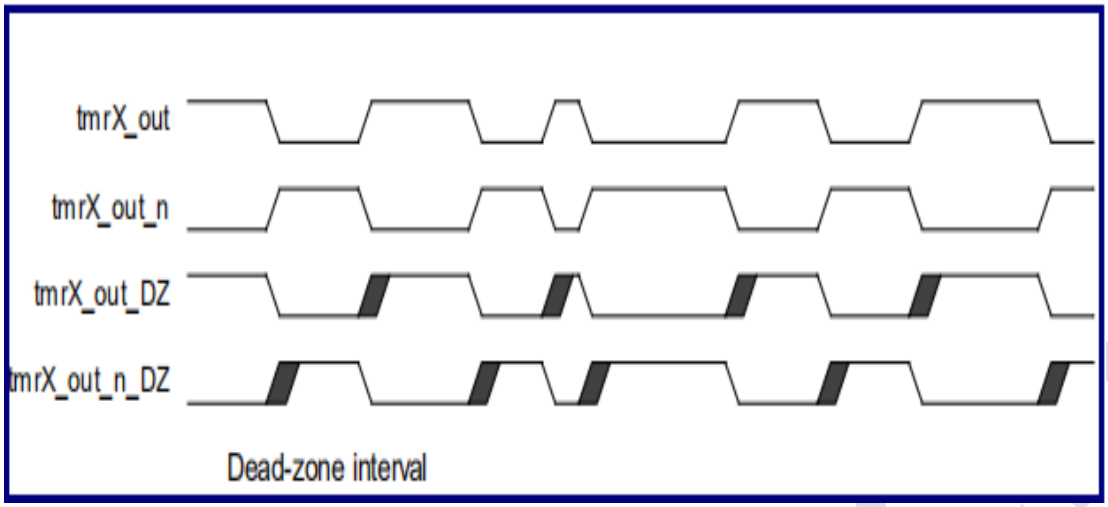


Figure 4-32. Waveform of Enabled Dead-zone Feature

4.9.4.5. DMA Request Mode

Instead of sending an interrupt at the end of a down-counter cycle, a timer can be configured to send a DMA request signal to one of the DMA channels. This mode allows the occurrence of a DMA transfer between a source and destination at the regular intervals. The idea is not to have the timer as the source or destination, but rather to have the timer to control the timing of data movement between two devices. The DMA request signal, tmrX_req, will be kept active high by the timer until the timer receives an ACK signal from the DMA unit. When the timer receives an ACK signal, the request signal will become inactive. Figure 4-33 shows how the DMA request remains active before the ACK signal is sent by DMA.

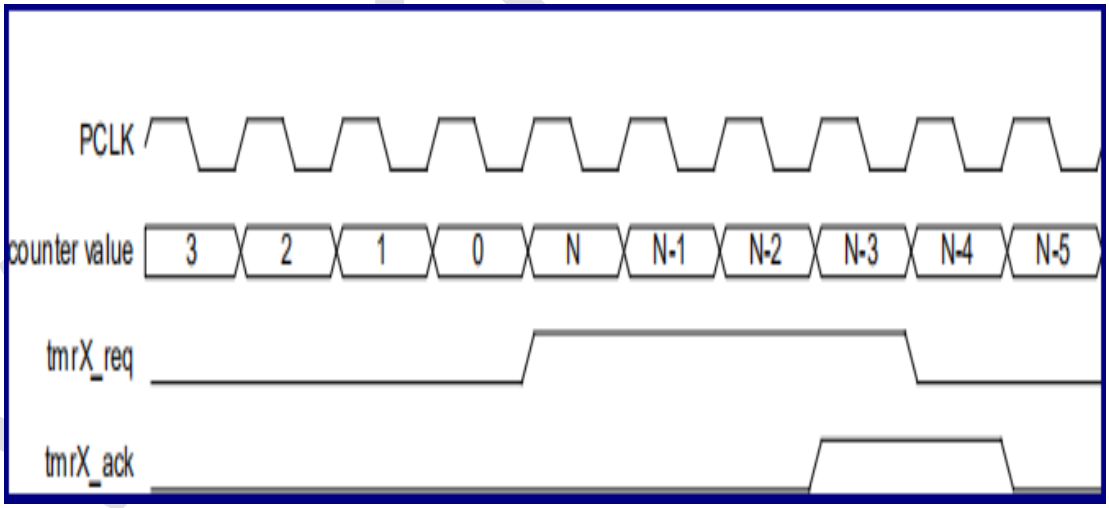


Figure 4-33. DMA Request and ACK

4.9.5. Register List

Module Name	Base Address
PWMTMR_0	0x5530_0000
PWMTMR_1	0x5540_0000
PWMTMR_2	0x56D0_0000
PWMTMR_3	0x56E0_0000

Register Name	Offset	Description
INT_CSTAT	000	Interrupt status and control register
TM1_CTRL	010	Timer 1 control register
TM1_CNTB	014	Timer 1 count buffer register
TM1_CMPB	018	Timer 1 compare buffer register
TM1_CNTO	01C	Timer 1 observation register
TM2_CTRL	020	Timer 2 control register
TM2_CNTB	024	Timer 2 count buffer register
TM2_CMPB	028	Timer 2 compare buffer register
TM2_CNTO	02C	Timer 2 observation register
TM3_CTRL	030	Timer 3 control register
TM3_CNTB	034	Timer 3 count buffer register
TM3_CMPB	038	Timer 3 compare buffer register
TM3_CNTO	03C	Timer 3 observation register
TM4_CTRL	040	Timer 4 control register
TM4_CNTB	044	Timer 4 count buffer register
TM4_CMPB	048	Timer 4 compare buffer register
TM4_CNTO	04C	Timer 4 observation register
TMR_REV	090	WMTMR revision register

4.9.6. Register Description

4.9.6.1. Interrupt Status and Control Register

The INT_CSTAT register shows individual interrupt status of each timer. If the global interrupt of FTPWMTMR010 is used by the system, CPU can check the timer that issues an interrupt by reading the INT_CSTAT register.

Table 4-268.Control Register (Offset: 0x00)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0
3	TM4_INT_STAT	RW1C	Timer 4 interrupt status This bit will auto clear in the pulse mode. Cleared by writing '1' to this bit in the level mode. 0: No effect 1: Counter value of Timer 4 reaches zero.	0
2	TM3_INT_STAT	RW1C	Timer 3 interrupt status This bit will auto clear in the pulse mode. Cleared by writing '1' to this bit in the level mode. 0: No effect	0

			1: Counter value of Timer 3 reaches zero.	
1	TM2_INT_STAT	RW1C	Timer 2 interrupt status This bit will auto clear in the pulse mode. Cleared by writing '1' to this bit in the level mode. 0: No effect 1: Counter value of Timer 2 reaches zero.	0
0	TM1_INT_STAT	RW1C	Timer 1 interrupt status This bit will auto clear in the pulse mode. Cleared by writing '1' to this bit in the level mode. 0: No effect 1: Counter value of Timer 1 reaches zero.	0

4.9.6.2. Timer Control Register

Table 4-269. Timer Control Register (TMX_CTRL (X = 1, 2, 3, 4), Offset = 0xX0)

Bit	Name	Type	Description	Default Value
[31:24]	TMX_DZ	RW	TimerX dead-zone length = TMX_CTRL[31:24] - 1	0
[23:9]	-	Rsvd	Reserved	0
8	TMX_PWM_EN	RW	TimerX PWM function enable/disable 0: Disable 1: Enable	0
7	TMX_DMA_EN	RW	TimerX DMA request mode enable/disable 0: Disable 1: Enable	0
6	TMX_INT_MO	RW	TimerX interrupt mode 0: Level (Clear an interrupt by writing '1' to TMX_INT_STAT) 1: Pulse	0
5	TMX_INT_EN	RW	TimerX interrupt mode enable/ disable 0: Disable 1: Enable	0
4	TMX_AUTO	RW	TimerX auto-reload on/ off 0: One-shot 1: Interval mode (Auto-reload)	0
3	TMX_OUT_INV	RW	TimerX output inverter on/off 0: Inverter off 1: Inverter on	0
2	TMX_UPDATE	RW	TimerX updates TMX_CNTB and TMX_CMPB. Write '1' to this register will trigger a manual update. This bit will be auto-cleared after the	0

Vangotech

			operation. 0: No operation 1: Manual update	
1	TMX_START	RW	TimerX start/stop 0: Stop 1: Start	0
0	TMX_CLK_SRC	RW	TimerX clock source 0: PCLK 1: ext_clk	0

4.9.6.3. Timer Counting Value Buffer Register

Table 4-270. Timer Counting Value Buffer Register (TMX_CNTB (X = 1, 2, 3, 4), Offset = 0xX4)

Bit	Name	Type	Description	Default Value
[31:0]	TMX_CNTB	RW	Start value of the TimerX counter	0

4.9.6.4. Timer Counter Compare Buffer Register

Table 4-271. Timer Counter Compare Buffer Register (TMX_CMPB (X = 1, 2, 3, 4), Offset = 0xX8)

Bit	Name	Type	Description	Default Value
[31:0]	TMX_CMPB	RW	Compare value of the TimerX counter	0

4.9.6.5. Timer Count Observation Register

Table 4-272. Timer Count Observation Register (TMX_CNTO (X = 1, 2,…, 8), Offset = 0xXC)

Bit	Name	Type	Description	Default Value
[31:0]	TMX_CNTO	RO	Observe value of the TimerX counter	0

4.9.6.6. Revision Register

Table 4-273. Revision Register (TMR_REV, Offset = 0x90)

Bit	Name	Type	Description	Default Value
[31:0]	TMR_REV	RO	Revision of FTPWMTMR010	0x0001_0000

Vangotech

4.10. GIC

The complete list of interrupt sources and the interrupt controller inputs to which they are assigned is shown in Table 4-274.

Table 4-274. Internal Interrupt Vectors for CA7

IRQ Source	IRQ Source Name	IRQ Controller Instance	IRQ Controller Pin Number	Inv ?	IRQ #	IRQ Trigger Mode	IRQ Trigger Level
u_SYSC	scu_intr	u_Glue_sync_ca7	IRQS[0]	-	IRQ 0	-	-
u_GPIO	gpio_intr	u_Glue_sync_ca7	IRQS[1]	-	IRQ 1	-	-
u_GPIO_u1	gpio_intr	u_Glue_sync_ca7	IRQS[2]	-	IRQ 2	-	-
u_WDT	wd_intr	u_Glue_sync_ca7	IRQS[3]	-	IRQ 3	-	-
u_WDT_u1	wd_intr	u_Glue_sync_ca7	IRQS[4]	-	IRQ 4	-	-
u_WDT_u2	wd_intr	u_Glue_sync_ca7	IRQS[5]	-	IRQ 5	-	-
u_WDT_u3	wd_intr	u_Glue_sync_ca7	IRQS[6]	-	IRQ 6	-	-
u_MAC	mac_intr	u_Glue_sync_ca7	IRQS[7]	-	IRQ 7	-	-
u_MAC_u1	mac_intr	u_Glue_sync_ca7	IRQS[8]	-	IRQ 8	-	-
u_MAC_u2	mac_intr	u_Glue_sync_ca7	IRQS[9]	-	IRQ 9	-	-
u_I2S	ssp_intr	u_Glue_sync_ca7	IRQS[10]	-	IRQ 10	-	-
u_I2S_u1	ssp_intr	u_Glue_sync_ca7	IRQS[11]	-	IRQ 11	-	-
u_SPI	ssp_intr	u_Glue_sync_ca7	IRQS[12]	-	IRQ 12	-	-
u_SPI_u1	ssp_intr	u_Glue_sync_ca7	IRQS[13]	-	IRQ 13	-	-
u_SPI_u2	ssp_intr	u_Glue_sync_ca7	IRQS[14]	-	IRQ 14	-	-
u_SPI_u3	ssp_intr	u_Glue_sync_ca7	IRQS[15]	-	IRQ 15	-	-
u_SPI_u4	ssp_intr	u_Glue_sync_ca7	IRQS[16]	-	IRQ 16	-	-
u_SPI_u5	ssp_intr	u_Glue_sync_ca7	IRQS[17]	-	IRQ 17	-	-
u_SPI_u6	ssp_intr	u_Glue_sync_ca7	IRQS[18]	-	IRQ 18	-	-
u_SPI_u7	ssp_intr	u_Glue_sync_ca7	IRQS[19]	-	IRQ 19	-	-
u_PWMTMR	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[20]	-	IRQ 20	-	-
u_PWMTMR	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[21]	-	IRQ 21	-	-
u_PWMTMR	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[22]	-	IRQ 22	-	-
u_PWMTMR	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[23]	-	IRQ 23	-	-

Vangotech

u_PWMTMR	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[24]	-	IRQ 24	-	-
u_PWMTMR_u1	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[25]		IRQ 25		
u_PWMTMR_u1	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[26]	-	IRQ 26	-	-
u_PWMTMR_u1	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[27]	-	IRQ 27	-	-
u_PWMTMR_u1	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[28]	-	IRQ 28	-	-
u_PWMTMR_u1	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[29]	-	IRQ 29	-	-
u_PWMTMR_u2	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[30]	-	IRQ 30	-	-
u_PWMTMR_u2	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[31]	-	IRQ 31	-	-
u_PWMTMR_u2	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[32]	-	IRQ 32	-	-
u_PWMTMR_u2	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[33]	-	IRQ 33	-	-
u_PWMTMR_u2	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[34]	-	IRQ 34	-	-
u_PWMTMR_u3	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[35]	-	IRQ 35	-	-
u_PWMTMR_u3	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[36]	-	IRQ 36	-	-
u_PWMTMR_u3	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[37]	-	IRQ 37	-	-
u_PWMTMR_u3	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[38]	-	IRQ 38	-	-
u_PWMTMR_u3	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[39]	-	IRQ 39	-	-
u_IIC	isi2c	u_Glue_sync_ca7	IRQS[40]	-	IRQ 40	-	-
u_IIC_u1	isi2c	u_Glue_sync_ca7	IRQS[41]	-	IRQ 41	-	-
u_IIC_u2	isi2c	u_Glue_sync_ca7	IRQS[42]	-	IRQ 42	-	-
u_IIC_u3	isi2c	u_Glue_sync_ca7	IRQS[43]	-	IRQ 43	-	-
u_IIC_u4	isi2c	u_Glue_sync_ca7	IRQS[44]	-	IRQ 44	-	-
u_UART	irda_intr1	u_Glue_sync_ca7	IRQS[45]	-	IRQ 45	-	-
u_UART	irda_intr2	u_Glue_sync_ca7	IRQS[46]	-	IRQ 46	-	-

Vangotech

u_UART_u1	irda_intr1	u_Glue_sync_ca7	IRQS[47]	-	IRQ 47	-	-
u_UART_u1	irda_intr2	u_Glue_sync_ca7	IRQS[48]	-	IRQ 48	-	-
u_UART_u2	irda_intr1	u_Glue_sync_ca7	IRQS[49]	-	IRQ 49	-	-
u_UART_u2	irda_intr2	u_Glue_sync_ca7	IRQS[50]	-	IRQ 50	-	-
u_UART_u3	irda_intr1	u_Glue_sync_ca7	IRQS[51]	-	IRQ 51	-	-
u_UART_u3	irda_intr2	u_Glue_sync_ca7	IRQS[52]	-	IRQ 52	-	-
u_UART_u4	irda_intr1	u_Glue_sync_ca7	IRQS[53]	-	IRQ 53	-	-
u_UART_u4	irda_intr2	u_Glue_sync_ca7	IRQS[54]	-	IRQ 54	-	-
u_UART_u5	irda_intr1	u_Glue_sync_ca7	IRQS[55]	-	IRQ 55	-	-
u_UART_u5	irda_intr2	u_Glue_sync_ca7	IRQS[56]	-	IRQ 56	-	-
u_UART_u6	irda_intr1	u_Glue_sync_ca7	IRQS[57]	-	IRQ 57	-	-
u_UART_u6	irda_intr2	u_Glue_sync_ca7	IRQS[58]	-	IRQ 58	-	-
u_UART_u7	irda_intr1	u_Glue_sync_ca7	IRQS[59]	-	IRQ 59	-	-
u_UART_u7	irda_intr2	u_Glue_sync_ca7	IRQS[60]	-	IRQ 60	-	-
u_UART_u8	irda_intr1	u_Glue_sync_ca7	IRQS[61]	-	IRQ 61	-	-
u_UART_u8	irda_intr2	u_Glue_sync_ca7	IRQS[62]	-	IRQ 62	-	-
u_UART_u9	irda_intr1	u_Glue_sync_ca7	IRQS[63]	-	IRQ 63	-	-
u_UART_u9	irda_intr2	u_Glue_sync_ca7	IRQS[64]	-	IRQ 64	-	-
u_DDR3030	ddr_int	u_Glue_sync_ca7	IRQS[65]	-	IRQ 65	-	-
u_DDR3030	ecc_int	u_Glue_sync_ca7	IRQS[66]	-	IRQ 66	-	-
u_SDC021_EMMC	sdc_int	u_Glue_sync_ca7	IRQS[67]	-	IRQ 67	-	-
u_SDC021_EMMC_u1	sdc_int	u_Glue_sync_ca7	IRQS[68]	-	IRQ 68	-	-
u_SPI020	intr	u_Glue_sync_ca7	IRQS[69]	-	IRQ 69	-	-
u_CAN_u0	ir_n	u_Glue_sync_ca7	IRQS[70]	Y	IRQ 70	-	-
u_CAN_u1	ir_n	u_Glue_sync_ca7	IRQS[71]	Y	IRQ 71	-	-
u_KBC	kb_int	u_Glue_sync_ca7	IRQS[72]	-	IRQ 72	-	-
u_KBC	kb_padint	u_Glue_sync_ca7	IRQS[73]	-	IRQ 73	-	-
-	-	u_Glue_sync_ca7	IRQS[74]	-	IRQ 74	-	-
-	-	u_Glue_sync_ca7	IRQS[75]	-	IRQ 75	-	-
u_ADCC	adc_int	u_Glue_sync_ca7	IRQS[76]	-	IRQ 76	-	-
u_TDCC	adc_int	u_Glue_sync_ca7	IRQS[77]	-	IRQ 77	-	-
-	-	u_Glue_sync_ca7	IRQS[78]	-	IRQ 78	-	-
-	-	u_Glue_sync_ca7	IRQS[79]	-	IRQ 79	-	-
-	-	u_Glue_sync_ca7	IRQS[80]	-	IRQ 80	-	-
u_FOTG210	sys_int_n	u_Glue_sync_ca7	IRQS[81]	Y	IRQ 81	-	-
u_FOTG210_u1	sys_int_n	u_Glue_sync_ca7	IRQS[82]	Y	IRQ 82	-	-
u_FOTG210_u2	sys_int_n	u_Glue_sync_ca7	IRQS[83]	Y	IRQ 83	-	-

Vangotech

u_LCDC210	LC_INTR	u_Glue_sync_ca7	IRQS[84]	-	IRQ 84	-	-
u_LCDC210	LC_MERRINTR	u_Glue_sync_ca7	IRQS[85]	-	IRQ 85	-	-
u_LCDC210	LC_FURINTR	u_Glue_sync_ca7	IRQS[86]	-	IRQ 86	-	-
u_LCDC210	LC_BAUPDINTR	u_Glue_sync_ca7	IRQS[87]	-	IRQ 87	-	-
u_LCDC210	LC_VSTATUSINTR	u_Glue_sync_ca7	IRQS[88]	-	IRQ 88	-	-
u_EMC030_ROM	intr	u_Glue_sync_ca7	IRQS[89]	-	IRQ 89	-	-
u_EMC030_SRAM	intr	u_Glue_sync_ca7	IRQS[90]	-	IRQ 90	-	-
u_TZC400_1F_ca7_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[91]	-	IRQ 91	-	-
u_TZC400_1F_lcdc_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[92]	-	IRQ 92	-	-
u_TZC400_2F_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[93]	-	IRQ 93	-	-
u_ba431	IRQ_TRNG	u_Glue_sync_ca7	IRQS[94]	-	IRQ 94	-	-
u_cryptomaster_ahb	IRQ	u_Glue_sync_ca7	IRQS[95]	-	IRQ 95	-	-
u_CA7Q	COMMRX[0]	u_Glue_sync_ca7	IRQS[96]	-	IRQ 96	-	-
u_CA7Q	COMMRX[1]	u_Glue_sync_ca7	IRQS[97]	-	IRQ 97	-	-
u_CA7Q	COMMRX[2]	u_Glue_sync_ca7	IRQS[98]	-	IRQ 98	-	-
u_CA7Q	COMMRX[3]	u_Glue_sync_ca7	IRQS[99]	-	IRQ 99	-	-
u_CA7Q	COMMTX[0]	u_Glue_sync_ca7	IRQS[100]	Y	IRQ 100	-	-
u_CA7Q	COMMTX[1]	u_Glue_sync_ca7	IRQS[101]	Y	IRQ 101	-	-
u_CA7Q	COMMTX[2]	u_Glue_sync_ca7	IRQS[102]	Y	IRQ 102	-	-
u_CA7Q	COMMTX[3]	u_Glue_sync_ca7	IRQS[103]	Y	IRQ 103	-	-
u_CA7Q	nAXIERRIR	u_Glue_sync_ca7	IRQS[104]	Y	IRQ 104	-	-

Vangotech

	Q						
u_CA7Q	nCTIIRQ[0]	u_Glue_sync_ca7	IRQS[105]	Y	IRQ 105	-	-
u_CA7Q	nCTIIRQ[1]	u_Glue_sync_ca7	IRQS[106]	Y	IRQ 106	-	-
u_CA7Q	nCTIIRQ[2]	u_Glue_sync_ca7	IRQS[107]	Y	IRQ 107	-	-
u_CA7Q	nCTIIRQ[3]	u_Glue_sync_ca7	IRQS[108]	Y	IRQ 108	-	-
u_CA7Q	nPMUIRQ[0]	u_Glue_sync_ca7	IRQS[109]	Y	IRQ 109	-	-
u_CA7Q	nPMUIRQ[1]	u_Glue_sync_ca7	IRQS[110]	Y	IRQ 110	-	-
u_CA7Q	nPMUIRQ[2]	u_Glue_sync_ca7	IRQS[111]	Y	IRQ 111	-	-
u_CA7Q	nPMUIRQ[3]	u_Glue_sync_ca7	IRQS[112]	Y	IRQ 112	-	-
u_DMAC020	dmaint	u_Glue_sync_ca7	IRQS[113]	-	IRQ 113	-	-
u_DMAC020	dmaint_tc	u_Glue_sync_ca7	IRQS[114]	-	IRQ 114	-	-
u_DMAC020	dmaint_err	u_Glue_sync_ca7	IRQS[115]	-	IRQ 115	-	-
u_DMAC030	DMACIntr	u_Glue_sync_ca7	IRQS[116]	-	IRQ 116	-	-
u_DMAC030	DMACIntr_err	u_Glue_sync_ca7	IRQS[117]	-	IRQ 117	-	-
u_DMAC030	DMACIntr_tc	u_Glue_sync_ca7	IRQS[118]	-	IRQ 118	-	-
u_DMAC030_u1	DMACIntr	u_Glue_sync_ca7	IRQS[119]	-	IRQ 119	-	-
u_DMAC030_u1	DMACIntr_err	u_Glue_sync_ca7	IRQS[120]	-	IRQ 120	-	-
u_DMAC030_u1	DMACIntr_tc	u_Glue_sync_ca7	IRQS[121]	-	IRQ 121	-	-

4.11. DMAC for AXI

4.11.1. Overview

FTDMAC030 is a Direct Memory Access (DMA) controller with the AMBA AXI interface. This controller has not only up to eight configurable DMA channels and one AXI master port but also one APB slave port to access the registers. The local descriptor, SRAM, stores the descriptors; therefore, the channel controller can read the local descriptor from the local SRAM instead of from the main memory. When the local

Vangotech

descriptor, SRAM, is configured to be active, an additional AXI slave port will be exploited for writing SRAM.

4.11.2. Features

- Operation based on DMA descriptor
- Supports eight DMA channels
- Supports multiple transfer types
  - Memory-to-memory type
  - Memory-to-peripheral type
  - Peripheral-to-memory type
  - Peripheral-to-peripheral type
- Supports depth 64 of data buffer
- Supports depth 8 of read/write command queues
- Local descriptor memory to quickly access descriptors
- Includes watchdog timer to detect deadlock situation of each channel
- Supports 64-bit wide data buses for AXI master port and slave port
- Supports read data reordering of AXI master port
- APB slave port to access register file and channel descriptor
- Supports write-only mode to write constant value to specified memory range
- Supports unaligned transfer mode to allow transfer without source/destination address alignment restriction with maximum bus width for better performance
- Supports up to 33-bit AXI address bus

4.11.3. Block Diagram

Figure 4-34 shows a block diagram of the DMA.

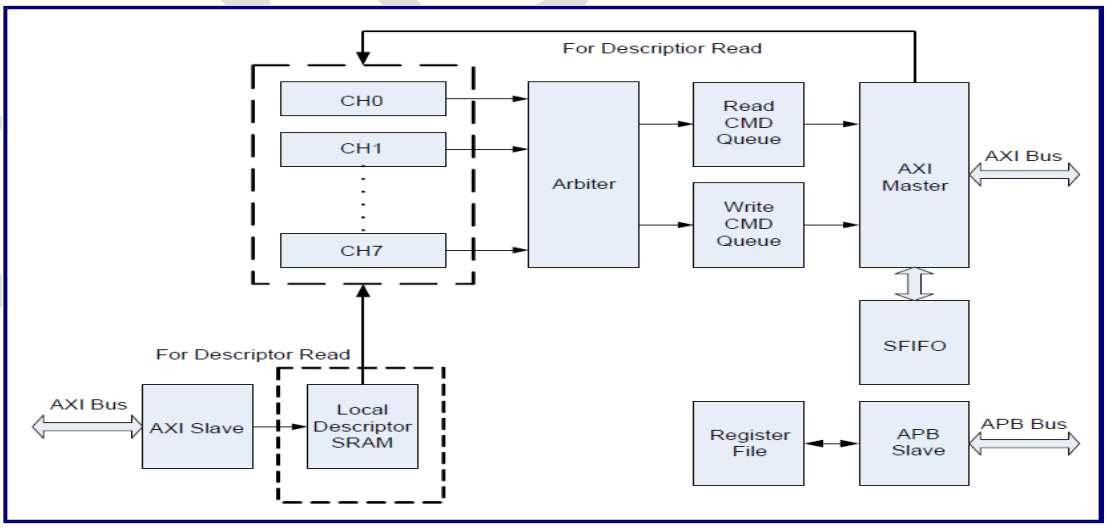


Figure 4-34. DMA Block Diagram

4.11.4. Programming Guidelines

4.11.4.1. AXI Master and SFIFO

The AXI master processes the DMA command from the command queue. First, it issues a read command

to the source and places the returned read data to the SFIFO. Once all the requested data are returned, the AXI master will issue the write command to the target. The AXI master is also responsible for the descriptor read command. The read operation is the same as processing the DMA command; however, the returned read data are put to the channel that requests this descriptor command and no write operation will be issued.

The AXI master supports the read data reordering, that is, the AXI master uses different AXI IDs to issue the read command from different DMA channels and the read data can be returned without the order of the command sequence. For example, both DMA channel 0 and DMA channel 1 generate one DMA command and push into the command queue. The AXI master then uses ID 0 to issue the read command for channel 0 and ID 1 to issue the read command for channel 1. Even though the ID 0 read command is issued first, the ID 1 read data can be accepted by the AXI master before receiving the ID 0 read data. This feature reduces the impact of the high-latency device.

The write interleaving is not supported by the AXI master. All write transactions are in-order even though the write command, AXI ID, is different. Table 4-275 shows the relationship between the DMA channel and the AXI ID of read/write or descriptor read transactions.

Table 4-275. Relationship between AXI ID and DMA Channels

DMA Channel Number	AXI ID of DMA Read/Write Transactions	AXI ID of Descriptor Read Transactions
0	0	8
1	1	9
2	2	10
3	3	11
4	4	12
5	5	13
6	6	14
7	7	15

The SFIFO is a shared data storage element for storing the DMA transfer data read from the source. The space in SFIFO is shared by all DMA channels. For the FIFO architecture, the data pushed to the tail and popped from the head is not traditional. Instead, the data can be pushed into any entry of SFIFO and can also be popped from any entry. Figure 4-35 shows the data movement in SFIFO. The ID 0 data and ID 1 data interlace in SFIFO. After all four beats of ID 1 data are received, the write command of ID 1 will be issued and the ID 1 data will be popped out without affecting the ID 0 data.

Vangotech

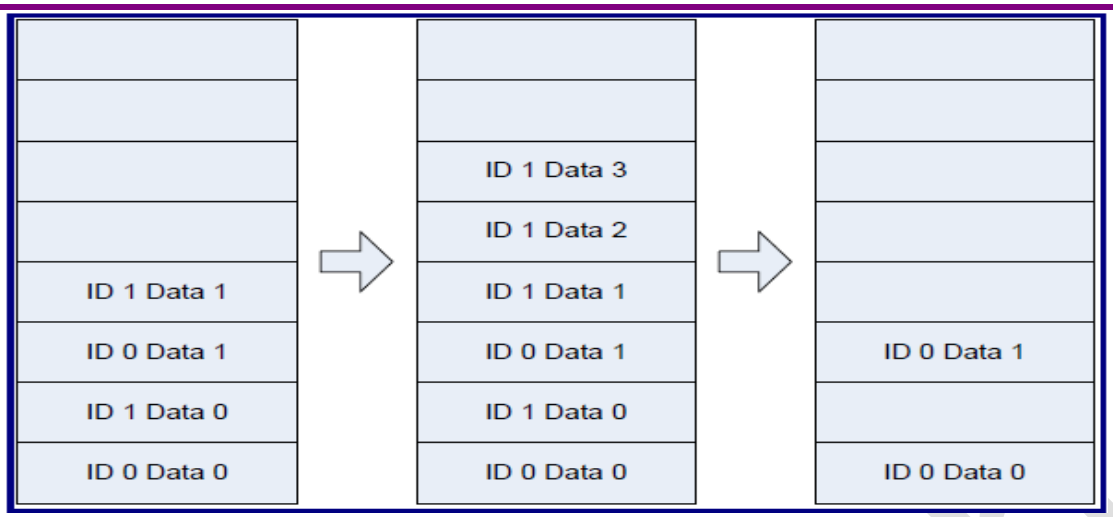


Figure 4-35. SFIFO Data Movement

4.11.4.2. APB Slave

The register file and the DMA channel register are programmed through the APB slave. The APB slave only supports the word access and the size of memory region is 1M bytes. While accessing the non-existent memory space, the PSlvErr will assert. The APB slave error can be turned off to prevent asserting of PSlvErr by programming the "PSlvErr Enable Register (PSE, Offset = 0x30)". This is useful in some debug cases.

The APB clock provided to the APB slave has no relationship with the AXI clock (Fully asynchronous). The only constraint is that the speed of the APB clock must not exceed the speed of the AXI clock.

4.11.4.3. AXI Slave and Local Descriptor Memory

The local descriptor memory is an additional SRAM for storing the DMA descriptors. By using LDM, the latency of the descriptor read can be significantly reduce compared to the read descriptor from the main memory in one system. LDM is written through the AXI slave. The sizes of the AXI transaction supported by the AXI slave can be word, double-word, and four word (4 bytes, 8 bytes, and 16 bytes). The supported AXI burst type is INCR. The content of the descriptor cannot be read from the AXI slave. Read to the AXI slave results all 0 data returned.

Each descriptor contains six words (24 bytes) or eight words (32 bytes). These two different types (six words or eight words) are determined by three different variables, X, Y, and Z of three different configurable items, DMAC_AXI_AW_X, DMAC_AXI_AWUSER_Y, and DMAC_AXI_ARUSER_Z, respectively. If X is larger than 32, Y does not equal to zero, or Z does not equal to zero, the descriptor is determined as the eight-word type. In contrast, if X = 32 and Y = Z = 0, the descriptor is determined as the six-word type. Besides, for ease of decoding, each descriptor stored in LDM is in a 32-byte aligned slot. Figure 4-36 depicts how the descriptors stored in a 32-entry LDM. The blue area is the expanded space of each descriptor when performing in the eight-word type. No matter what type of the descriptor is, it is always eight-word aligned. Please note that the blue area is not an available space in the six-word type.

Vangotech

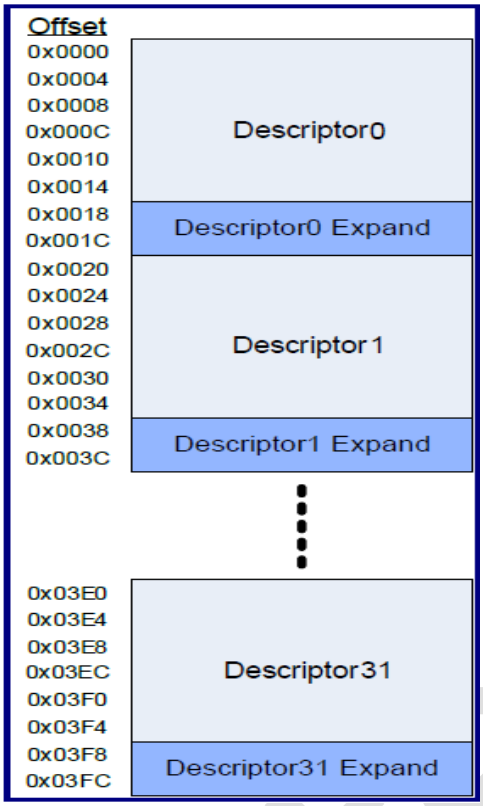


Figure 4-36. Descriptors Stored in LDM

The channel controller will fetch the descriptor from LDM if LLP hits the "Local Descriptor Memory Base". For example, if the local descriptor memory base is 0xA800_0000 and LLP is 0xA800_0020, the channel will fetch the descriptor 1 after the current descriptor finishes.

When a slot of LDM is written, the corresponding free space flag will be cleared. After the descriptor in the slot is fetched by a channel controller, the free space flag will be set again. The availability of LDM can be checked by the free space flag for the best utilization.

4.11.4.4. Channel Arbiter

The channel arbiter is dedicated for arbitrating. The command is pushed by the channel controller into the command queue. The channel arbiter adopts a two-level round-robin mechanism to perform the arbitration. Each channel can be programmed to be in the high-priority group or the low-priority group . Figure 4-37 shows how the two-level round-robin mechanism is performed. Assuming that all channel controllers 0 ~ 5 have commands needed to be issued and the channel controllers 0 ~ 2 are in the high-priority group and the rests are in the low-priority group, the sequence of arbitration will be:

CH0 → CH1 → CH2 → CH3 → CH0 → CH1 → CH2 → CH4 → CH0 → CH1 → CH2 → CH5 → …

The low-priority group can be considered as one member in the high-priority group.

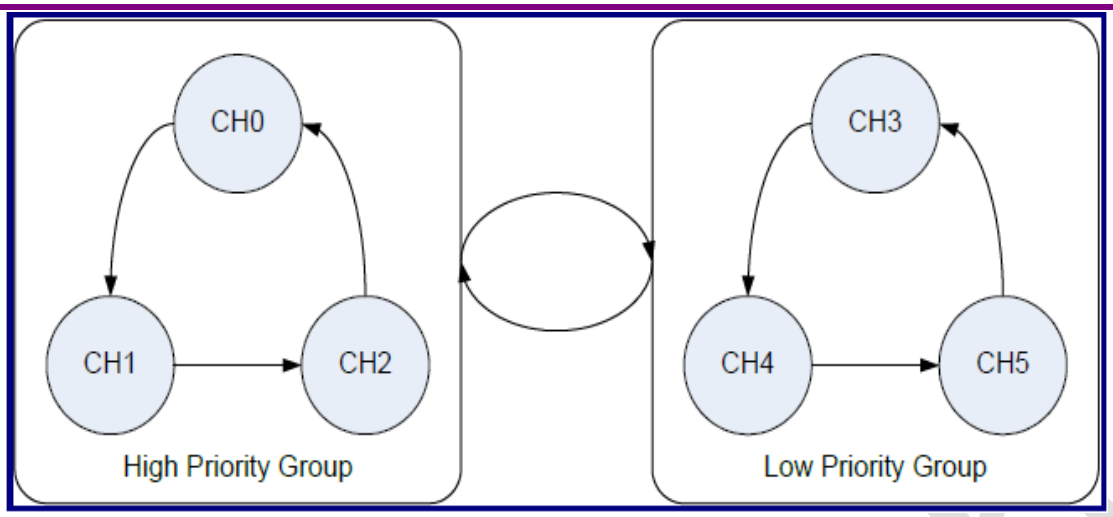


Figure 4-37. Two-level Round-robin Arbitration Mechanism

Once a channel controller is granted, the grant will be retrieved when the grant window value is exhausted or no more command is need to be issued. The grant window value unit is the length of the command (Number of beat on AXI bus). For example, if the grant window value is set to 64 for CH0, CH0 can issue commands to the command queue and the grant will be retrieved after the total length of the issued commands exceeds 64.

4.11.4.5. Channel Controller

There are up to eight channel controllers in FTDMAC030. Each channel controller contains a set of the registers that can be updated by directly programming through the APB slave or the descriptor read. The descriptor contains the information which is required for performing the DMA transfer. Figure 4-38 and Figure 4-39 show the structure of the descriptor. For the one-dimension structure, the total transfer size (XCNT) is 22-bit wide. For the two-dimension structure, the total transfer size is split into two parts (XCNT and YCNT) and both of them are 16-bit wide. The red frame lines in Figure 4-38 and Figure 4-39 represent the expanded field for the eight-word descriptor.

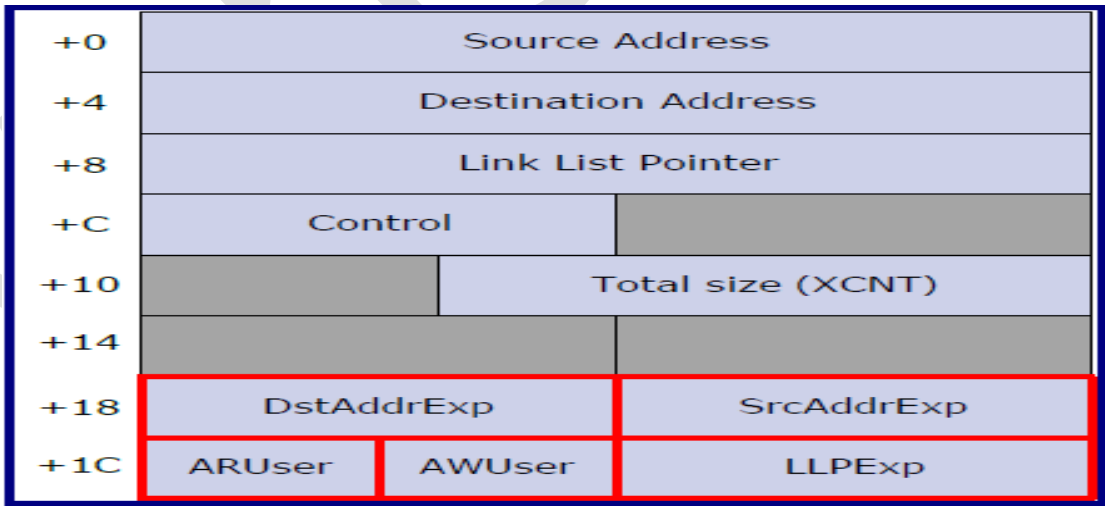


Figure 4-38. One-dimension Descriptor Structure

+0	Source Address		
+4	Destination Address		
+8	Link List Pointer		
+C	Control		Expanded Control
+10	YCNT		Total size (XCNT)
+14	DSTRIDE		SSTRIDE
+18	DstAddrExp		SrcAddrExp
+1C	ARUser	AWUser	LLPExp

Figure 4-39. Two-dimension Descriptor Structure

Please note that when performing in the six-word type descriptor, the offsets of the +18 and +1C field (i.e. SrcAddrExp, DstAddrExp, LLPExp, AWUser, and ARUser) are not available. On the other hand, users are only permitted to write the content to the offset of +18 and +1C when performing in the eight-word type descriptor. For the difference of the six-word type and eight-word type descriptors.

While preparing the descriptor in the main memory, it must be word-aligned if the AXI bus width is 32 bits; it must be double-word-aligned if the AXI bus width is 64 bits; and it must be 128 bits aligned if the AXI bus width is 128 bits.

4.11.4.6. Link List Pointer

The descriptors can be chained by using the link list pointer. When a channel controller finishes performing the current descriptor, it automatically uses the address recorded in the link list pointer to fetch the next descriptor. Figure 4-40 depicts the usage of the link list pointer. Once the descriptor 0 is finished, the channel controller either issues the read transaction on the AXI bus or reads from LDM to acquire the descriptor 1. When the descriptor 1 is finished, it will fetch the descriptor 2. When the descriptor 2 is finished, the channel controller will stop because the LLP values are all zeros. All zero value represents the end of the chain. It is suggested setting the TCMsk bit in the channel controller in the last descriptor so that the terminal count status and interrupt can be issued after finishing the last descriptor in one chain.

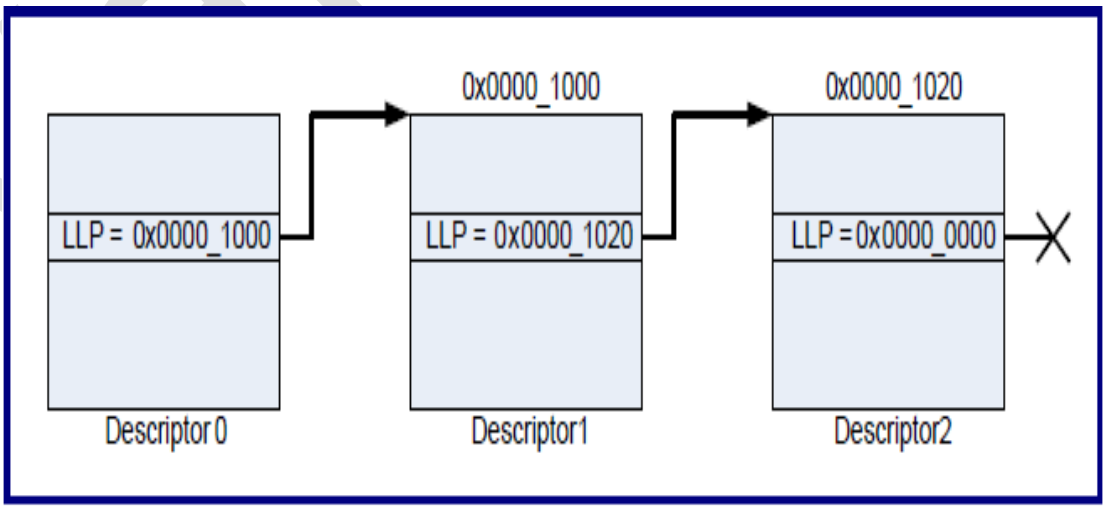


Figure 4-40. Usage of Link List Pointer

4.11.4.7. DMA Peripheral Interface

The DMA peripheral interface provides a handshake mechanism between FTDMAC030 and the peripheral devices. It includes two signals: dma_req (DMA request) and dma_ack (DMA acknowledge). When a channel controller is performing a descriptor with the handshake enabled (Either source or destination or both), it will wait for the dma_req signal before starting a transfer. When the peripheral device is ready for reading (Source) or writing (Destination) data, it will assert the dma_req signal. Afterwards, the channel controller will issue the DMA command. The count value of source transfer that decides the length of the transfer will be performed in one dma_req/dma_ack handshake. After transferring the desired length, the dma_req signal will de-assert when the DMA controller asserts the dma_ack signal. The dma_req signal can be asserted by the peripheral after the dma_ack signal is de-asserted.

Figure 4-41 depicts the usage of the peripheral interface with the count value of the source transfer as 32 and the total size as 64 (Source data width is 32 bits). Please note that when performing in the DMA handshake mode, the total transfer count must be the multiples of the source transfer count.

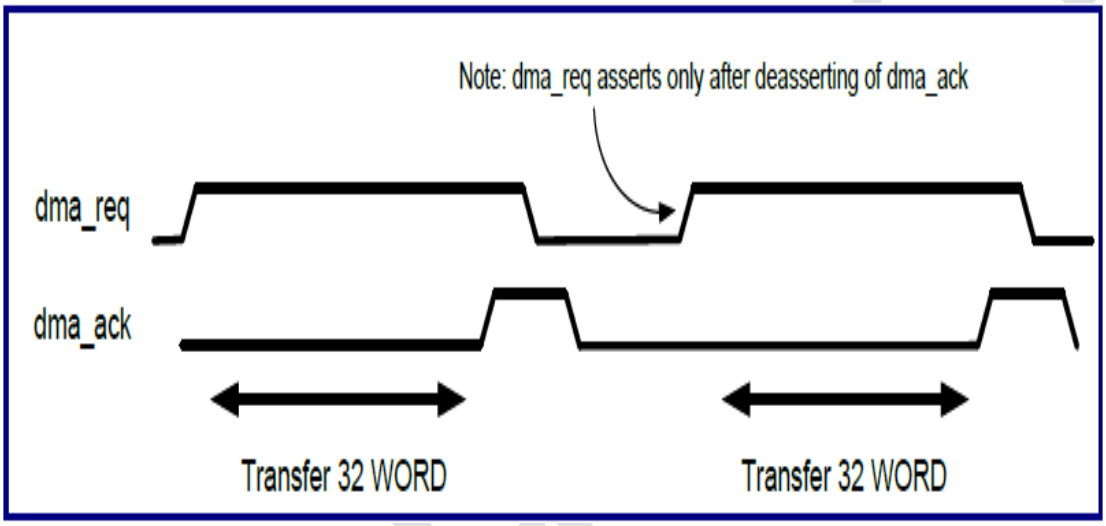


Figure 4-41. Peripheral Interface Protocol

There are up to 16 pairs of dma_req/dma_ack. Each pair can be connected to either the source or the destination and can be assigned to one of the channel controllers.

The source transfer count is only valid in one dimension mode. In the two-dimension transfer mode, one DMA handshake is related to a whole frame transfer. The source transfer count has no effect in the two-dimension transfer mode.

4.11.4.8. Example of One-dimension Transfer

The example in this section depicts a one-dimension transfer. Table 4-276 is an example of the descriptor content of the one-dimension DMA transfer.

Table 4-276. Descriptor Content of One-dimension DMA Transfer

Register	
Source address	0x0080h
Destination address	0x0880h
Linked list pointer	0x0000
Transfer size	XCNT = 40h
S/D stride	x

Table 4-277. Control Register of One-dimension DMA Transfer

Control Register		
[31:29]	SrcTcnt	x
[28]	TC_MSK	1'b0
[27:25]	SRC_WIDTH	010: 32 bits
[24:22]	DST_WIDTH	010: 32 bits
[21:20]	SRC_CTRL	00: INCR
[19:18]	DST_CTRL	00: INCR
[17]	WDTEn	x
[16]	CH_EN	0 1
[15]	EXPAND_EN	1'b0
[14]	2D_EN	x
[13]	WAIT_EVENT_EN	x
[12]	SET_EVENT_EN	x
[11:9]	ChSEvent	x
[8]	Wsync	x
[7:0]	EVENT	x

The DMA transfer manages to move the data from 0x0080 to 0x0880 with a length of 64 bytes. There is no chained descriptor after this descriptor. The total data transfer length is 256 bytes.

There are four DMA commands generated when the channel controller is processing this descriptor. These four DMA commands are issued by the AXI master as four read and four write commands. The length of each AXI command is 16. Please note that if the DSFIFO depth is 8, 16, or 32, the maximum length of the AXI transaction will be limited to 8. Figure 4-42 depicts the one-dimension DMA transfer.

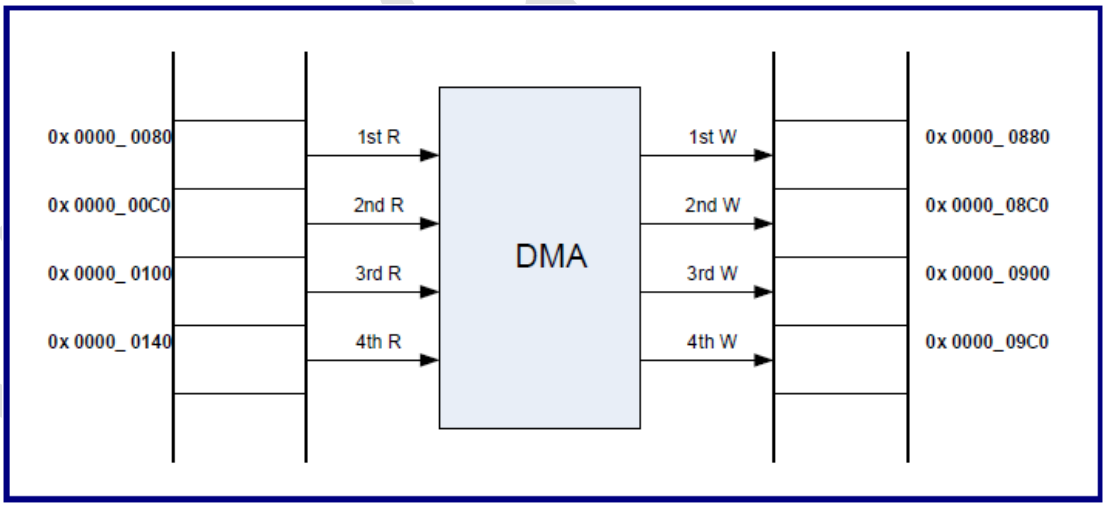


Figure 4-42. One-dimension DMA Transfer

4.11.4.9. Example of Two-dimension Transfer

The example in this section depicts a two-dimension DMA transfer. Table 4-278 is an example of the descriptor content of the two-dimension DMA transfer.

Table 4-278. Descriptor Content of Two-dimension DMA Transfer

Register	
Source address	0x0080h

Destination address	0x0880h
Linked list pointer	0x0000
Transfer size	XCNT = 32 YCNT = 2
S/D stride	SSTRIDE = 0x100h DSTRIDE = 0x100h

The transfer size contains two parts. XCNT represents the number of beats which will be transferred before loading the stride value. YCNT represents the total number of iterations. The stride value is used to reload the source and destination address when XCNT is reached. In this example, the source address is reloaded from 0x0080 (Add SSTRIDE 0x100) to 0x0180 and the destination address is reloaded from 0x0880 (Add DSTRIDE 0x100) to 0x0980. Please note that the stride is 2's complement value (Can be minus).

Table 4-279. Control Register of Two-dimension DMA Transfer

Control Register		
[31:29]	SrcTcnt	x
[28]	TC_MSK	1'b0
[27:25]	SRC_WIDTH	000: 8 bits
[24:22]	DST_WIDTH	000: 8 bits
[21:20]	SRC_CTRL	00: INCR
[19:18]	DST_CTRL	00: INCR
[17]	WDTEn	x
[16]	CH_EN	0 → 1
[15]	EXPAND_EN	1'b1
[14]	2D_EN	1'b1
[13]	WAIT_EVENT_EN	x
[12]	SET_EVENT_EN	x
[11:9]	ChSEvent	x
[8]	Wsync	x
[7:0]	ChWEvent	x

Please note that the 2D_EN bit is set to '1' to enable the two-dimension DMA transfer. Figure 4-43 depicts the data transfer scheme.

Vangotech

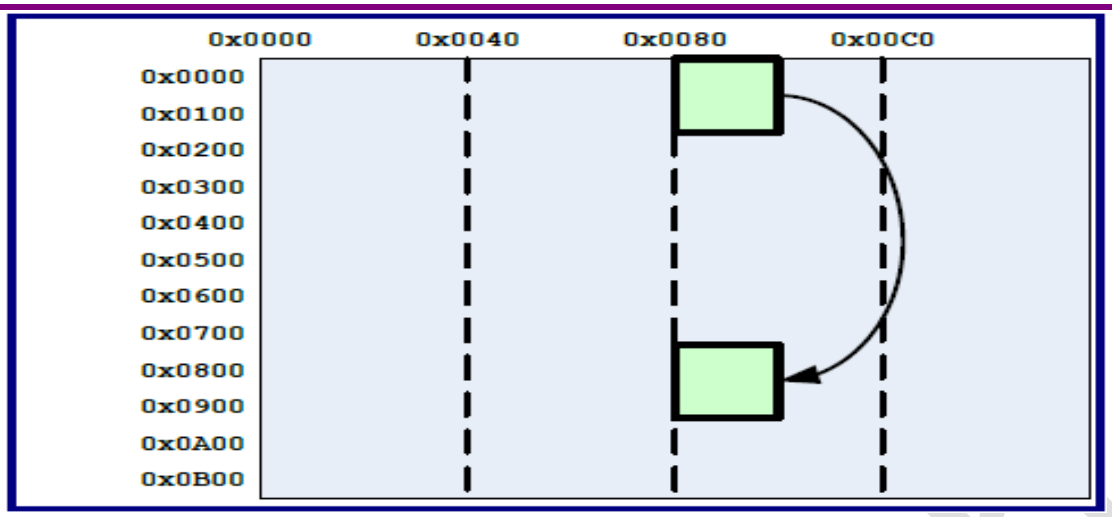


Figure 4-43. Two-dimension DMA Transfer

The data arrays from (0x0080 → 0x009F and 0x0180 → 0x019F) are transferred to (0x0880 → 0x089F and 0x0980 → 0x099F).

4.11.4.10. Write-only Mode

The write-only mode provides an efficient way to fill a range of memory with a specific constant value (Word). When the write-only mode is enabled, the channel controller will issue only the write command without the read command.

4.11.4.11. Unalign Transfer Mode

Without the unalign transfer mode, the source and destination addresses must be aligned with the source and destination data width, respectively. However, the performance will not be ideal if the source and destination address are not aligned to a wider data width. The worst-case condition is that both addressees are aligned to the byte location and DMA can only issue the AXI read/write command with the "byte" AXI size. The unaligned transfer mode can be useful in the above condition. Once the unalign transfer mode is enabled, the source and destination addresses will have no restriction against the source and destination data width. Therefore, DMA can issue the AXI read/write with the largest possible AXI size to reduce the number of required beats for data transfer.

4.11.4.12. Examples of Channel Control Settings

There are some rules for SrcTcnt ([31:29] of CnCTL), SrcWidth ([27:25] of CnCTL), DstWidth ([24:22] of CnCTL), and CnTSIZE (0x114). First of all, CnTSIZE can be determined by formula:

CnTSIZE = Total_Transfer_Bytes / SrcWidth , which Total_Transfer_Bytes is the total bytes that users want to transfer.

For example, assume that AXI data bus is 64 bits and set SrcWidth = DstWidth = 64 bits (assume source and destination are all 64 bits wide). If users want to transfer the whole 1024 bytes of data, then CnTSIZE = Total_Transfer_Bytes/SrcWidth = 1024 bytes/64 bits = 128. Hence, CnTSIZE should be set to 128 (i.e. ARLenM will transfer the whole 128 beats of data in several AR commands).

Second, if users don't know what value should be set to SrcWidth or DstWidth (assume that the width of source device and destination device are all same as the AXI bus width), the value of AXI data bus width can be set to both SrcWidth and DstWidth for better performance (as the example shown above). The size of ARSizeM is the width set in SrcWidth. Similarly, the size of AWSizeM is the width set in DstWidth.

Vangotech

SrcTcnt is only valid in the handshake mode and is used to determine how many beats should be transferred in one handshake operation (one operation means a request with an acknowledgement). As the example shown above, assume that SrcTcnt is set to 16 beats (i.e. one handshake operation only transfer 16 beats of data), there will be 8 (CnTSIZE/SrcTcnt = 128/16 = 8) handshake operations to transfer the whole 128 beats of data.

For another example, if the source device is 8-bit wide and the destination device is 64-bit wide. Users want to transfer the whole 256 bytes and hope that all transfers can be completed within two handshake operations. Then,

1. SrcWidth = source device width = 8 bits
2. DstWidth = destination device width = 64 bits
3. CnTSIZE = Total_Transfer_Bytes / SrcWidth = 256 bytes/8 bit = 256 beats
4. SrcTcnt = 256 beats/2 = 128 beats (because 256 beats should be transferred by two times.)

4.11.4.13. Usage of Event

By using the set/wait event function, the operation sequence of two different channels can be arranged before the channels are enabled. For example, the source location of the DMA operation executed by the channel 1 is the destination location of the DMA operation executed by the channel 0. For accurate data coherency, the channel 1 should wait for channel 0 to finish its DMA operation and starts.

In this case, channel 0 and channel 1 can use the event function to hand over the job. Users can configure channel 0 to set event 0 and channel 1 to wait for event 0 before enabling these two channels. After enabling these two channels, channel 1 will not start the DMA operation until event 0 is set. Channel 0 will set event 0 when all of the requests of the DMA operations to channel 0 are done. WSYNC of channel 0 should be set to '1' to ensure that all the transfers will be accomplished before the event is set. Channel 1 will clear event 0 when all of the requests of the DMA operations to channel 1 are accomplished.

4.11.5. Register List

Module Name	Base Address
DMA_0	0x1000_0000
DMA_1	0x1010_0000

Register Name	Offset	Description
INT	00	Interrupt
TCINT	04	Terminal count interrupt
TCINTC	08	Terminal count interrupt clear
EAINT	0C	Error/Abort interrupt
EAINTC	10	Error/Abort interrupt clear
TCS	14	Terminal count status
EAS	18	Error/Abort status
CES	1C	Channel enable status
SPI	20	Synchronization of peripheral interface
LDMBA	24	Local descriptor memory base address
WDT	28	Watchdog timer

Vangotech

GER	2C	Global event register
PSE	30	PSlvErr enable
REV	34	Revision number
FEA	38	Hardware feature
LMFFS0	3C	Local descriptor memory free flag set 0
LMFFS1	40	Local descriptor memory free flag set 1
LMFFS2	44	Local descriptor memory free flag set 2
LMFFS3	48	Local descriptor memory free flag set 3
EDNC	4C	Endian conversion
CVWOM	50	Constant value of the write-only mode
FEA2	54	Hardware feature 2
C0CTL	100	Control register
C0CFG	104	Configuration register
C0SRCA	108	Source address
C0DSTA	10C	Destination address
C0LNK	110	Link list pointer
C0TSIZE	114	Transfer size
C0SSDA	118	Stride of the source/destination address
C1CTL	120	Control register
C1CFG	124	Configuration register
C1SRCA	128	Source address
C1DSTA	12C	Destination address
C1LNK	130	Link list pointer
C1TSIZE	134	Transfer size
C1SSDA	138	Stride of the source/destination address
C2CTL	140	Control register
C2CFG	144	Configuration register
C2SRCA	148	Source address
C2DSTA	14C	Destination address
C2LNK	150	Link-list pointer
C2TSIZE	154	Transfer size
C2SSDA	158	Stride of the source/destination address
C3CTL	160	Control register
C3CFG	164	Configuration register
C3SRCA	168	Source address
C3DSTA	16C	Destination address
C3LNK	170	Link list pointer
C3TSIZE	174	Transfer size
C3SSDA	178	Stride of source/destination address
C4CTL	180	Control register
C4CFG	184	Configuration register

Vangotech

C4SRCA	188	Source address
C4DSTA	18C	Destination address
C4LNK	190	Link list pointer
C4TSIZE	194	Transfer size
C4SSDA	198	Stride of the source/destination address
C5CTL	1A0	Control register
C5CFG	1A4	Configuration register
C5SRCA	1A8	Source address
C5DSTA	1AC	Destination address
C5LNK	1B0	Link list pointer
C5TSIZE	1B4	Transfer size
C5SSDA	1B8	Stride of the source/destination address
C6CTL	1C0	Control register
C6CFG	1C4	Configuration register
C6SRCA	1C8	Source address
C6DSTA	1CC	Destination address
C6LNK	1D0	Link list pointer
C6TSIZE	1D4	Transfer size
C6SSDA	1D8	Stride of the source/destination address
C7CTL	1E0	Control register
C7CFG	1E4	Configuration register
C7SRCA	1E8	Source address
C7DSTA	1EC	Destination address
C7LNK	1F0	Link list pointer
C7TSIZE	1F4	Transfer size
C7SSDA	1F8	Stride of the source/destination address
C0SADAEXP	200	Channel 0source address and destination address expand
C0LLPUSEREXP	204	Channel0 link list pointer and user-defined signal expand
C1SADAEXP	220	Channel 1 source address and destination address expand
C1LLPUSEREXP	224	Channel1 link list pointer and user-defined signal expand
C2SADAEXP	240	Channel 2 source address and destination address expand
C2LLPUSEREXP	244	Channel2 link list pointer and user-defined signal expand
C3SADAEXP	260	Channel 3 source address and destination address expand
C3LLPUSEREXP	264	Channel3 link list pointer and user-defined signal expand
C4SADAEXP	280	Channel 4 source address and destination address expand
C4LLPUSEREXP	284	Channel4 link list pointer and user-defined signal expand
C5SADAEXP	2a0	Channel 5 source address and destination address expand
C5LLPUSEREXP	2a4	Channel5 link list pointer and user-defined signal expand
C6SADAEXP	2c0	Channel 6 source address and destination address expand
C6LLPUSEREXP	2c4	Channel6 link list pointer and user-defined signal expand
C7SADAEXP	2e0	Channel 7 source address and destination address expand

vangotech

C7LLPUSEREXP	2e4	Channel7 link list pointer and user-defined signal expand

4.11.6. Register Description

4.11.6.1. Interrupt Register

The interrupt register is the result of the terminal count and the error/abort interrupt. It cannot be directly cleared and will be cleared when the corresponding terminal count or the error/abort interrupt is cleared.

Table 4-280. Interrupt Register (INT, Offset = 0x00)

Bit	Name	Type	Description	Default Value
[31:8]	-	RO	Reserved and read as zero	0
[7:0]	Intr	RO	Interrupt of each DMA channel	0

4.11.6.2. Terminal Count Interrupt Register

When a DMA channel finishes all the requested transfers, the corresponding terminal count interrupt will be asserted if the mask is not set. The mask bit is in the DMA channel configuration register.

Table 4-281. Terminal Count Interrupt Register (TCINT, Offset = 0x04)

Bit	Name	Type	Description	Default Value
[31:8]	-	RO	Reserved and read as zero	0
[7:0]	TCIntr	RO	DMA channel terminal count interrupt	0

4.11.6.3. Terminal Count Interrupt Clear Register

Once the terminal count interrupt register is set, it can be cleared by writing '1' to the corresponding bit in this register. The terminal count status register can also be cleared by writing '1' to the corresponding bit in this register. Please note that when the channel is aborted, the terminal count must be cleared after asserting the abort interrupt.

Table 4-282. Terminal Count Interrupt Clear Register (TCINTC, Offset = 0x08)

Bit	Name	Type	Description	Default Value
[31:8]	-	RO	Reserved and read as zero	0
[7:0]	TCIntrClr	RW1C	DMA channel terminal count interrupt clear	0

4.11.6.4. Error/Abort Interrupt Register

The error interrupt register will be set when the response error (Either the read transaction or the write transaction) is received on the AXI bus. The WDT timeout interrupt register will be set when the watchdog timer reaches the watchdog timer value. These two interrupts can be masked if no interrupt needs to be generated. The mask bit is in the DMA channel configuration register.

The abort interrupt register will be set when a channel is aborted and the corresponding mask is not set. The channel abort is resulted from one of the following conditions:

Write '0' to "Channel enable status" (Offset = 0x1C)

Receive error response on the AXI bus while performing the descriptor read operation

The current handshake data will be finished by the aborted channel in the DMA handshake mode. When in the normal operation mode, the aborted channel will issue the last DMA transfer with the write synchronization and wait for the write response of the last write transfer (This guarantees that all the commands related to the aborted channels in the command queue will be drained out).

Table 4-283. Error/Abort Interrupt Register (EAINT, Offset = 0x0C)

Bit	Name	Type	Description	Default Value
[31:24]	-	RO	Reserved and read as zero	0
[23:16]	AbtIntr	RO	DMA channel abort interrupt	0
[15:8]	WdtIntr	RO	DMA channel WDT timeout interrupt	0
[7:0]	ErrIntr	RO	DMA channel error interrupt	0

4.11.6.5. Error/Abort Interrupt Clear Register

Once this register is set, it can be cleared by writing '1' to the corresponding clear bit in this register.

Table 4-284. Error/Abort Interrupt Clear Register (EAINTC, Offset = 0x10)

Bit	Name	Type	Description	Default Value
[31:24]	-	RO	Reserved and read as zero	0
[23:16]	AbtIntrClr	RW1C	DMA channel abort interrupt clear	0
[15:8]	WdtIntrClr	RW1C	DMA channel WDT timeout interrupt clear	0
[7:0]	ErrIntrClr	RW1C	DMA channel error interrupt clear	0

4.11.6.6. Terminal Count Status Register

When a DMA channel finishes all the requested transfers, the corresponding terminal count status register will be set regardless what the mask value is. The terminal count status will be cleared by writing '1' to the corresponding bit in the terminal count interrupt clear register (Offset = 0x08).

Table 4-285. Terminal Count Status Register (TCS, Offset = 0x14)

Bit	Name	Type	Description	Default Value
[31:8]	-	RO	Reserved and read as zero	0
[7:0]	TCStatus	RO	DMA channel abort interrupt clear	0

4.11.6.7. Error/Abort Status Register

The error status register will be set when the response error (Either the read transaction or the write transaction) is received on the AXI bus. The WDT timeout status register will be set when the watchdog timer reaches the watchdog timer value. The abort status register will be set when a channel is aborted. Both status registers will not be affected by the mask register and will be cleared by writing '1' to the error/abort interrupt clear register (Offset = 0x10).

Table 4-286. Error/Abort Interrupt Register (EAS, Offset = 0x18)

Bit	Name	Type	Description	Default Value
[31:24]	-	RO	Reserved and readas zero	0
[23:16]	AbtStatus	RO	DMA channel abort status	0
[15:8]	WdtStatus	RO	DMA channel WDT timeout status	0
[7:0]	ErrStatus	RO	DMA channel error status	0

4.11.6.8. Channel Enable Register

The channel enable register will be set when the corresponding channel is enabled.

Table 4-287. Channel Enable Register (CES, Offset = 0x1C)

Bit	Name	Type	Description	Default

				Value
[31:8]	-	RO	Reserved and read as zero	0
[7:0]	ChEnable	RW	DMA channel enable status This bit can be read to check the enable status of each channel. Writing '0' to this register will cause the corresponded channel to start an abort operation. Writing '1' to this register has no effect (The channel is enabled through the control register). For example, writing this register with 0xFE will trigger Channel 0 to abort the operation but will not affect other channels. However, the abort will take some time to proceed and the enable bit will only be cleared after the channel finishes an abort operation.	0

4.11.6.9. Synchronization of Peripheral Interface Register

The inbound dma_req signal may come from the peripheral located in different clock domains in the AClk clock domain of FTDMAC030. Therefore, it is required to synchronize the dma_req signal. To add two additional registers to synchronize the corresponding dma_req signal, please set this register to '1'.

Table 4-288. Synchronization of Peripheral Interface Register (SPI, Offset = 0x20)

Bit	Name	Type	Description	Default Value
[31:16]	-	RO	Reserved and read as zero	0
[15:0]	SyncDmaReq	RW	Synchronization of the peripheral interface	0

4.11.6.10. Local Descriptor Memory Base Register

The base address of the local descriptor memory is set in this register. The default size of this region is 1M byte. Please note that this value will be valid only when it is not zero.

Note: Bit[31:20] is for setting the base address in the 32-bit AXI address bus system. When the N of the hardware macro, DMAC_AXI_AW_N, is larger than 32, the expand base address of the AXI address bus can still set in this register by using the field, LDMBExp ([M:0]), where M equals to N-33. For example, if users configure DMAC_AXI_AW_48, (i.e. The AXI system uses the 48-bit address bus), then users should fill bit[31:20] of the address into the field, LDMBase ([31:20] of register LDMBA), and fill bit[47:32] of the address into the field, LDMBExp ([15:0] of register LDMBA (M = N-33 = 48-33 = 15)), when the writing base address.

Table 4-289. Local Descriptor Memory Base Register (LDMBA, Offset = 0x24)

Bit	Name	Type	Description	Default Value
[31:20]	LDMBase	RW	Base register of the local descriptor memory	0
[19:M+1]	-	RO	Reserved and read as zero When the N of DMAC_AXI_AW_N is larger than	0

			32, M equals to N-33. If the N of DMAC_AXI_AW_N equals to 32, the bit of this field is [19:0], and the signal below [M:0] does not exist.	
[M:0]	LDMBExp	RW	Bit expand of base register of the local descriptor memory This field exists only when the N of DMAC_AXI_AW_N is larger than 32, and M equals to N-33.	0

4.11.6.11. Watchdog Timer Register

The watchdog timer provides a channel hang-free mechanism to detect the hanging channel. Every channel contains one counter and the counter will count when the channel is not at the IDLE state. Once the counter reaches the value set in this register and the corresponding watchdog timer enable is set, the status of the channel can be checked by bit[15:8] of register EAS (0x18). Please note that there will be no interrupt for watchdog timer. The expiration means that a channel cannot be granted by the channel arbiter or cannot receive the AXI write response in time.

Table 4-290. Watchdog Timer Register (WDT, Offset = 0x28)

Bit	Name	Type	Description	Default Value
[31:16]	-	RO	Reserved and read as zero	0
[15:0]	WDTimer	RW	Watchdog timer	80

4.11.6.12. Global Event Register

The global events are used to synchronize different channels (Channel 1 can start only when channel 0 finishes). A total of eight events can be set or cleared by accessing the event register. The event can also be set by the channel controllers. Please note that this register should not be written in advance once an event is expected to be set by a channel controller.

Table 4-291. Global Event Register (GER, Offset = 0x2C)

Bit	Name	Type	Description	Default Value
[31:24]	-	RO	Reserved and read as zero	0
[23:16]	GlbEventClr	RW1C	Clear the global event by setting this bit to '0' 1: Clear to '0' 0: No effect	0
[15:8]	GlbEventSet	RW1C	Set the global event by setting this bit to '1' 1: Set to '1' 0: No effect	0
[7:0]	GlbEvent	RO	Global event register	0

4.11.6.13. PSlvErr Enable Register

The APB slave error response can be activated by setting this register to '1' (This is useful for debugging).

Please note that the region of APB register setting is within 1M. On the other hand, this PSlverr Enable Register only checks for non-existing registers within 1M boundary (Bit[19:0] of APB bus address), but it ignores bit[31:20] of the APB bus address.

Table 4-292. PSlvErr Disable Register (PSE, Offset = 0x30)

Bit	Name	Type	Description	Default Value
[31:1]	-	RO	Reserved and read as zero	0
0	PSlvErrEn	RW	APB slave error response of disabled register 1: APB slave responds the error when accessing the non-existing registers. 0: APB slave does not respond the error when accessing the non-existing registers.	0

4.11.6.14. Revision Number Register

Table 4-293. Revision Number Register (REV, Offset = 0x34)

Bit	Name	Type	Description	Default Value
[31:0]	RevNum	RO	Revision number of FTDMAC030	0x0001_0500

4.11.6.15. Hardware Feature Register

Table 4-294. Hardware Feature Register (FEA, Offset = 0x38)

Bit	Name	Type	Description	Default Value
[31:30]	-	RO	Reserved and read as zero	0
[29:28]	CmdDepth	RO	Depth setting of the read/write command queue 0: Depth = 2 1: Depth = 4 2: Depth = 8 3: Depth = 16	2
[27:26]	-	RO	Reserved and read as zero	0
[25:24]	LdmDepth	RO	Depth setting of the local descriptor memory 2: Depth = 128	2
[23:21]	-	RO	Reserved and read as zero	0
20	LdmOn	RO	1: The local descriptor memory is configured. 0: The local descriptor memory is not configured.	1
[19:16]	PriNum	RO	Number of the peripheral interfaces (dma_req/dma_ack pairs) 15: 16 pairs	15
[15:13]	-	RO	Reserved and read as zero	0
12	PriOn	RO	1: The peripheral interface exists. 0: The peripheral interface does not exist.	1
11	-	RO	Reserved and read as zero	0

Vangotech

[10:8]	DFDepth	RO	Depth setting of data FIFO 0: Depth = 8 1: Depth = 16 2: Depth = 32 3: Depth = 64 4: Depth = 128	3
[7:6]	SLVDWidth	RO	AXI data bus width of the AXI slave port 00: 32 bits 01: 64 bits 10: 128 bits	01
[5:4]	DWidth	RO	AXI data bus width of the AXI master port 00: 32 bits 01: 64 bits 10: 128 bits	01
3	UnalignMode	RO	Unalign transfer mode 0: Unalign transfer mode is not configured. 1: Unalign transfer mode is configured.	1
[2:0]	ChNum	RO	Number of the DMA channels 0: One channel … 7: Eight channels	7

Table 4-295. Hardware Feature Register 2 (FEA2, Offset = 0x54)

Bit	Name	Type	Description	Default Value
[31:20]	-	RO	Reserved and read as zero	0
[19:16]	ARUserWidth	RO	Width of the user-defined read address channel (0 ~ 8)	0
[15:12]	-	RO	Reserved and read as zero	0
[11:8]	AWUserWidth	RO	Width of the user-defined write address channel (0 ~ 8)	0
[7:6]	-	RO	Reserved and read as zero	0
[5:0]	AddrWidth	RO	AXI address width (32 ~ 48)	32

4.11.6.16. Local Descriptor Memory Free Flag Register

The local descriptor memory free flag register indicates if it is occupied or free. Once a descriptor is written to the memory, the corresponding free flag bit will be cleared. After the descriptor is read by the DMA channel controller, the free flag bit will be set to '1', which indicates that the memory space is available again. Writing '1' to this register can set the free flag to '1'. If DMAC_LM is not defined, the default value of these registers will equal to '0'.

"n" is the index number from 0 to 3.

Table 4-296. Local Descriptor Memory Free Flag Set 0 Register (LMFFS0, Offset = 0x3C)

Bit	Name	Type	Description	Default Value
[31:0]	FreeFlagS0	RW1C	Local descriptor memory free flag 1: The space is free. 0: The space is occupied.	FFFF_FFFF

Table 4-297. Local Descriptor Memory Free Flag Set 1 Register* (LMFFS1, Offset = 0x40)

Bit	Name	Type	Description	Default Value
[31:0]	FreeFlagS1	RW1C	Local descriptor memory free flag 1: The space is free. 0: The space is occupied.	FFFF_FFFF

*It only exists when the depth of the local descriptor memory is 64 or 128.

Table 4-298. Local Descriptor Memory Free Flag Set 2 Register* (LMFFS2, Offset = 0x44)

Bit	Name	Type	Description	Default Value
[31:0]	FreeFlagS2	RW1C	Local descriptor memory free flag 1: The space is free. 0: The space is occupied.	0xFFFF_FFFF

*It only exists when the depth of the local descriptor memory is 128.

Table 4-299. Local Descriptor Memory Free Flag Set 3 Register* (LMFFS3, Offset = 0x48)

Bit	Name	Type	Description	Default Value
[31:0]	FreeFlagS3	RW1C	Local descriptor memory free flag 1: The space is free. 0: The space is occupied.	0xFFFF_FFFF

*It only exists when the depth of the local descriptor memory is 128.

4.11.6.17. Endian Conversion Register

Table 4-300. Endian Conversion Register (EDNC, Offset = 0x4C)

Bit	Name	Type	Description	Default Value
[31:17]	-	RO	Reserved and read as zero	0
16	LMEndianConvert	RW	Endian conversion of the AXI slave port 1: BE-8 to LE conversion 0: No conversion The local descriptor memory is in the LE format. If the system is in the BE-8 format, this bit must be set to '1'.	0
[15:8]	-	RO	Reserved and read as zero	0

[7:0]	EndianConvert	RW	Endian conversion from channel 7 to channel 0 1: BE-8 to LE or LE to BE-8 conversion 0: No conversion Please note that this field should be set to '0' when DMA is in one of the following modes: Unalign transfer mode Write-only mode DMA will not support the endian conversion when performing double word access.	0

4.11.6.18. Write-only Mode Constant Value Register

Table 4-301. Write-only Mode Constant Value Register (CVWOM, Offset = 0x50)

Bit	Name	Type	Description	Default Value
[31:0]	WOValue	RW	Constant value for the write-only mode When the write-only mode is enabled, the AXI master will write the specified memory range with this constant value.	0

4.11.6.19. DMA Channel N Control Register

The control register is one element of descriptor. It is updated through the DMA channel descriptor read or through the access from the APB slave port.

"n" means channel index.

Table 4-302. DMA Channel N Control Register (CnCTL, Offset = 0x100 + 0x20 * N)

Bit	Name	Type	Description	Default Value
[31:29]	SrcTcnt	RW	Source transfer count of one DMA handshake operation This value is the length for the source read command in one DMA handshake operation (This value is only valid in the DMA handshake mode). 000: 1 beat 001: 2 beats 010: 4 beats 011: 8 beats 100: 16 beats 101: 32 beats 110: 64 beats 111: 128 beats Restriction: If performing "packing", source transfer count must be equal or larger than the value of DstWidth/SrcWidth when operating in the handshake mode.	0
28	TCMsk	RW	Terminal count mask	0

			1: Mask the terminal count status update 0: Do not mask the terminal count status update	
[27:25]	SrcWidth	RW	Source data width 000: 8 bits 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits Notes: 1. When the unalign mode is enabled, the source data width must be set to the value of the data bus width. 2. When the write-only mode is set, the source data width must be set to the value of the destination data width and must not be less than 32 bits. When the AXI data bus width is 128 bits and the data FIFO depth is equal or less than 32, the combination of source data width = 8 bits and the destination data width = 128 bits will not be allowed. Please note that SrcWidth cannot be larger than AXI data bus width.	0
[24:22]	DstWidth	RW	Destination data width 000: Eight bits 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits Notes: 1. When the unalign mode is enabled, the destination data width must be set to the value of the data bus width. 2. When the write-only mode is set, the destination data width must be set to the value of the source data width and no less than 32 bits. When the AXI data bus width is 128 bits and the data FIFO depth is equal or less than 32, the combination of source data width = 8 bits and destination data width = 128 bits will not be allowed. Please note that DstWidth cannot be larger than AXI data bus width.	0
[21:20]	SrcCtrl	RW	Source burst type control 00: Incremental 10: Fixed Please note that the burst type can only be incremental in the	0

Vangotech

			following modes: 2D transfer mode Write only mode Unalign transfer mode	
[19:18]	DstCtrl	RW	Destination burst type control 00: Incremental 10: Fixed Please note that the burst type can only be incremental in the following modes: 2D transfer mode Write only mode Unalign transfer mode	0
17	WDTEn	RW	Watchdog timer enable 1: Enable 0: Disable Once a channel is set to be aborted, the Watchdog Timer enable will be ineffective (No watchdog expiration will occur after the abort is set).	0
16	ChEn	RW	Channel enable 1: Enable 0: No effect This bit will be automatically disabled when all requested transfers are finished.	0
15	ExpEn	RW	Expand control enable 1: Enable (The control register bits, bits[14:0], are effective.) 0: Disable (The control register bits, bits[14:0], are ineffective.)	0
14	2DEn	RW	2D transfer enable 1: Enable 0: Disable	0
13	WEventEn	RW	Wait event enable 1: Enable (The DMA channelwill start once the value of the current global event register equalsto the value of the channel wait event register) 0: Disable	0
12	SEventEn	RW	Set event enable* 1: Enable (The DMA set global event register as the value in the channel set event register) 0: Disable	0
[11:9]	ChSEvent	RW	Channel set event This signalwill set the specified event after all the DMA	0

Vangotech

			descriptors are executed. Please note that the set event can only be set in the final descriptor in one descriptor chain.	
8	WSync	RW	Wait last write done 1: The DMA channel will finish after the last write command is finished on the AXI bus. 0: The DMA channel will finish after the last write command is pushed into the command queue.	0
[7:0]	ChWEvent	RW	Channel Wait Event Wait for the specified event before the channel starts the DMA operation. A channel can be set to wait for the multiple events.	0

Note: If "set event" is enabled and the channel is aborted, this event will not be set (The software needs to prevent other channels from keeping waiting for this event).

4.11.6.20. DMA Channel N Configuration Register

The channel configuration register can only be modified through the APB slave port access and will not be updated during performing the descriptor read.

"n" means channel index.

Table 4-303. DMA Channel N Configuration Register (CnCFG, Offset = 0x0104 + 0x20 * N)

Bit	Name	Type	Description	Default Value
31	UnalignMode	RW	Unalign transfer mode 1: Enable 0: Disable If the unalign transfer mode is enabled, the source/destination data width must be set to the value of the AXI data bus width. The data pack/unpack will not be executed when this bit is '1'. The source and destination address has no alignment restriction; therefore, the source and destination can be aligned to any byte address. The transfer size unit will always be byte when this mode is enabled.	0
30	WOMode	RW	Write only mode 1: Enable 0: Disable If the write only mode is enabled: The source data width must be equal or greater than 32 bits. The destination data width must be equal or greater than 32 bits. The source data width and destination data width must be programmed with the same value.	0

			The write-only mode and unalign transfer mode are exclusive and cannot be enabled at the same time.	
29	-	RO	Reserved and read as zero	0
28	ChPri	RW	Channel arbitration priority The priority is used by the channel arbiter. 1: High priority 0: Low priority	1
[27:20]	ChGntWin	RW	Channel arbitration grant window This signal is used by the channel arbiter.	0
[19:16]	LLPCnt	RO	Link-list pointer increase counter This counter will increase by 1 when a new descriptor is read by the DMA channel controller.	0
[15:14]	-	RO	Reserved and read as zero	0
13	DstHEn	RW	Destination handshake enable 1: Enable (The destination uses the peripheral interface to handshake with DMA.) 0: Disable Please note that this bit should be set to '0' when DMA is in the following modes: Unalign transfer mode Write-only mode	0
[12:9]	DstRS	RW	Destination handshake request source select 0: Use the dma_req[0]/dma_ack[0] pair … 15: Use the dma_req[15]/dma_ack[15] pair Please note that the values of DstRS and SrcRS cannot be the same.	0
8	-	RO	Reserved and read as zero	0
7	SrcHEn	RW	Source handshake enable 1: Enable (The source uses the peripheral interface to handshake with DMA.) 0: Disable Please note that this bit should be set to '0' when DMA is in the following modes: Unalign transfer mode Write-only mode	0
[6:3]	SrcRS	RW	Source handshake request source selection 0: Use the dma_req[0]/dma_ack[0] pair …	0

Vangotech

			15: Use the dma_req[15]/dma_ack[15] pair Please note that the values of DstRS and SrcRS cannot be the same.	
2	AbtIntMsk	RW	Channel abort interrupt mask 1: Enable the mask 0: Disable the mask	0
1	ErrIntMsk	RW	Channel error interrupt mask 1: Enable the mask 0: Disable the mask	0
0	TCIntMsk	RW	Terminal count interrupt mask 1: Enable the mask 0: Disable the mask	0

4.11.6.21. DMA Channel N Source Address Register

"n" means channel index.

Table 4-304. DMA Channel N Source Address Register (CnSRCA, Offset = 0x108 + 0x20 * N)

Bit	Name	Type	Description	Default Value
[31:0]	SrcAddr	RW	Source address register The source address must be aligned with SrcWidth (Source data width) in the channel control register. When the unalign transfer mode is enabled, the source address can be aligned to any byte address without above restriction.	0

4.11.6.22. DMA Channel N Destination Address Register

"n" means channel index.

Table 4-305. DMA Channel N Destination Address Register (CnDSTA, Offset = 0x10C + 0x20 * N)

Bit	Name	Type	Description	Default Value
[31:0]	DstAddr	RW	Destination address register The destination address must be aligned with DstWidth (Destination data width) in the channel control register. When the unalign transfer mode is enabled, the destination address can be aligned to any byte address without above restriction.	0

4.11.6.23. DMA Channel N Link List Pointer Register

"n" means channel index.

Table 4-306. DMA Channel N Link List Pointer Register (CnLNK, Offset = 0x110 + 0x20 * N)

Bit	Name	Type	Description	Default Value

[31:0]	LLP	RW	Link list pointer The channel controller uses the link list pointer as the target address to read a descriptor after the current descriptor finishes.	0

4.11.6.24. DMA Channel N Transfer Size Register

The transfer size is the number of beats that are read from source while processing the current descriptor. If 2DEn is set, the register format will be changed.

Restriction: If the descriptor is set by directly writing to the DMA channel register (i.e. set from APB) and is expected to perform in two-dimension data transfer, the DMA Channel N Control Register (CnCTL) must be set before DMA Channel N Transfer Size Register (CnTSIZE). This is to ensure 2DEn is set first, and this register will get the value for XTCnt and YTCnt but not for TCnt.

"n" means channel index.

Table 4-307. DMA Channel N Transfer Size Register (CnTSIZE, Offset = 0x114 + 0x20 * N, when 2DEn is not set.)

Bit	Name	Type	Description	Default Value
[31:22]	-	RO	Reserved and read as zero	0
[21:0]	TCnt	RW	Transfer size of current descriptor When the unalign transfer mode is enabled, the unit of the transfer size will be byte.	0

Table 4-308. DMA Channel N Transfer Size Register (CnTSIZE, Offset = 0x114 + 0x2 0 *N, when 2DEn is set.)

Bit	Name	Type	Description	Default Value
[31:16]	YTCnt	RW	Y dimension transfer size of current descriptor	0
[15:0]	XTCnt	RW	X dimension transfer size of current descriptor When the unalign transfer mode is enabled, the unit of the transfer size will be byte.	0

4.11.6.25. DMA Channel N Source/Destination Stride Register

The stride value is only used in the 2D enabled mode. When the X dimension transfer size is reached, the source and the destination address will be reloaded with the original address value plus/minus the stride value.

"n" means channel index.

Table 4-309. DMA Channel N Source/Destination Stride Register (CnSSDA, Offset = 0x118 + 0x20 * N)

Bit	Name	Type	Description	Default Value

[31:16]	DstStride	RW	Destination stride It is a signed value (2's complement). 0x0000: 0 … 0x7FFF: 32767 0x8000: -32768 0x8001: -32767 … 0xFFFF: -1 Please note that the destination stride value must be the multiple of the destination data width.	0
[15:0]	SrcStride	RW	Source stride It is a signed value (2's complement). 0x0000: 0 … 0x7FFF: 32767 0x8000: -32768 0x8001: -32767 … 0xFFFF: -1 Please note that the source stride value must be the multiple of the source data width.	0

4.11.6.26. DMA Channel N Source Address and Destination Address Expand Register

"n" means channel index.

This register is for filling the expanded address of the AXI address bus when the N of the hardware macro, DMAC_AXI_AW_N, is larger than 32.

Table 4-310. DMA Channel N Source Address and Destination Address Expand Register (CnSADAEXP, Offset = 0x200 + 0x20 * N)

Bit	Name	Type	Description	Default Value
[31:16]	DstAddrExp	RW	Destination address expansion Additional space for the destination address to store the expand destination address bit [N-1:32], where N is determined by the hardware macro, DMAC_AXI_AW_N.	0
[15:0]	SrcAddrExp	RW	Source address expansion Additional space for the source address to store the expand destination address bit [N-1:32], where N is determined by the hardware macro, DMAC_AXI_AW_N.	0

4.11.6.27. DMA Channel N Link List Pointer and UserUser-defined Signal Expand Register

"n" means channel index.

The bit[15:0] of this register is for filling the expanded link list pointer when the N of the hardware macro, DMAC_AXI_AW_N, is larger than 32. Bit[23:16] is for users to fill the user-defined data of the AXI write address channel when the X of the hardware macro, DMAC_AXI_AWUSER_X, does not equal to zero. Bit[31:24] is for users to fill the user-defined data of the AXI read address channel when the Y of the hardware macro, DMAC_AXI_ARUSER_Y, does not equal to zero.

Table 4-311. DMA Channel N Link List Pointer and User-defined Signal Expand Register (CnLLPUSEREXP, Offset = 0x204 + 0x20 * N)

Bit	Name	Type	Description	Default Value
[31:24]	ARUserDef	RW	User-defined signal for read address channel This field is for storing the value of ARUser of the AXI read address channel when the N of the hardware macro, DMAC_AXI_ARUSER_N, does not equal to zero.	0
[23:16]	AWUserDef	RW	User-defined signal for write address channel This field is for storing the value of AWUser of the AXI write address channel when the N of the hardware macro, DMAC_AXI_AWUSER_N, does not equal to zero.	0
[15:0]	LLPExp	RW	Link list pointer expansion Additional space for the link list pointer to store the expanded link list pointer bits [N-1:32], where N is determined by the hardware macro, DMAC_AXI_AW_N.	0

Vangotech

4.12. DMAC for AHB

4.12.1. Overview

The Direct Memory Access (DMA) controller is designed to enhance the system performance and reduce the interrupt generations for processor. The system efficiency is improved by employing the high-speed data transfers between the system and device. The DMA controller provides eight configurable channels for the memory-to-memory, memory-to-peripheral, peripheral-to-peripheral, and peripheral-to-memory transfers with shared buffer.

4.12.2. Features

- Compliant with AMBA specification (Rev. 2.0)
- Supports only AHB OKAY response
- Supports AHB slave port to configure DMA controller
- Supports two AHB master interfaces for data transfers
- Supports eight configurable DMA channels
- Supports up to 16 DMA requests/acknowledges
- Supports chain transfer
- Supports memory-to-memory, memory-to-peripheral, peripheral-to-peripheral, and peripheral-to-memory transfers with shared buffer
- Supports hardware handshake
- Supports transactions with 8-bit, 16-bit, and 32-bit data width
- Supports only little-endian transfer

4.12.3. Block Diagram

Figure 4-44 illustrates the functional block diagram of the DMA controller configuration.

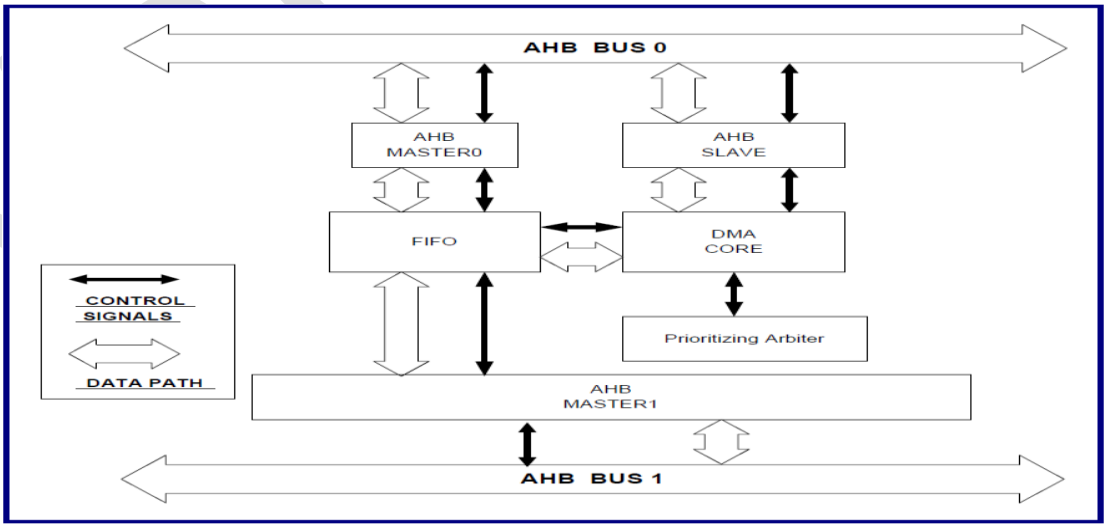


Figure 4-44. DMA Controller Configuration

The DMA controller consists of five main building blocks: Two AHB master interfaces, one AHB slave interface, one FIFO buffer, and one DMA core.

4.12.4. Programming Guidelines

4.12.4.1. General Description

The DMA controller consists of up to eight DMA channels, one DMA engine, and one channel prioritizing arbiter. The DMA channels are based on the round-robin scheme.

4.12.4.2. Chain Transfer

The DMA controller provides the chain transfer function. Multiple blocks can be transferred consecutively without CPU processing after the end of a transfer of each block. Before starting the chain transfer, a link list structure (Shown in Figure 4-45) must build in some system memory area for each channel. Then, link list descriptor pointer register (Cn_LLP) for this channel must be programmed to point to the head of this link list structure.

For channel n, after finishing the first block data transfer, the DMA engine will verify whether or not the link list descriptor pointer address (Bit 31 to bit 2) of link list descriptor pointer register (Cn_LLP) is '0'. If the value is not a '0', it means that the chain transfer function is enabled for this channel. In this case, the DMA controller will fetch link list descriptor of the first channel from the memory attached to AHB Bus 0 or AHB Bus 1 depending on bit 0 of link list descriptor pointer register (Cn_LLP). After finishing the data transfer of link list descriptor of the first channel, the DMA engine will continue the data transfer of the next channel until the end of link list structure. As shown in Figure 4-45, there are three link list descriptors in the link list structure. Therefore, there can be four consecutive blocks to be transferred without CPU processing. The first block is specified in Cn_CSR and Cn_SIZE. The second block is specified in first link list descriptor. The third block is specified in second link list descriptor. The fourth block is specified in the third link list descriptor. Please note that for the first block data transfer, the source, destination, total size and control are based on Cn_SrcAddr, Cn_DstAddr, Cn_Size, and Cn_CSR. For transferring the second and later blocks, the source, destination, total size, and control are based on link list descriptor.

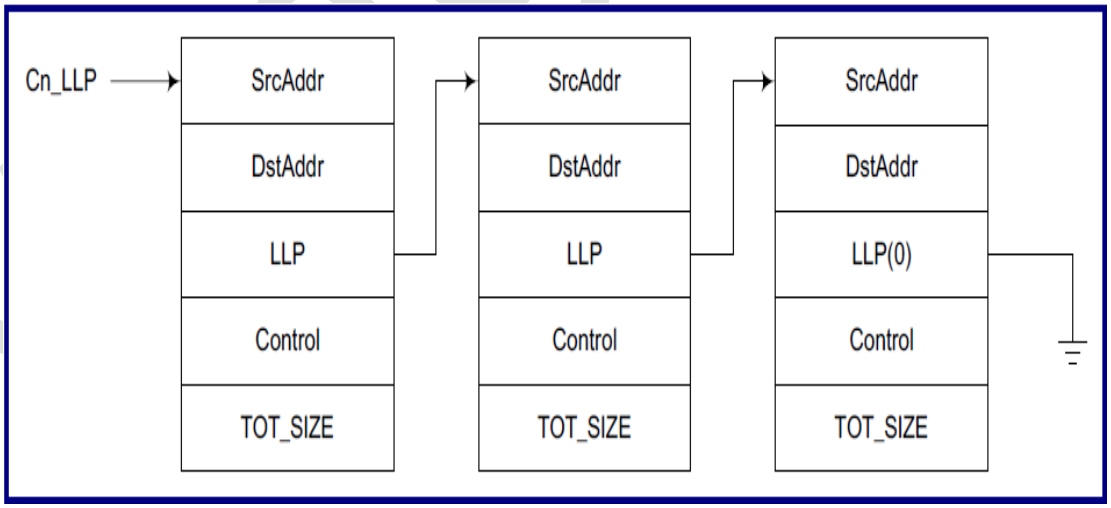


Figure 4-45. Link List Structure of Chain Transfer Operation

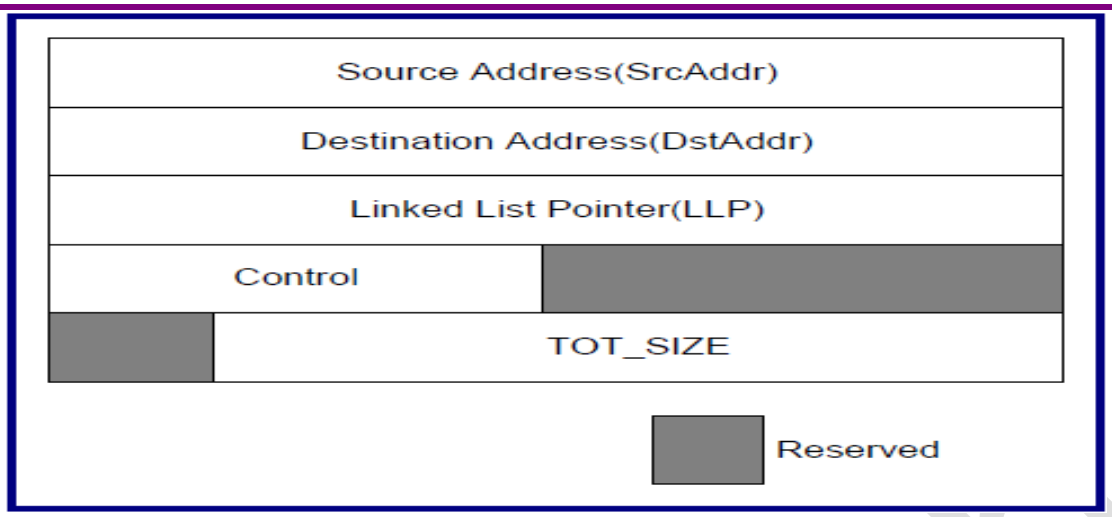


Figure 4-46. Link List Descriptor

Each time the DMA controller fetches link list descriptor (Word-aligned), SrcAddr will be copied to the Cn_SrcAddr register, DstAddr will be copied to the Cn_DstAddr register, LLP will be copied to the Cn_LLP register, total transfer size will be copied to the TOT_SIZE field in the Cn_SIZE register, and Control will be copied to the respective bits in the Cn_CSR register (Please note that total size for LLP table of FTDMAC020 cannot be set to 0).

Table 4-312. Address Map for Link List Descriptor (Base Address: Cn_LLP[31:2])

Name	Offset	Width	Description
SrcAddr	+0	32	Source address
DstAddr	+4	32	Destination address
LLP	+8	32	Link list pointer
Control	+C	32	Control and total transfer size
Total size	+10	32	Total transfer size

Table 4-313. Control Field Definition in Link List Descriptor

Bit	Description
[31:29]	DMA_FF_TH: DMA FIFO threshold value (Same as DMA_FF_TH in Cn_CSR)
28	TC_MSK: Channel terminal count status mask (Same as TC_MSK in Cn_CSR)
[27:25]	SRC_WIDTH: Source transfer width (Same as SRC_WIDTH in Cn_CSR)
[24:22]	DST_WIDTH: Destination transfer width (Same as DST_WIDTH in Cn_CSR)
[21:20]	SRCAD_CTL: Source address control (Same as SRCAD_CTL in Cn_CSR)
[19:18]	DSTAD_CTL: Destination address control (Same as DSTAD_CTL in Cn_CSR)
17	SRC_SEL: Source selection (Same as SRC_SEL in Cn_CSR)
16	DST_SEL: Destination selection (Same as DST_SEL in Cn_CSR)
[15:0]	Reserved

Table 4-314. Total Transfer Size Definition in Link List Descriptor

Bit	Description
[21:0]	TOT_SIZE: Total transfer size (Same as TOT_SIZE in Cn_SIZE)

For each channel, when the chain transfer is enabled, after finishing transfer of one block, the DMA engine will assert the interrupt (dmaint) to inform CPU that the data transfer is complete. Then, CPU should

de-assert the interrupt in time by writing '1' to the respective bit of the Terminal Count Interrupt Status Clear Register before the data transfer of the next block is done. However, in some cases, CPU will not de-assert the interrupt in time before the data transfer of the next block is done. In such cases, CPU does not know whether the data transfer of two blocks is done. This will cause problem to CPU during the chain transfer operation.

For example, Figure 4-47 shows the chain transfer operation composed of transactions 1, 2, and 3, i.e., two link list descriptors are fetched. At time t1, transaction 1 is finished and the DMA controller sets INT_TC to assert dmaint. At time t2, CPU reads INT_TC to know the interrupt source in the interrupt service routine. At time t3, transaction 2 is finished and then the DMA controller sets INT_TC. At time t4, the interrupt service routine for time t2 clears INT_TC. At time t5, transaction 3 is finished, and CPU does not know that transaction 2 had been finished (At t4, the software clears INT_TC of transactions 1 and 2). This causes an interrupt conflict. To avoid this conflict situation, DMA is designed to provide the LLP counter to record the transaction number when the chain transfer is enabled.

The LLP counter (Denoted as LLP_CNT, bit 19 to bit 16 of Channel Configuration Register) operation is shown in Figure 4-48. When ch_en is set to '1' (When the chain transfer is enabled), LLP_CNT is reset to '0'. When each transaction is finished, LLP_CNT is increased by '1'. Please note that when the last transaction is finished, LLP_CNT is also increased by '1'. LLP_CNT will not be increased when LLP_ADDR is '0' and error or abort happens.

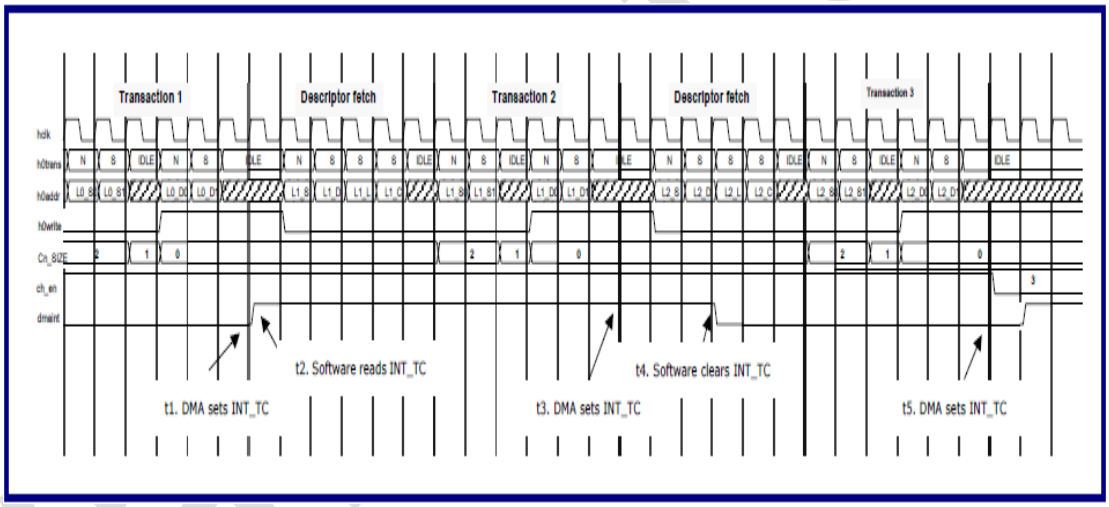


Figure 4-47. Example of Interrupt Conflict during Chain Transfer Operation

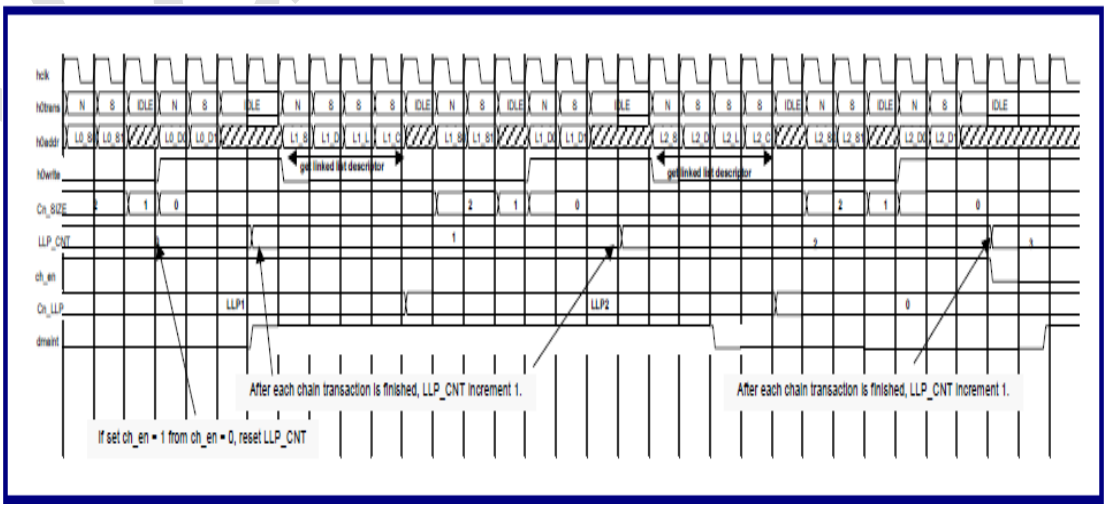


Figure 4-48. LLP_CNT Operation

4.12.4.3. DMA Hardware Handshake Mode

Each channel of the DMA controller must be programmed in the DMA hardware handshake mode or DMA normal mode. The DMA hardware handshake mode will be explained in this section and the DMA normal mode will be explained in the next section.

The DMA hardware handshake mode can be enabled by setting bit 7 of Channel Control register (Cn_CSR). For example, the DMA hardware handshake mode of channel 0 can be enabled by setting bit 7 of C0_CSR. If the DMA hardware handshake mode of channel 0 is enabled, after channel 0 wins the arbitration, the DMA controller will wait for the external DMA request to be asserted before starting the DMA transfer. Each time when the DMA request is asserted, the controller transfers units equal to SRC_BURST_SIZE[1]. When the SRC_BURST_SIZE transfer is completed, the DMA controller will assert acknowledge and then re-arbitrate among all DMA requests. After detecting the assertion of acknowledge, the external device should de-assert the DMA request to let the DMA controller de-assert acknowledge. Please note that DMA_request cannot be asserted again until the de-assertion is acknowledged. After the TOT_SIZE transfers are complete, the DMA controller will assert TC[0] (Bit 0 of the Terminal Count Status register (TC)), dma_tc[0] and both the dmaint_tc and dmaint interrupts (If not masked). Figure 4-49 illustrates an example of the DMA transfer between two AHB interfaces in the hardware handshake mode. Figure 4-50 shows the hardware handshake protocol. Please note that the DMA controller will simultaneously de-assert dma_ack and dma_tc when dma_req is de-asserted.

During the transfer, if the source or destination slave returns an ERROR response, DMA will set the ERR bit (Bits[7:0] of the Error/Abort Status register (ERR_ABT)) and terminate the DMA transfer at once. (Please note that if the source and destination slaves are on different AHB buses and the other destination or source slave returns a RETRY response at the same time, the DMA controller will finish the RETRY cycle before setting the ERR bit). Then, if INT_ERR_MSK (Bit 1 of the Cn_CFG register) is not set, the DMA controller will assert dmaint_err and dmaint. In this case, the DMA controller will not assert acknowledge (dma_ack), and the device must de-assert dma_req by itself. For example, Figure 4-51 shows that an ERROR response happens when the DMA controller reads address S6 of source slave, and that a RETRY cycle happens when the DMA controller writes address D2 of destination slave; therefore, it will force the DMA controller to complete the RETRY cycle before asserting dmaint_err and dmaint interrupts. Please note that, in this case, dma_ack will not be asserted by the DMA controller.

[1]SRC_BURST_SIZE is set according to SRC_SIZE in the Cn_CSRregister.

During a transfer, if the software sets the abort bit (Bit 15 of the Channel Control register (Cn_CSR)), after finishing the SRC_BURST_SIZE transfers or the TOT_SIZE transfers, the DMA controller will set the ABT bit (Bit 23 to bit 16 of the Error/Abort status register (ERR_ABT)) and terminate the DMA transfer at once. (Please note that if the source and destination slaves return a RETRY response at the same time, the DMA controller will finish the RETRY cycle before setting the ABT bit.) Then, if INT_ABT_MSK (Bit 2 of the Cn_CFG register) is '0', the DMA controller will assert dmaint. In such a case, the DMA controller will not assert acknowledge (dma_ack), and the device must de-assert dma_req by itself. For example, Figure 4-52 shows the abort operation when this channel is active. At t1, the internal abort state is set, and after finishing the SRC_BURST_SIZE transfer or the TOT_SIZE transfer (Time t2) and completing the RETRY cycle (Time t3), the DMA controller asserts the dmaint interrupt without asserting dma_ack at time t4. Figure 4-53 shows the abort operation when this channel is inactive. At t1, the internal abort state is set.

Because the channel is inactive, the DMA controller asserts the dmaint interrupt but does not assert the dma_ack at time t2.

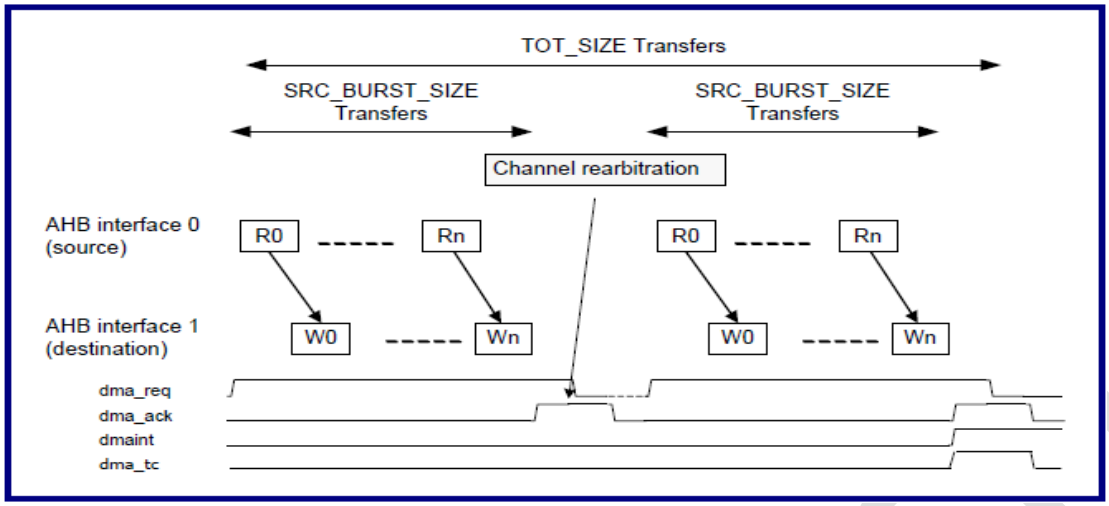


Figure 4-49. Example of Hardware Handshake Mode Transfer (SRC_WIDTH[2] = DST_WIDTH[3])

[2]SRC_WIDTHis the width of a source transfer in the Cn_CSRregister.

[3]DST_WIDTHis the width of a destination transfer in the Cn_CSRregister.

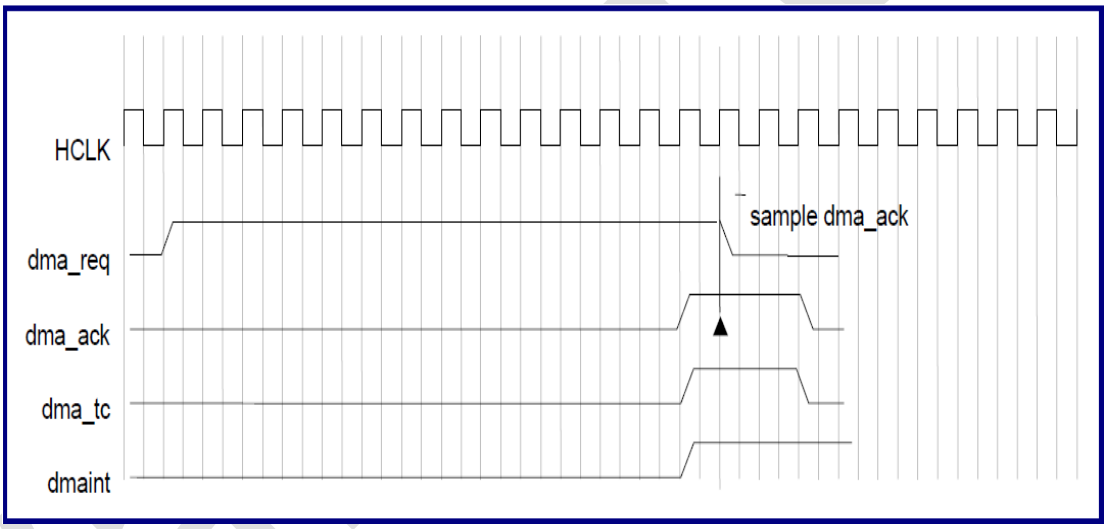


Figure 4-50. DMA Hardware Handshake Protocol

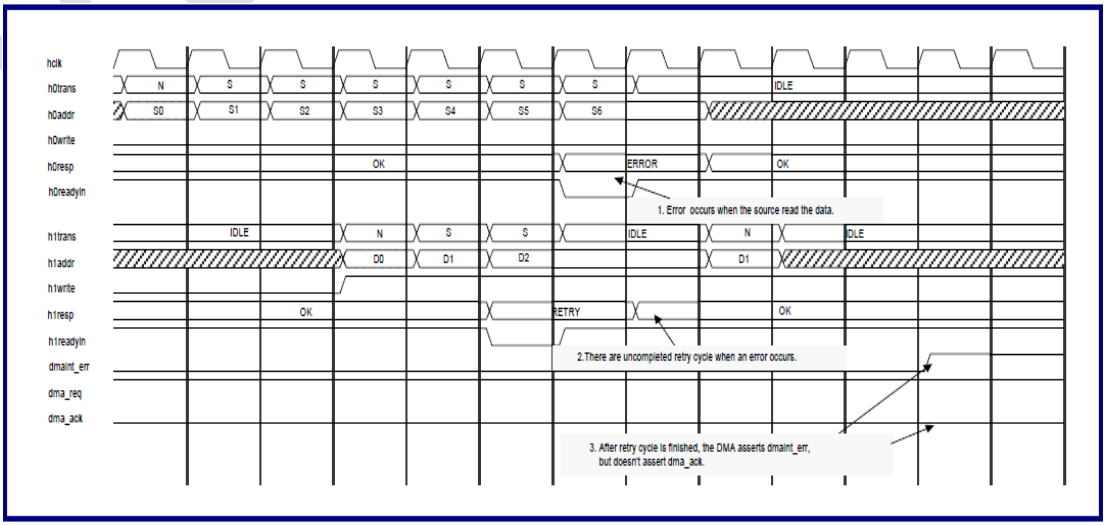


Figure 4-51. ERROR Response of Transfer in Hardware Handshake Mode

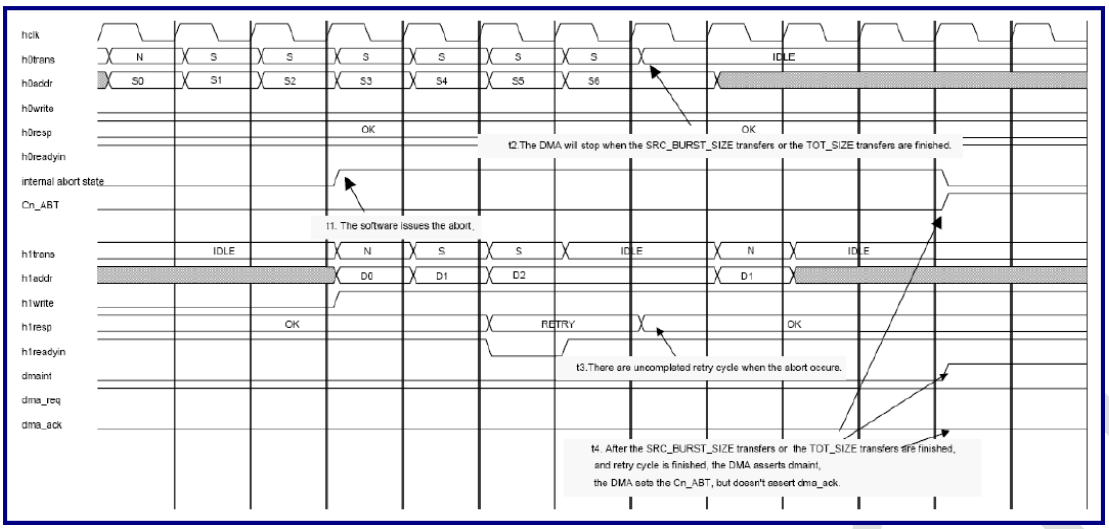


Figure 4-52. Abort Operation with Active Channel

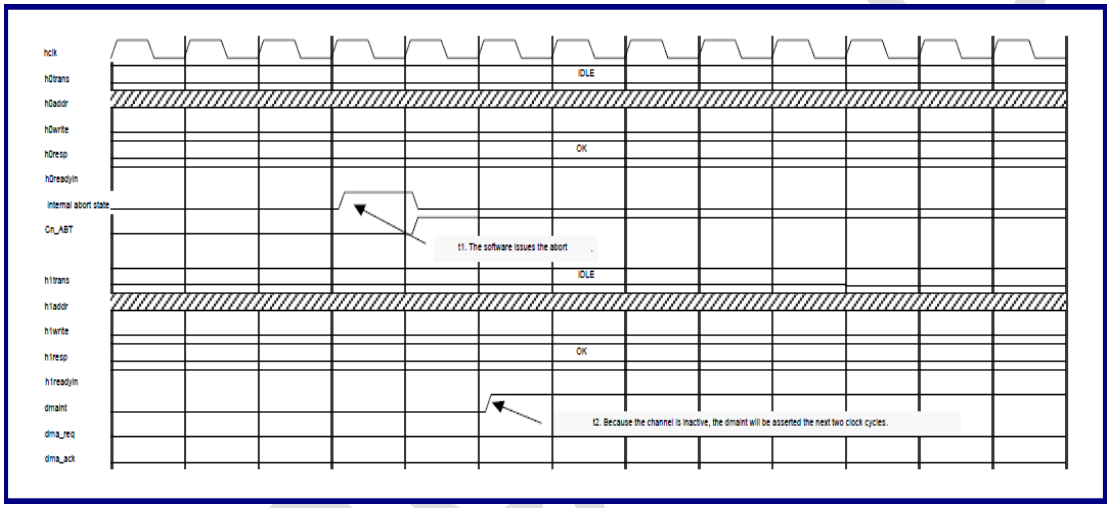


Figure 4-53. Abort Operation with Inactive Channel

4.12.4.4. DMA Normal Mode

In this mode, no external DMA request is needed. The DMA controller will automatically generate a transfer request signal internally. Figure 4-54 illustrates an example of a transfer in this mode.

The abort status will be updated to register if AHB receives ERROR after abort happened. The error and abort operations are the same as those in the DMA hardware handshake mode.

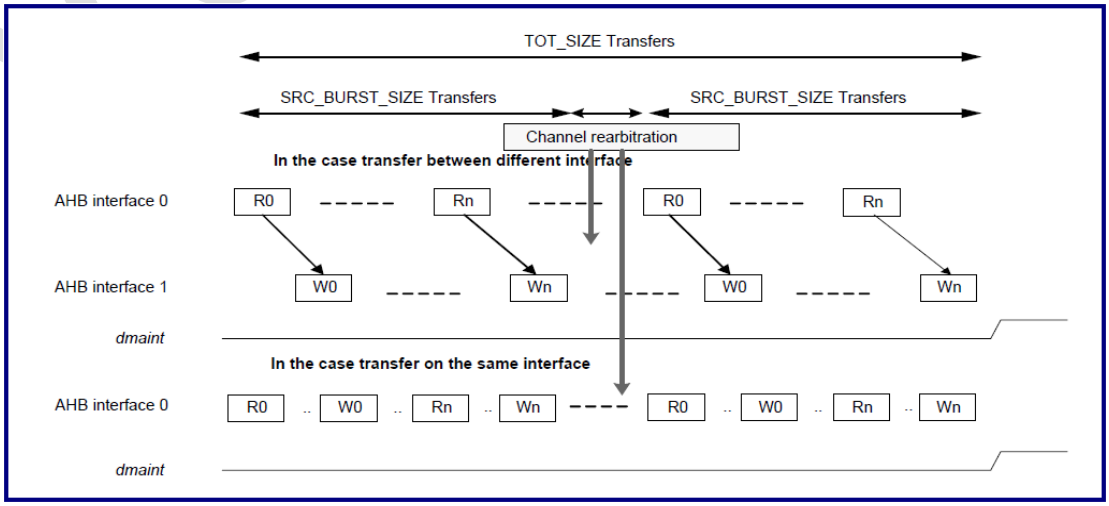


Figure 4-54. Transfer Example in DMA Normal Mode

4.12.5. Register List

Module Name	Base Address
DMA_H	0x5000_0000

Register Name	Offset	Description
INT	000	Interrupt status register
INT_TC	004	Interrupt for terminal count status register
INT_TC_CLR	008	Interrupt for terminal count clear register
INT_ERR/ABT	00C	Interrupt for Error/Abort status register
INT_ERR/ABT_CLR	010	Interrupt for Error/Abort clear register
TC	014	Terminal count status register
ERR/ABT	018	Error/Abort status register
CH_EN	01C	Channel enable status register
CH_BUSY	020	Channel busy register status register
CSR	024	Main configuration status register
SYNC	028	Sync register
DMAC_REVISION	030	FTDMAC020 revision register
DMAC_FEATURE	034	FTDMAC020 feature register
Channel n registers, n = 0 ~ 7		
Cn_CSR	100+ (020 * n)	Channel n control register
Cn_CFG	104+ (020 * n)	Channel n configuration register
Cn_SrcAddr	108+ (020 * n)	Channel n source register
Cn_DstAddr	10C + (020 * n)	Channel n destination register
Cn_LLP	110+ (020 * n)	Channel n link list pointer register
Cn_SIZE	114+ (020 * n)	Channel n transfer size register

4.12.6. Register Description

4.12.6.1. Interrupt Status Register

This register is used to keep the result of the following equation:

INT[n] = INT_ABT[n] | INT_ERR[n] | INT_TC[n]

Where "n" is channel number n, and "|" is logic OR.

INT_ABT[n] means from bit 23 to bit 16 of the error/abort interrupt status register (INT_ERR/ABT).

INT_ERR[n] means from bit 7 to bit 0 of the error/abort interrupt status register (INT_ERR/ABT).

INT_TC[n] means from bit 7 to bit 0 of the terminal count interrupt status register (INT_TC).

Vangotech

dmaint will be asserted by the DMA controller if at least one bit of this register is set; on the other hand, dmaint will be de-asserted by the DMA controller if no bit of this register is set. If the maximum channel number of DMA is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-315. Interrupt Status Register (INT, Offset = 0x00)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	INT[7]	RO	Result of INT_ABT[7]|INT_ERR[7]|INT_TC[7] 0: Channel 7 has no pending interrupt. 1: Channel 7 has a pending interrupt.	0
6	INT[6]	RO	Result of INT_ABT[6]|INT_ERR[6]|INT_TC[6] 0: Channel 6 has no pending interrupt. 1: Channel 6 has a pending interrupt.	0
5	INT[5]	RO	Result of INT_ABT[5]|INT_ERR[5]|INT_TC[5] 0: Channel 5 has no pending interrupt. 1: Channel 5 has a pending interrupt.	0
4	INT[4]	RO	Result of INT_ABT[4]|INT_ERR[4]|INT_TC[4] 0: Channel 4 has no pending interrupt. 1: Channel 4 has a pending interrupt.	0
3	INT[3]	RO	Result of INT_ABT[3]|INT_ERR[3]|INT_TC[3] 0: Channel 3 has no pending interrupt. 1: Channel 3 has a pending interrupt.	0
2	INT[2]	RO	The result of INT_ABT[2]|INT_ERR[2]|INT_TC[2] 0: Channel 2 has no pending interrupt. 1: Channel 2 has a pending interrupt.	0
1	INT[1]	RO	The result of INT_ABT[1]|INT_ERR[1]|INT_TC[1] 0: Channel 1 has no pending interrupt. 1: Channel 1 has a pending interrupt.	0
0	INT[0]	RO	The result of INT_ABT[0]|INT_ERR[0]|INT_TC[0] 0: Channel 0 has no pending interrupt. 1: Channel 0 has a pending interrupt.	0

4.12.6.2. Terminal Count Interrupt Status Register

This register shows the status of the DMA terminal count interrupts after masking. The mask bit of these interrupts is bit 0 (INT_TC_MSK) of the Channel Configuration register (Cn_CFG). If this mask bit is set, the content of this register is always '0' regardless of the presence of a pending DMA terminal count interrupt. dmaint_tc is asserted by the DMA controller if at least one bit of this register is set; on the other hand, dmaint_tc is de-asserted by the DMA controller if no bit of this register is set. Please note that both of TC_MSKand INT_TC_MSK should be concurrently masked/unmasked. If the maximum channel number of the DMA controller is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-316. Terminal Count Interrupt Status Register (INT_TC, Offset = 0x04)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	INT_TC[7]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 7 has no pending interrupt. 1: Channel 7 has a pending interrupt.	0
6	INT_TC[6]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 6 has no pending interrupt. 1: Channel 6 has a pending interrupt.	0
5	INT_TC[5]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 5 has no pending interrupt. 1: Channel 5 has a pending interrupt.	0
4	INT_TC[4]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 4 has no pending interrupt. 1: Channel 4 has a pending interrupt.	0
3	INT_TC[3]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 3 has no pending interrupt. 1: Channel 3 has a pending interrupt.	0
2	INT_TC[2]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 2 has no pending interrupt. 1: Channel 2 has a pending interrupt.	0
1	INT_TC[1]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 1 has no pending interrupt. 1: Channel 1 has a pending interrupt.	0
0	INT_TC[0]	RO	Status of the DMA terminal count interrupts after masking 0: Channel 0 has no pending interrupt. 1: Channel 0 has a pending interrupt.	0

Value after reset: 0000h

4.12.6.3. Terminal Count Interrupt Status Clear Register

Writing '1' to bit n of this register clears both INT_TC[n] and TC[n]. INT_TC[n] means bit 7 to bit 0 of the terminal count interrupt status register (INT_TC) and TC[n] means bit 7 to bit 0 of the terminal count status register (TC). If the maximum channel number of the DMA controller is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-317. Terminal Count Interrupt Status Clear Register (INT_TC_CLR, Offset = 0x08)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	INT_TC_CLR[7]	W1C	Writing '1' clears the INT_TC[7] and TC[7] statuses.	0
6	INT_TC_CLR[6]	W1C	Writing '1' clears the INT_TC[6] and TC[6] statuses.	0
5	INT_TC_CLR[5]	W1C	Writing '1' clears the INT_TC[5] and TC[5] statuses.	0

Vangotech

4	INT_TC_CLR[4]	W1C	Writing '1' clears the INT_TC[4] and TC[4] statuses.	0
3	INT_TC_CLR[3]	W1C	Writing '1' clears the INT_TC[3] and TC[3] statuses.	0
2	INT_TC_CLR[2]	W1C	Writing '1' clears the INT_TC[2] and TC[2] statuses.	0
1	INT_TC_CLR[1]	W1C	Writing '1' clears the INT_TC[1] and TC[1] statuses.	0
0	INT_TC_CLR[0]	W1C	Writing '1' clears the INT_TC[0] and TC[0] statuses.	0

4.12.6.4. Error/Abort InterruptStatus Register

INT_ERR is the status of the DMA error interrupts after masking. The mask bit of these interrupts is bit 1 (INT_ERR_MSK) of the channel configuration register (Cn_CFG). If this mask bit is set, the content of INT_ERR[n] of this register will always be '0' regardless of the presence of a pending DMA error interrupt. If an AHB ERROR response happens during the DMA transfer, the DMA controller will stop the current DMA transfer and will set ERR[n] (Bit 7 to bit 0 of the error/abort status register (ERR/ABT)) to '1'. Then, if INT_ERR_MSK is not set, the DMA controller will set INT_ERR[n] to '1' and assert the dmaint_err and dmaint interrupts. Please note that dmaint_err is asserted by the DMA controller if at least one bit of the INT_ERR[7:0] register is set; on the other hand, dmaint_err is de-asserted by the DMA controller if no bit of the INT_ERR[7:0] register is set.

INT_ABT is the status of the DMA abort interrupts after masking. The mask bit of these interrupts is bit 2 (INT_ABT_MSK) of the channel configuration register (Cn_CFG). If this mask bit is set, the content of INT_ABT[n] of this register will always be '0', regardless of the presence of a pending DMA abort interrupt. If the ABT bit (Bit 15 of the channel control register (Cn_CSR)) is set, the DMA controller will stop the current DMA transfer and set ABT[n] (Bit 23 to bit 16 of the error/abort status register (ERR/ABT)) to '1'. Then, if INT_ABT_MSK is not set, the DMA controller will set INT_ABT[n] to '1' and assert the dmaint interrupt. If the maximum channel number of the DMA controller is configured as n, bit n to bit 15 and bit n + 16 to bit 31 of this register will be reserved.

Table 4-318. Error/Abort Interrupt Status Register (INT_ERR/ABT, Offset = 0x0C)

Bit	Name	Type	Description	Default Value
[31:24]	-	RO	Reserved	0
23	INT_ABT[7]	RO	Status of the DMA abort interrupts after masking 0: Channel 7 has no pending interrupt. 1: Channel 7 has a pending interrupt.	0
22	INT_ABT[6]	RO	Status of the DMA abort interrupts after masking 0: Channel 6 has no pending interrupt. 1: Channel 6 has a pending interrupt.	0
21	INT_ABT[5]	RO	Status of the DMA abort interrupts after masking 0: Channel 5 has no pending interrupt. 1: Channel 5 has a pending interrupt.	0
20	INT_ABT[4]	RO	Status of the DMA abort interrupts after masking 0: Channel 4 has no pending interrupt. 1: Channel 4 has a pending interrupt.	0
19	INT_ABT[3]	RO	Status of the DMA abort interrupts after masking	0

Vangotech

			0: Channel 3 has no pending interrupt. 1: Channel 3 has a pending interrupt.	
18	INT_ABT[2]	RO	Status of the DMA abort interrupts after masking 0: Channel 2 has no pending interrupt. 1: Channel 2 has a pending interrupt.	0
17	INT_ABT[1]	RO	Status of the DMA abort interrupts after masking 0: Channel 1 has no pending interrupt. 1: Channel 1 has a pending interrupt.	0
16	INT_ABT[0]	RO	Status of the DMA abort interrupts after masking 0: Channel 0 has no pending interrupt. 1: Channel 0 has a pending interrupt.	0
[15:8]	-	Rs vd	Reserved	0
7	INT_ERR[7]	RO	Status of the DMA error interrupts after masking 0: Channel 7 has no pending interrupt. 1: Channel 7 has a pending interrupt.	0
6	INT_ERR[6]	RO	Status of the DMA error interrupts after masking 0: Channel 6 has no pending interrupt. 1: Channel 6 has a pending interrupt.	0
5	INT_ERR[5]	RO	Status of the DMA error interrupts after masking 0: Channel 5 has no pending interrupt. 1: Channel 5 has a pending interrupt.	0
4	INT_ERR[4]	RO	Status of the DMA error interrupts after masking 0: Channel 4 has no pending interrupt. 1: Channel 4 has a pending interrupt.	0
3	INT_ERR[3]	RO	Status of the DMA error interrupts after masking 0: Channel 3 has no pending interrupt. 1: Channel 3 has a pending interrupt.	0
2	INT_ERR[2]	RO	Status of the DMA error interrupts after masking 0: Channel 2 has no pending interrupt. 1: Channel 2 has a pending interrupt.	0
1	INT_ERR[1]	RO	Status of the DMA error interrupts after masking 0: Channel 1 has no pending interrupt. 1: Channel 1 has a pending interrupt.	0
0	INT_ERR[0]	RO	Status of the DMA error interrupts after masking 0: Channel 0 has no pending interrupt. 1: Channel 0 has a pending interrupt.	0

Value after reset: 0000h

4.12.6.5. Error/Abort Interrupt Status Clear Register

Writing '1' from bit 7 to bit 0 of this register clears both INT_ERR[n] and ERR[n]. Writing '1' from bit 23 to

Vangotech

bit 16 of this register clears both INT_ABT[n] and ABT[n]. If the maximum channel number of DMA is configured as n, bit n to bit 15 and bit n + 16 to bit 31 of this register will be reserved.

Table 4-319. Error/Abort Interrupt Status Clear Register (INT_ERR/ABT_CLR, Offset = 0x10)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
23	INT_ABT_CLR[7]	W1C	Writing '1' clears the INT_ABT[7] and ABT[7] statuses.	0
22	INT_ABT_CLR[6]	W1C	Writing '1' clears the INT_ABT[6] and ABT[6] statuses.	0
21	INT_ABT_CLR[5]	W1C	Writing '1' clears the INT_ABT[5] and ABT[5] statuses.	0
20	INT_ABT_CLR[4]	W1C	Writing '1' clears the INT_ABT[4] and ABT[4] statuses.	0
19	INT_ABT_CLR[3]	W1C	Writing '1' clears the INT_ABT[3] and ABT[3] statuses.	0
18	INT_ABT_CLR[2]	W1C	Writing '1' clears the INT_ABT[2] and ABT[2] statuses.	0
17	INT_ABT_CLR[1]	W1C	Writing '1' clears the INT_ABT[1] and ABT[1] statuses.	0
16	INT_ABT_CLR[0]	W1C	Writing '1' clears the INT_ABT[0] and ABT[0] statuses.	0
[15:8]	-	Rsvd	Reserved	0
7	INT_ERR_CLR[7]	W1C	Writing '1' clears the INT_ERR[7] and ERR[7] statuses.	0
6	INT_ERR_CLR[6]	W1C	Writing '1' clears the INT_ERR[6] and ERR[6] statuses.	0
5	INT_ERR_CLR[5]	W1C	Writing '1' clears the INT_ERR[5] and ERR[5] statuses.	0
4	INT_ERR_CLR[4]	W1C	Writing '1' clears the INT_ERR[4] and ERR[4] statuses.	0
3	INT_ERR_CLR[3]	W1C	Writing '1' clears the INT_ERR[3] and ERR[3] statuses.	0
2	INT_ERR_CLR[2]	W1C	Writing '1' clears the INT_ERR[2] and ERR[2] statuses.	0
1	INT_ERR_CLR[1]	W1C	Writing '1' clears the INT_ERR[1] and ERR[1] statuses.	0
0	INT_ERR_CLR[0]	W1C	Writing '1' clears the INT_ERR[0] and ERR[0] statuses.	0

Vangotech

4.12.6.6. Terminal Count Status Register

This register shows the status of the DMA terminal count after masking. The mask bit for the DMA terminal count is bit 31 (TC_MSK) of the channel control register (Cn_CSR). If this mask bit of the Cn_CSR register is set, TC[n] of this register will always be '0' regardless of the presence of a DMA terminal count. If the maximum channel number of DMA is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-320. Terminal Count Status Register (TC, Offset = 0x14)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	TC[7]	RO	Status of the DMA terminal count 0: Channel 7 has no terminal count status. 1: Channel 7 has a terminal count status.	0
6	TC[6]	RO	Status of the DMA terminal count 0: Channel 6 has no terminal count status. 1: Channel 6 has a terminal count status.	0
5	TC[5]	RO	Status of the DMA terminal count 0: Channel 5 has no terminal count status. 1: Channel 5 has a terminal count status.	0
4	TC[4]	RO	Status of the DMA terminal count 0: Channel 4 has no terminal count status. 1: Channel 4 has a terminal count status.	0
3	TC[3]	RO	Status of the DMA terminal count 0: Channel 3 has no terminal count status. 1: Channel 3 has a terminal count status.	0
2	TC[2]	RO	Status of the DMA terminal count 0: Channel 2 has no terminal count status. 1: Channel 2 has a terminal count status.	0
1	TC[1]	RO	Status of the DMA terminal count 0: Channel 1 has no terminal count status. 1: Channel 1 has a terminal count status.	0
0	TC[0]	RO	Status of the DMA terminal count 0: Channel 0 has no terminal count status. 1: Channel 0 has a terminal count status.	0

4.12.6.7. Error/Abort Status Register

ERR is the status of the DMA error. If an AHB ERROR response happens during a DMA transfer, the DMA controller will stop the current DMA transfer and set ERR[n] to '1'. If INT_ERR_MSK is not set, the DMA controller will set INT_ERR[n] to '1' and assert both the dmaint_err and dmaint interrupts.

ABT is the status of the DMA abort. If the ABT bit (Bit 15 of the Channel Control register (Cn_CSR)) is set, the DMA controller will stop the current DMA transfer and set ABT[n] to '1'. If INT_ABT_MSK is not set, the DMA controller will set INT_ABT[n] to '1' and assert the dmaint interrupt. If the maximum channel number

Vangotech

of DMA is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-321. Error/Abort Status Register (ERR/ABT, Offset = 0x18)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
23	ABT[7]	RO	DMA abort status 0: Channel 7 has no abort status. 1: Channel 7 has an abort status.	0
22	ABT[6]	RO	DMA abort status 0: Channel 6 has no abort status. 1: Channel 6 has an abort status.	0
21	ABT[5]	RO	DMA abort status 0: Channel 5 has no abort status. 1: Channel 5 has an abort status.	0
20	ABT[4]	RO	DMA abort status 0: Channel 4 has no abort status. 1: Channel 4 has an abort status.	0
19	ABT[3]	RO	DMA abort status 0: Channel 3 has no abort status. 1: Channel 3 has an abort status.	0
18	ABT[2]	RO	DMA abort status 0: Channel 2 has no abort status. 1: Channel 2 has an abort status.	0
17	ABT[1]	RO	DMA abort status 0: Channel 1 has no abort status. 1: Channel 1 has an abort status.	0
16	ABT[0]	RO	DMA abort status 0: Channel 0 has no abort status. 1: Channel 0 has an abort status.	0
[15:8]	-	Rsvd	Reserved	0
7	ERR[7]	RO	DMA error status 0: Channel 7 has no error status. 1: Channel 7 has an error status.	0
6	ERR[6]	RO	DMA error status 0: Channel 6 has no error status. 1: Channel 6 has an error status.	0
5	ERR[5]	RO	DMA error status 0: Channel 5 has no error status. 1: Channel 5 has an error status.	0
4	ERR[4]	RO	DMA error status 0: Channel 4 has no error status.	0

Vangotech

			1: Channel 4 has an error status.	
3	ERR[3]	RO	DMA error status 0: Channel 3 has no error status. 1: Channel 3 has an error status.	0
2	ERR[2]	RO	DMA error status 0: Channel 2 has no error status. 1: Channel 2 has an error status.	0
1	ERR[1]	RO	DMA error status 0: Channel 1 has no error status. 1: Channel 1 has an error status.	0
0	ERR[0]	RO	DMA error status 0: Channel 0 has no error status. 1: Channel 0 has an error status.	0

4.12.6.8. Channel Enable Status Register

This register shows the DMA channel enable status. It is a read-only register. If the maximum channel number of the DMA controller is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-322. Channel Enable Status Register (CH_EN, Offset = 0x1C)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	CH_EN[7]	RO	Setting status of channel 7 using the CH_EN bit of the C7_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
6	CH_EN[6]	RO	Setting status of channel 6 using the CH_EN bit of the C6_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
5	CH_EN[5]	RO	Setting status of channel 5 using the CH_EN bit of the C5_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
4	CH_EN[4]	RO	Setting status of channel 4 using the CH_EN bit of the C4_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
3	CH_EN[3]	RO	Setting status of channel 3 using the CH_EN bit of the C3_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0

Vangotech

2	CH_EN[2]	RO	Setting status of channel 2 using the CH_EN bit of the C2_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
1	CH_EN[1]	RO	Setting status of channel 1 using the CH_EN bit of the C1_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0
0	CH_EN[0]	RO	Setting status of channel 0 using the CH_EN bit of the C0_CSR register 0: CH_EN = '0' 1: CH_EN = '1'	0

4.12.6.9. Channel Busy Status Register

This register shows the DMA channel busy status. It is a read-only register. If the maximum channel number of the DMA controller is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-323. Channel Busy Status Register (CH_BUSY, Offset = 0x20)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	CH_BUSY[7]	RO	Setting status of channel 7 using the BUSY bit of the C7_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
6	CH_BUSY[6]	RO	Setting status of channel 6 using the BUSY bit of the C6_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
5	CH_BUSY[5]	RO	Setting status of channel 5 using the BUSY bit of the C5_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
4	CH_BUSY[4]	RO	Setting status of channel 4 using the BUSY bit of the C4_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
3	CH_BUSY[3]	RO	Setting status of channel 3 using the BUSY bit of the C3_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
2	CH_BUSY[2]	RO	Setting status of channel 2 using the BUSY bit of the C2_CFG register	0

Vangotech

			0: BUSY = '0' 1: BUSY = '1'	
1	CH_BUSY[1]	RO	Setting status of channel 1 using the BUSY bit of the C1_CFG register 0: BUSY = '0' 1: BUSY = '1'	0
0	CH_BUSY[0]	RO	Setting status of channel 0 using the BUSY bit of the C0_CFG register 0: BUSY = '0' 1: BUSY = '1'	0

Value after reset: 0000h

4.12.6.10. Main Configuration Status Register

Table 4-324. Main Configuration Status Register (CSR, Offset = 0x24)

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
2	M1ENDIAN	RW	Endian configuration of AHB MASTER1 0 = Little endian 1 = Reserved Reset value = '0' The bit must be tied to '0'.	0
1	M0ENDIAN	RW	Endian configuration of AHB MASTER0 0 = Little endian 1 = Reserved Reset value = '0' The bit must be tied to '0'.	0
0	DMACEN	RW	DMA controller enable 0 = Disable 1 = Enable Reset value = '0'	0

4.12.6.11. Synchronization Register

The synchronization register is used to control the internal synchronization logic of the DMA controller for the hardware DMA request (dma_req[n:0]). If the internal synchronization logic is enabled, it will synchronize the hardware DMA request through the synchronization of two clocks. If the internal synchronization logic is disabled, it will synchronize the hardware DMA request bypass synchronization circuit. It is necessary to enable the internal synchronization logic if the DMA controller and the external device that driving dma_req will run at different clock domains. If the maximum channel number of DMA is configured as n, bit n to bit 31 of this register will be reserved.

Table 4-325. Synchronization Register (SYNC, Offset = 0x28)

Vangotech

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	SYNC[7]	RW	DMA synchronization logic control for requesting channel 7 0: Disable 1: Enable	0
6	SYNC[6]	RW	DMA synchronization logic control for requesting channel 6 0: Disable 1: Enable	0
5	SYNC[5]	RW	DMA synchronization logic control for requesting channel 5 0: Disable 1: Enable	0
4	SYNC[4]	RW	DMA synchronization logic control for requesting channel 4 0: Disable 1: Enable	0
3	SYNC[3]	RW	DMA synchronization logic control for requesting channel 3 0: Disable 1: Enable	0
2	SYNC[2]	RW	DMA synchronization logic control for requesting channel 2 0: Disable 1: Enable	0
1	SYNC[1]	RW	DMA synchronization logic control for requesting channel 1 0: Disable 1: Enable	0
0	SYNC[0]	RW	DMA synchronization logic control for requesting channel 0 0: Disable 1: Enable	0

Value after reset: 0000h

4.12.6.12. DMAC Revision Register

This register shows the current revision of the DMA controller. Table 4-326 lists the format of the revision register.

Table 4-326. FTDMAC020 Revision Register (DMAC_REVISION, Offset = 0x30)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:16]	MAJOR_REV	RO	Major revision number	0x01
[15:8]	MINOR_REV	RO	Minor revision number	0x13
[7:0]	REL_REV	RO	Release number	0x00

4.12.6.13. FTDMAC020 Feature Register

Since the DMA controller is highly configurable, the individual function of the DMA controller can be enabled or disabled before synthesizing a real design. For more details on the configuration definition,

Vangotech

Table 4-327 shows the format of the feature register.

Table 4-327. FTDMAC020 Feature Register (DMAC_FEATURE, Offset = 0x34)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
[15:12]	DMAC_MAX_CHNO_N	RO	DMA maximum channel number, N is ranging from 1 to 8. (Default)	8
11	-	Rsvd	Reserved	0
10	-	Rsvd	Reserved	0
9	DMAC_HAVE_AHB1_YES	RO	1: DMA has AHB 0 and AHB 1. 0: DMA only has AHB 0.	0
8	DMAC_HAVE_LINKLIST_YES	RO	1: DMA supports link list. 0: DMA does not support link list.	1
[7:4]	-	Rsvd	Reserved	0
[3:0]	DMAC_FF_ADD_WIDTH	RO	FIFO RAM address width	5

4.12.6.14. Channel Control Register

Table 4-328. Channel Control Register (Cn_CSR, Offset = 0x100 + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
31	TC_MSK	RW	Terminal count status mask for the current transaction 0: When terminal count happens, the TC status register will be set (Default). 1: When terminal count happens, the TC status register will not be set.	0
[30:27]	-	Rsvd	Reserved	0
[26:24]	DMA_FF_TH	RW	DMA FIFO threshold value: 000: Threshold value = 1 001: Threshold value = 2 010: Threshold value = 4 011: Threshold value = 8 100: Threshold value = 16 101 ~ 111: Reserved When DMA FIFO space ≥ DMA_FF_TH, the DMA controller will start to transfer data from source to FIFO. When the number of valid data in DMA FIFO is greater than DMA_FF_TH, the DMA controller will start to pop out data from FIFO to the destination. Please note that DMA_FF_TH cannot be larger than the 1/2 DMA FIFO size.	0
[23:22]	-	Rsvd	Reserved	0

Vangotech

21	PROT3	RW	Protection information for cacheability [4] 0: Not cacheable (Default) 1: Cacheable	0
20	PROT2	RW	Protection information for bufferablity [5] 0: Not bufferable (Default) 1: Bufferable	0
19	PROT1	RW	Protection information for mode indication [6] 0: User mode (Default) 1: Privileged mode	0
[18:16]	SRC_SIZE	RW	Source burst size selection 000: Burst size = 1 (Default) 001: Burst size = 4 010: Burst size = 8 011: Burst size = 16 100: Burst size = 32 101: Burst size = 64 110: Burst size = 128 111: Burst size = 256 Notes: 1. The source burst size is not related to HBURST (The AHB signals). It only indicates the number of existing transfers before DMA re-arbitrates among the enabled channels. The number of bytes to be transferred for one burst depends on the source burst size and the source transfer width. For example, if the source burst size is 64 (Bit 18 to bit 16 are set as '101') and the source transfer width is 16 bits (Bit 13 to bit 11 are set as '001'), the total number of bytes of this burst transfer will be 128 (64 * 2). 2. (Burst size * SRC_WIDTH) must be equal to or larger than DST_WIDTH. Therefore, the following settings are not allowed: (For DMAC32 or DMAC64) Burst size = 1, source width = 8, destination width = 16 Burst size = 1, source width = 8, destination width = 32 Burst size = 1, source width = 16, destination width = 32 (For DMAC64) Burst size = 1, source width = 8, destination width = 64 Burst size = 1, source width = 16, destination width = 64 Burst size = 1, source width = 32, destination width = 64 Burst size = 4, source width = 8, destination width = 64	0

Vangotech

15	ABT	W1C	Transaction abort Writing '1' to this bit causes DMA to stop the current transfer, then set the ABT[n] bit of the Error/Abort Status register and assert the dmaint interrupt if INT_ABT_MSK = '0'. Note: When writing '1' to bit 15 (ABT), all other bits of this register will remain the same. That is, users cannot program bit 15 to bit 1 and other bits of this register simultaneously.	0
14	-	RsvdP	Reserved and Preserved	0
[13:11]	SRC_WIDTH	RW	DMAC32 Source transfer width Hardware will automatically pack and unpack data as required. 000: Transfer width is 8 bits. 001: Transfer width is 16 bits. 010: Transfer width is 32 bits (Default). Others: Reserved Notes: 1. If source transfer width < destination transfer width, DMA will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack four sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set SRCAD_CTL = '01' (Decrement source address) when the pack function works; otherwise DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack source of 32-bit data and transfer four sets of 8-bit data to destination. DMAC64 Source transfer width Hardware will automatically pack and unpack data as required. 000: Transfer width is 8 bits. 001: Transfer width is 16 bits. 010: Transfer width is 32 bits. 011: Transfer width is 64 bits (Default). Others: Reserved Notes: 1. If source transfer width < destination transfer width, DMA	2

Vangotech

			will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack four sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set SRCAD_CTL = '01' (Decrement source address) when the pack function works; otherwise, DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack the source of 32-bit data and transfer four sets of 8-bit data to destination.	
[10:8]	DST_WIDTH	RW	DMAC32 Destination transfer width Hardware will automatically pack and unpack data as required. 000: Transfer width is 8 bits. 001: Transfer width is 16 bits. 010: Transfer width is 32 bits (Default). Others: Reserved Notes: 1. If source transfer width < destination transfer width, DMA will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack four sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set DSTAD_CTL to '01' (Decrement source address) when the pack function works; otherwise DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack the source of 32-bit data and transfer four sets of 8-bit data to destination. DMAC64 Destination transfer width Hardware will automatically pack and unpack data as required. 000: Transfer width is 8 bits. 001: Transfer width is 16 bits.	2

Vangotech

			010: Transfer width is 32 bits. 011: Transfer width is 64 bits (Default). Others: Reserved Notes: 1. If source transfer width < destination transfer width, DMA will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack for sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set DSTAD_CTL to '01' (Decrement source address) when the pack function works; otherwise DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack the source of 32-bit data and transfer four sets of 8-bit data to destination.	
7	MODE	RW	0: Normal mode (Default) 1: Hardware handshake mode	0
[6:5]	SRCAD_CTL	RW	Source address control 00: Increment source address (Default) 01: Decrement source address 10: Fixed source address 11: Reserved Notes: 1. If source transfer width < destination transfer width, DMA will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack four sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set SRCAD_CTL = 01 (Decrement source address) when the pack function works; otherwise DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack the source of 32-bit data and transfer four sets of 8-bit data to destination.	0

Vangotech

[4:3]	DSTAD_CTL	RW	Destination address control 00: Increment destination address (Default) 01: Decrement destination address 10: Fixed destination address 11: Reserved Notes: 1. If source transfer width < destination transfer width, DMA will pack the source input data. For example, if the source transfer width equals to 8 bits and the destination transfer width equals to 32 bits, then DMA will pack four sets of 8-bit source data and transfer one set of 32-bit data to destination. Limitation: Do not set DSTAD_CTL = 01 (Decrement source address) when the pack function works; otherwise DMA will have a wrong action. 2. If source transfer width > destination transfer width, DMA will unpack the source input data. For example, if the source transfer width equals to 32 bits and the destination transfer width equals to 8 bits, then DMA will unpack the source of 32-bit data and transfer four sets of 8-bit data to destination.	0
2	SRC_SEL	RW	0: AHB Master 0 is the source (Default). 1: AHB Master 1 is the source (Do not set to 1 if DMAC_HAVE_AHB1_NO is defined).	0
1	DST_SEL	RW	0: AHB Master 0 is the destination (Default). 1: AHB Master 1 is the destination (Do not set to 1 if DMAC_HAVE_AHB1_NO is defined).	0
0	CH_EN	RW	Channel enable 0: Disable (Default) 1: Enable	0

[4]This bit controls AHB HPROT[3].

[5]This bit controls AHB H PROT[2].

[6]This bit controls AHB HPROT[1].

4.12.6.15. Channel Configuration Register

Table 4-329. Channel Configuration Register (Cn_CFG, Offset = 0x104 + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
[31:20]	-	Rsvd	Reserved	0
[19:16]	LLP_CNT	RO	Chain transfer counter This counter is reset to '0' when CH_EN changes from '0' to '1'.	0

			If the chain transfer is enabled after each block data transfer is finished, LLP_CNT will be increased by 1.	
[15:14]	-	Rsvd	Reserved	0
13	DST_HE	RW	Destination hardware handshake mode enable 0: Disable 1: Enable When users disable the destination hardware handshake, DMA will start transferring data without waiting for the destination request. This bit will be only valid when FTDMAC020 is in the hardware handshake mode	1
[12:9]	DST_RS	RW	Destination DMA request select It specifies dma_req as the destination, req, and can only be used when the DMA hardware handshake mode is enabled.	0
8	BUSY	RO	The DMA channel is busy. Please note that only one channel can be busy at a time.	0
7	SRC_HE	RW	Source Hardware Handshake Mode enable 0: Disable 1: Enable When users disable the source hardware handshake, DMA will start transferring data without waiting for the source request. This bit will be only valid when DMAC is in the hardware handshake mode.	1
[6:3]	SRC_RS	RW	Source DMA request select It specifies h dma_req as the source, req, and can only be used when the DMA hardware handshake mode is enabled.	0
2	INT_ABT_MSK	RW	Channel abort interrupt mask 0: No mask interrupt 1: Mask interrupt (Default)	1
1	INT_ERR_MSK	RW	Channel error interrupt mask 0: No mask interrupt 1: Mask interrupt (Default)	1
0	INT_TC_MSK	RW	Channel terminal count interrupt mask 0: No mask interrupt 1: Mask interrupt(Default)	1

Value after reset: 0000_2087h

Vangotech

4.12.6.16. Channel Source Address Register

Table 4-330. Channel Source Address Register (Cn_SrcAdd, Offset = 0x108 + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
[31:0]	Cn_SrcAddr	RW	Source starting address Please note that when the DMA transaction is complete, its value will be changed to the DMA source ending address.	0

4.12.6.17. Channel Destination Address Register

Table 4-331. Channel Destination Address Register (Cn_DstAddr, Offset = 0x10C + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
[31:0]	Cn_DstAddr	RW	Destination starting address Please note that when the DMA transaction is complete, its value will be changed to the DMA destination ending address.	0

4.12.6.18. Link List Descriptor Pointer

Table 4-332. Link List Descriptor Pointer (Cn_LLP, Offset = 0x110 + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
[31:2]	LLP_ADDR	RW	Link list descriptor pointer address	0
1	-	RsvdP	Reserved and Preserved	0
0	LLP_MASTER	RW	Master for loading the next LLP 0: Load the next LLP from AHB MASTER0 (Default) 1: Load the next LLP from AHB MASTER1 When DMAC_LINK_LIST is not defined, this bit is RO. When DMAC_LINK_LIST is defined and MAC_HAVE_AHB1_YES is not defined, this bit cannot be set to 1.	0

Value after reset: 0000_0000h

4.12.6.19. Transfer Size Register

Table 4-333. Transfer Size Register (Cn_SIZE, Offset = 0x114 + 0x20 * n, n = 0 ~ 7)

Bit	Name	Type	Description	Default Value
[31:22]	-	Rsvd	Reserved	0
[21:0]	TOT_SIZE	RW	Total transfer beats This register will be decreased when DMA reads data from slave.	0

Vangotech

4.13. ADC

4.13.1. Overview

The Analog-to-Digital converter (ADC) controller is a design to connect the FTADCC010 analog IP with the APB bus for the SoC integration.

4.13.2. Features

- APB compatible interface
- Supports 12-bit ADC hard macro
- Single step, single scan, and continuous scan conversion modes
- Interrupt generation at end of single conversion, end of single scan conversion, end of continuous conversion, threshold detection, and DMA channel data overrun/underrun events
- Programmable order and scan channel number for automatic conversions in scan mode
- External and software programming triggers
- Automatic power-down mode
- Programmable hardware delay insertion between conversions
- DMA request generation during scan conversion
- Supports 16 threshold detect sets
- Supports 4 DMA channels

4.13.3. Block Diagram

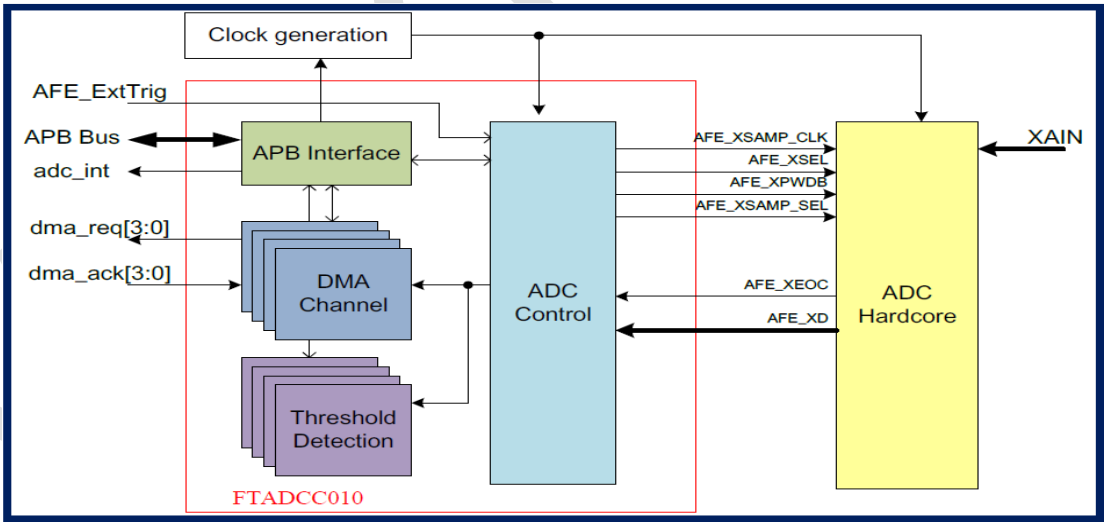


Figure 4-55. FTADCC010 Block Diagram

4.13.4. Functional Descriptions

4.13.4.1. Sampling Modes

The FTADCC010 controller is designed for ADC hard macros. Two sampling modes are implemented the controller. The external sampling clock mode is used, and the timing chart is shown in Figure 4-56,and the internal sampling mode is shown in Figure 4-57. The several timing parameters are controlled by

programmable registers and users should adjust these registers based on the FTADCC010 hard macro specification.

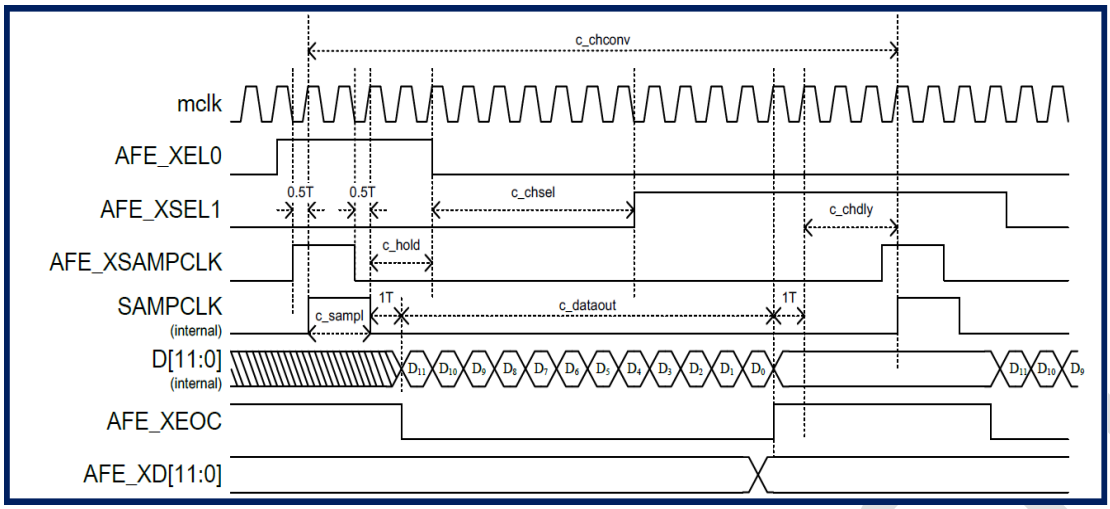


Figure 4-56. Timing Diagram of External Sampling Clock (External Sampling Mode)

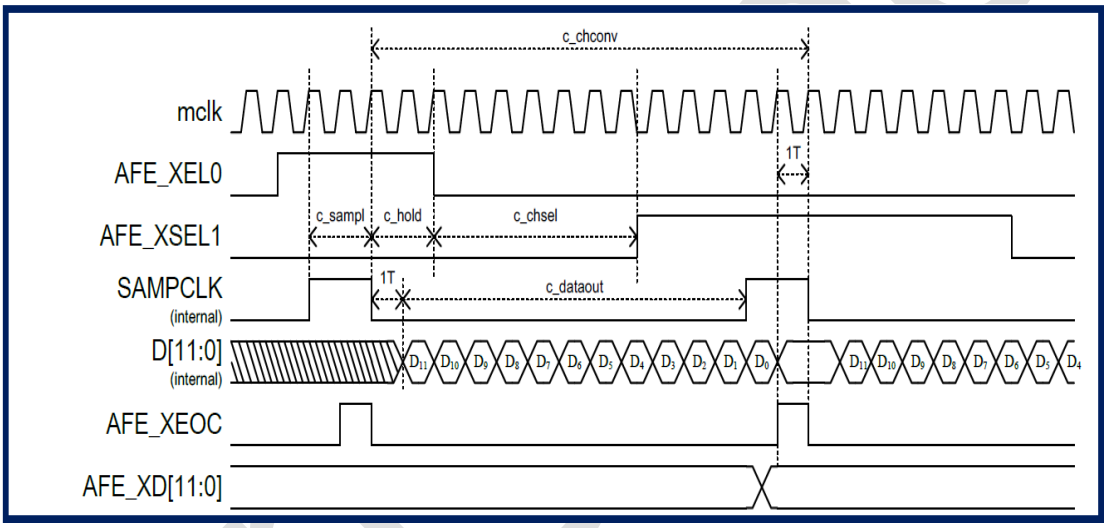


Figure 4-57. Timing Diagram of Internal Sampling Clock (Internal Sampling Mode)

4.13.4.2. Discharge Time

The scan discharge cycle is a programmable timing parameter to reserve for discharging capacitors of the pre-amplifier circuit in each FTADCC010 scan. The discharge time will be decided by the system application and FTADCC010 specification. An example timing chart is shown in Figure 4-58, the discharging cycles, c_dish, are programmed as 8 cycles of mclk. For the FTADCC010 controller interrupt, the event will be triggered when each scanning cycle is finished, the interrupt will be issued after discharging time.

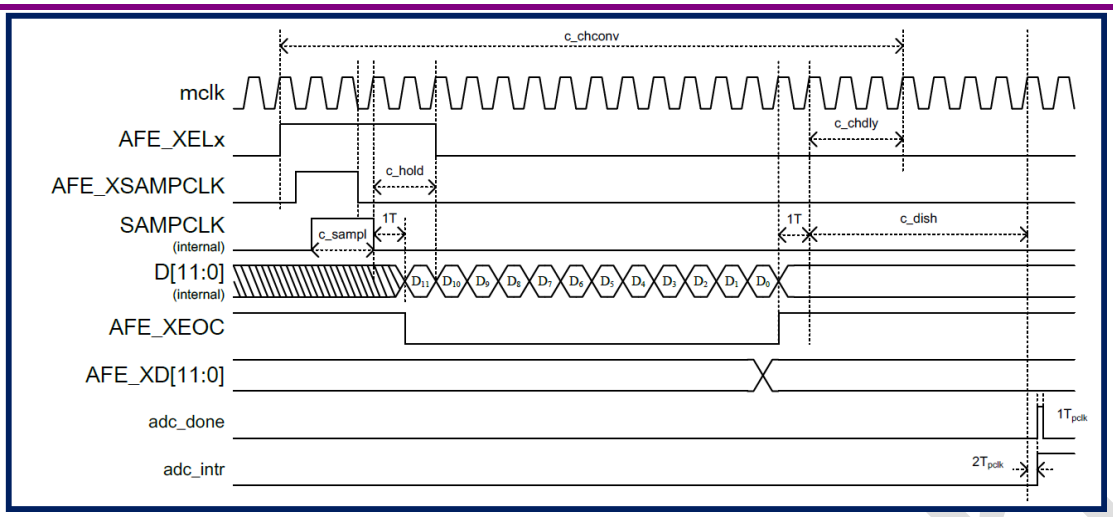


Figure 4-58. Timing Diagram of Scan Discharging Time

The channel selection discharge cycle is off all analog input before switching to the analog input channels. In Figure 4-58, the analog input selected is changed from AFE_XSEL0 to AFE_XSEL1, there are 7 channel disable cycles, c_chsel. Figure 4-59 shows a special case that AFE_XEOC is ready for the FTADCC010 conversion, and the channel disable cycles, c_chsel, will be extended after AFE_XEOC is low.

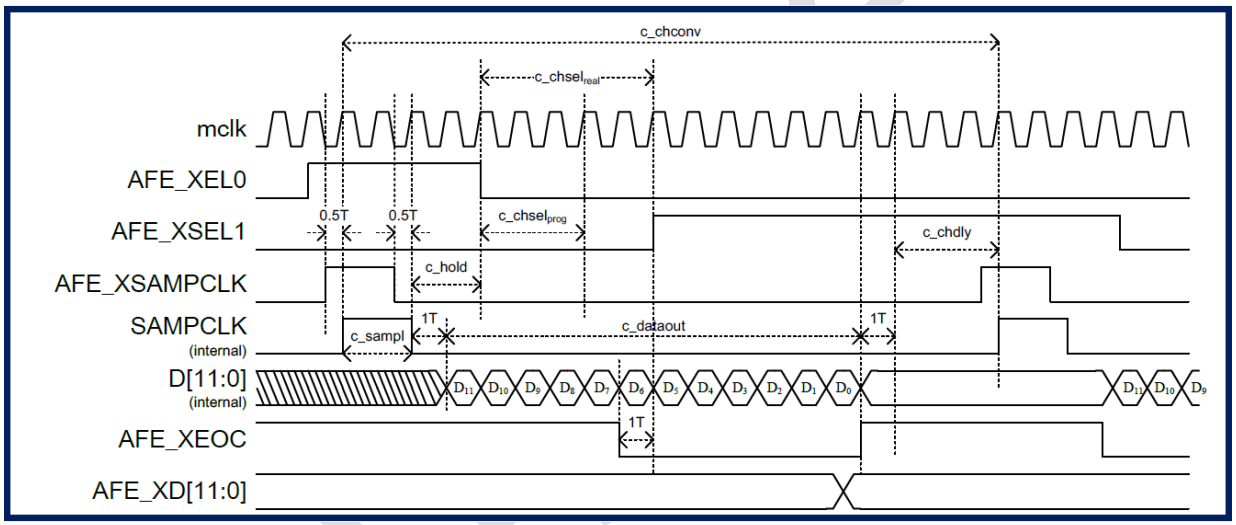
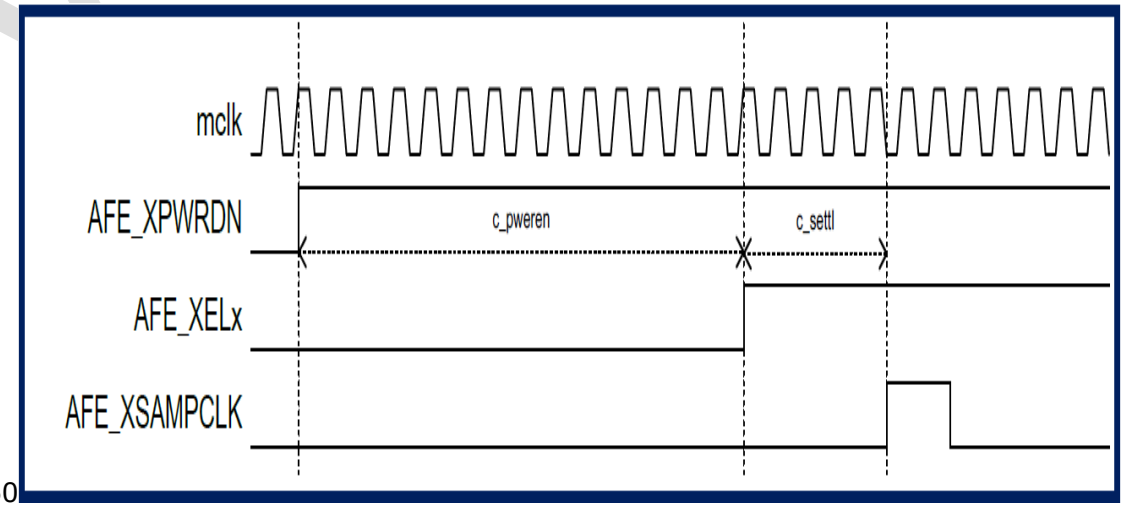


Figure 4-59. Timing Diagram of Channel Discharging Time when AFE_XEOC is unready

4.13.4.3. Power Enable Time

FTADCC010 needs to wait the wakeup time when FTADCC010 is turned on. An example of timing chart is shown in Figure



4-60

Figure 4-60. The FTADCC010 controller provides a programmable register of PWRENTIME to adjust the FTADCC010 wakeup time based on the mclk cycle time.

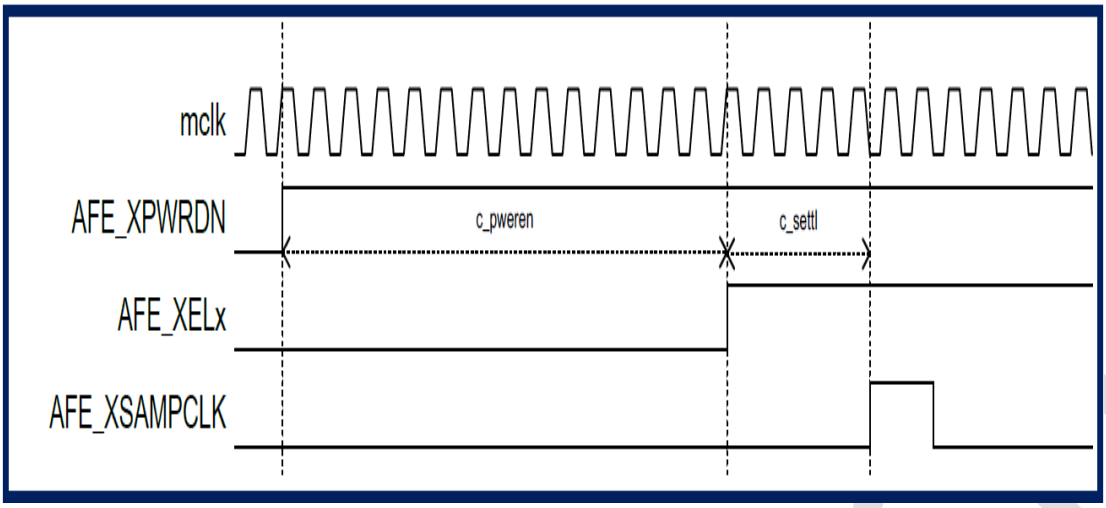


Figure 4-60. Timing Diagram of FTADCC010 Power Enable

4.13.4.4. Sampling Rate

For the FTADCC010 conversion cycles, an example of FTADCC010 is shown as Figure 4-61. It is programmed as external sample mode.

c_chconv = c_sampl + 1*Tmclk + c_dataout (12Tmclk) + 1*Tmclk+ c_chdly.

mclk can be adjusted by the MCLK_DIV register.

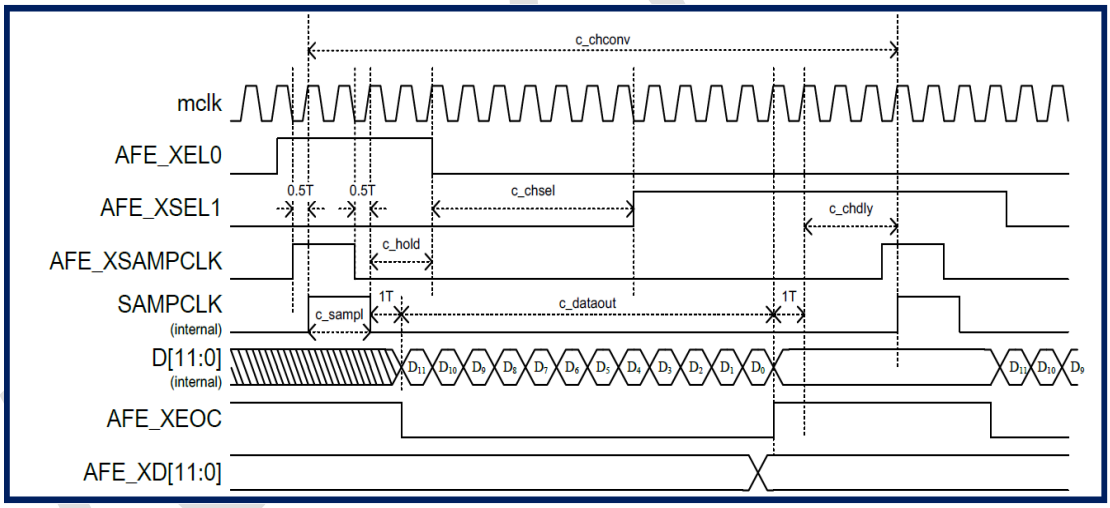


Figure 4-61. Timing Diagram of FTADCC010 Channel Sensing

4.13.4.5. Conversion Mode

There are three FTADCC010 conversion modes, single step conversion mode, single scan conversion mode, and continuous scan mode, which can be selected by programming COVN_MODE of ADC_CTRL register. In addition, users can select the number of scan conversion channels and the scan sequence by programming SCAN_NUM of ADC_CTRL register and ADC_SQR register. The timing diagram of each FTADCC010 scan is shown in

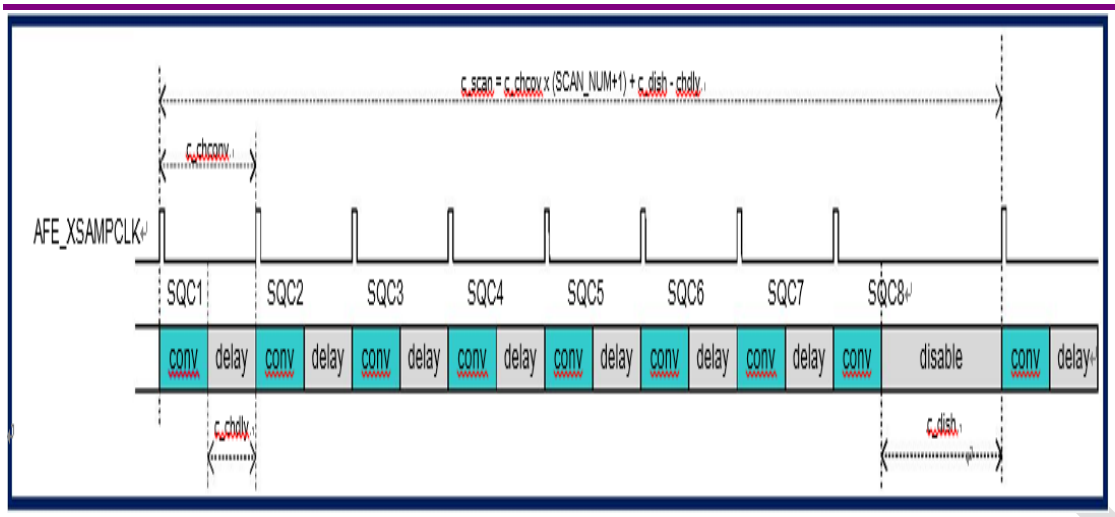


Figure **4-62**. The total sampling cycles, c_scan, are as below:

c_scan = c_chcov * (SCAN_NUM + 1) + c_dish - chdly.

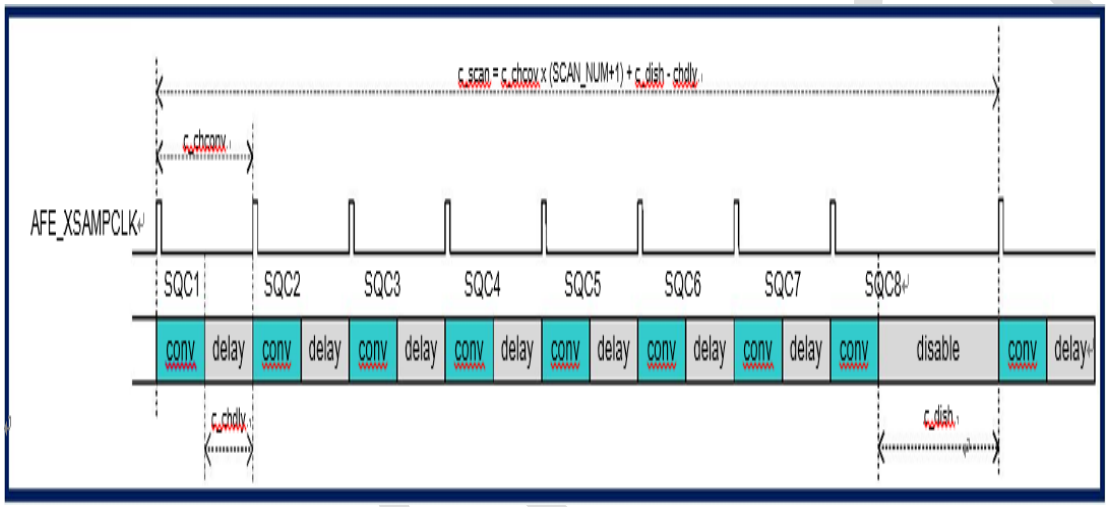


Figure 4-62. Timing Diagram of FTADCC010 Scan

In the FTADCC010 single step conversion mode, which COVN_MODE is set as 2'b00, the controller will wait for the trigger events, such as SWSTART or AFE_ExtTrig_IN to start FTADCC010 conversion until the number of scan conversion is done. Figure 4-63 shows the example of a single step conversion mode when SCAN_NUM = 3.

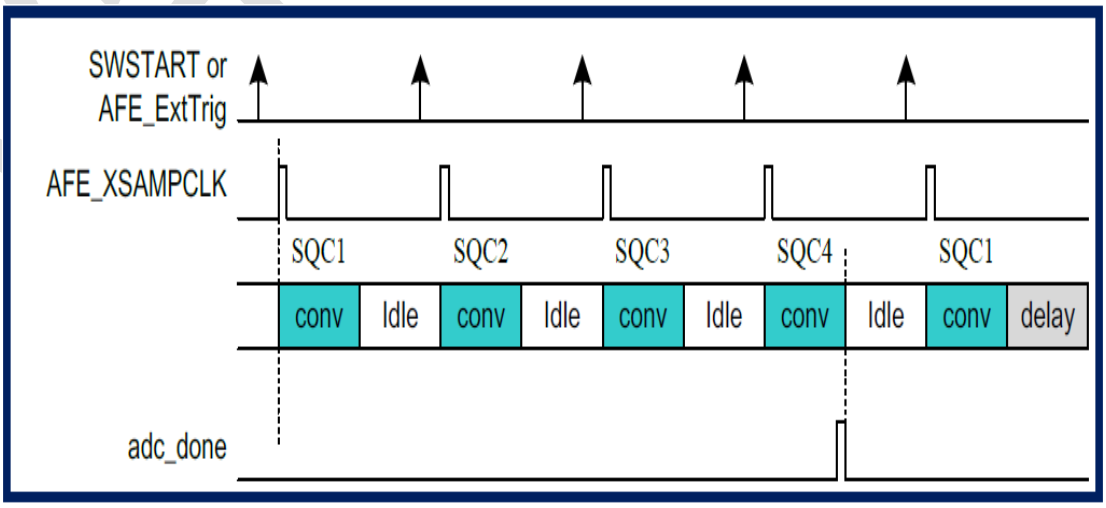


Figure 4-63. FTADCC010 Single Step Conversion Mode

The FTADCC010 single scan conversion mode is enabled when COVN_MODE is set as 2'b01. The controller

will execute an FTADCC010 scan cycle for all channels according to SCAN_NUM and scan sequence register when trigger event asserts.

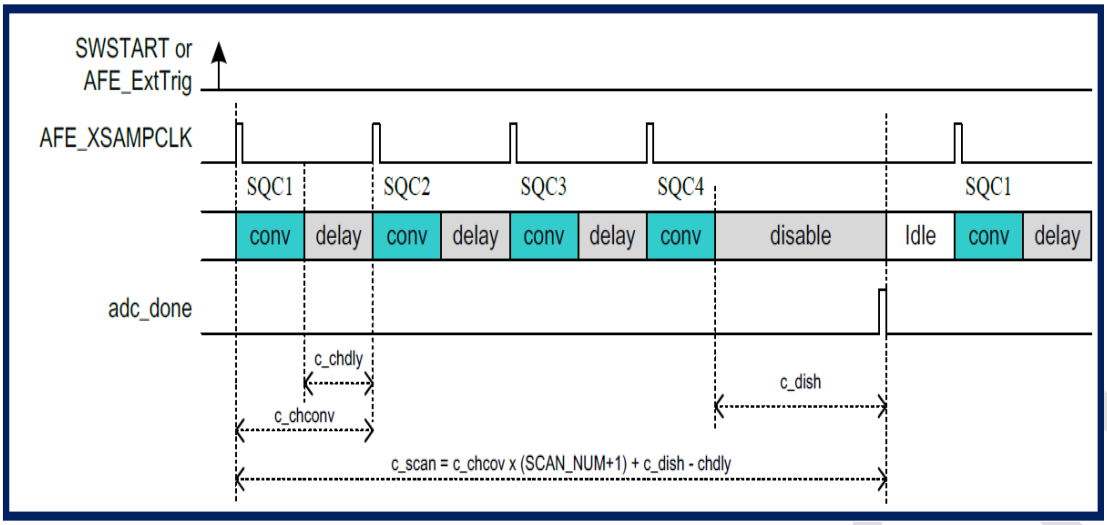


Figure 4-64. FTADCC010 Single Scan Conversion Mode

Similar to the single scan mode, the continuous scan conversion mode is to continuously perform FTADCC010 conversion until ADCC_EN is clear.

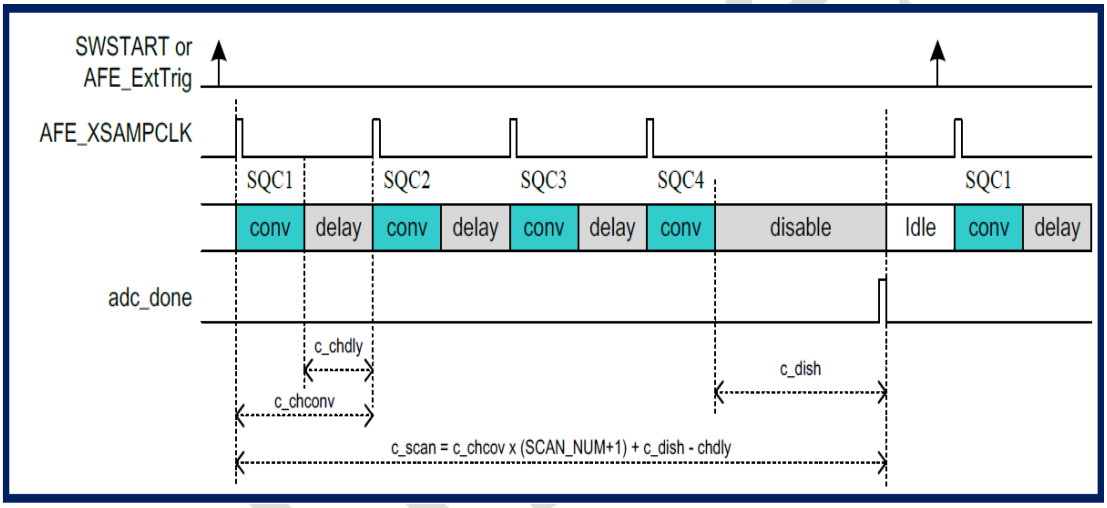


Figure 4-65. FTADCC010 Continuous Scan Conversion Mode

4.13.4.6. DMA Function

FTADCC010 supports up to four configurable DMA channels. Each DMA channel consists of four registers, data port register, function control register, interrupt enable register, and interrupt status register.

Users can determine the scan sequence by programming the scan sequence register. When FTADCC010 conversion starts, the controller will latch the FTADCC010 data to DMA FIFO. Users can select the data that is injected from all conversion data or from specific FTADCC010 channel to DMA FIFO by programming DMA_INJECT bit.

Figure 4-66 is an example that FIFO of DMA channel 0 is set to be injected from all FTADCC010 channels. FIFO of DMA channel 1 is set to be injected from FTADCC010 channel 3. DMA_CH in DMA control register selects the specific FTADCC010 channel to be latched.

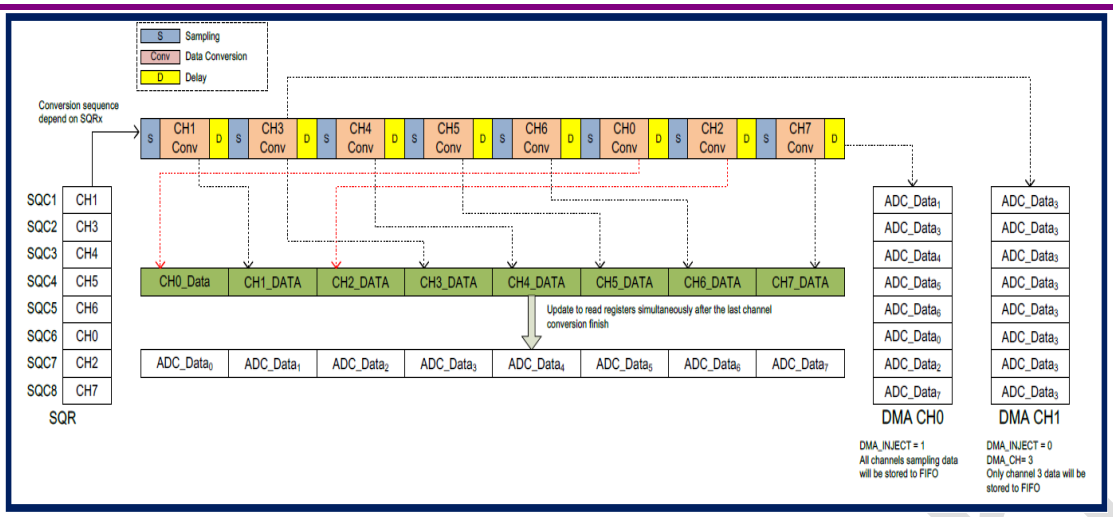


Figure 4-66. FTADCC010 Channel Conversion and DMA Function

4.13.4.7. Threshold Detection

Users can set up 16 threshold settings at 0x80~0x9f. The controller will monitor AFE_XD_IN and assert interrupt when AFE_XD_IN of corresponding FTADCC010 channel is over/under threshold as shown in Figure 4-67. Each ADC_THRHOLD register can set two threshold values, and each threshold value is comprised of THRHDx_MODE, THRHDx_CH, and THRHDx_VALUE.

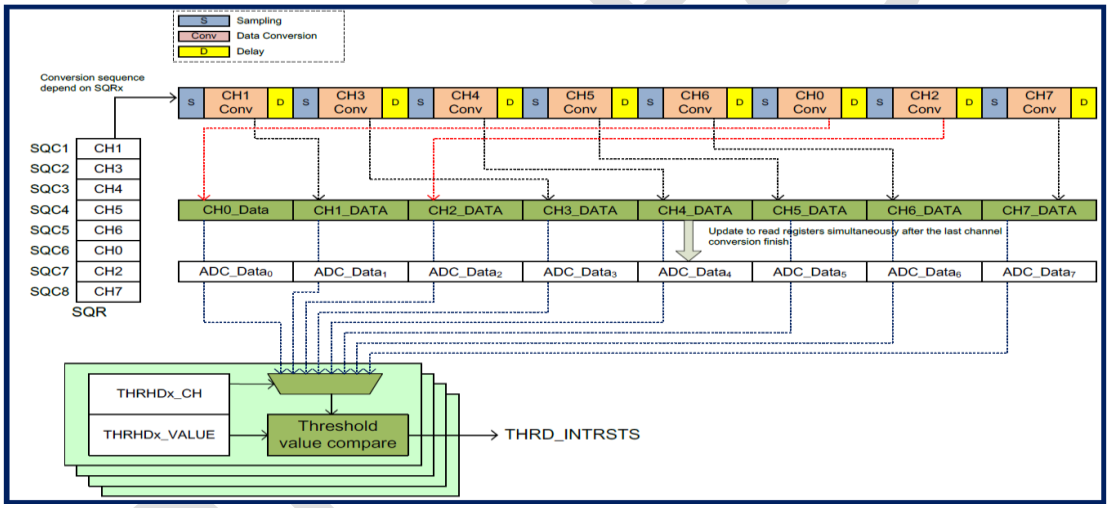


Figure 4-67. FTADCC010 Threshold Function

4.13.4.8. Auto Power-down Mode

FTADCC010 supports auto power-down feature to save power consumption if FTADCC010 supports internal power-down by the AFE_XPWRDN pin. When AUTOPWDN of ADC_CTRL register is set and the conversion mode is single step or single scan conversion mode, the controller will assert AFE_XPWRDN after FTADCC010 conversion is done. Figure 4-68 shows the timing of auto power-down function.

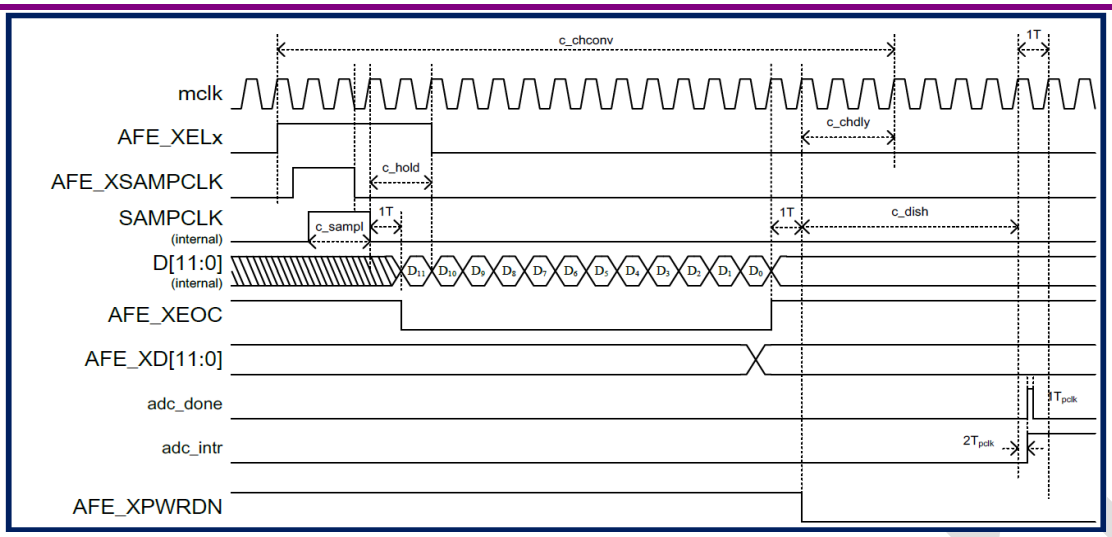


Figure 4-68. FTADCC010 Auto Power-down Function

4.13.4.9. Temperature Sensing Mode

FTADCC010 supports DC temperature sensing by enabling TEMP_EN bit. Figure 4-69 is the timing of temperature sensing.

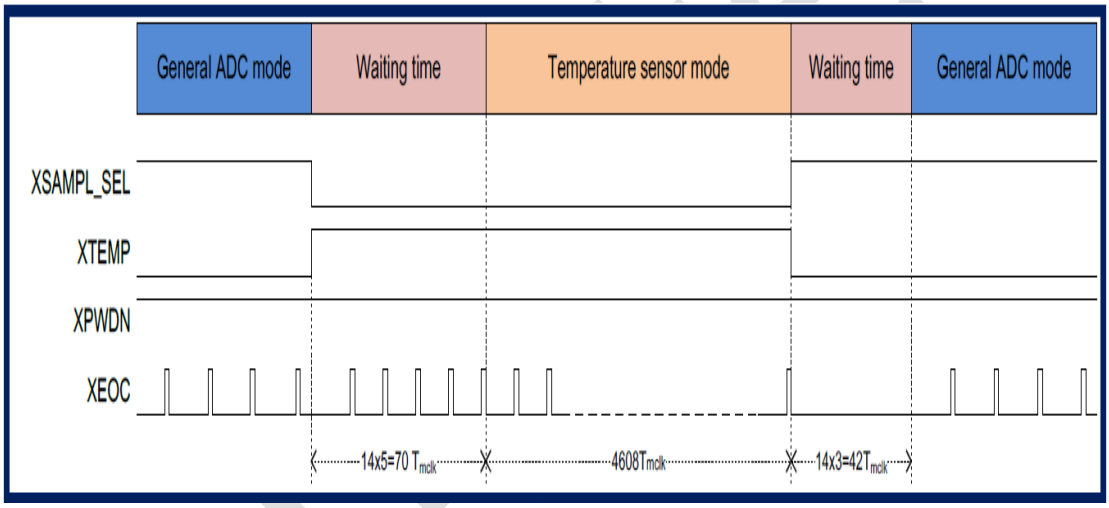


Figure 4-69. FTADCC010 Temperature Function

4.13.5. Register List

Module Name	Base Address
ADC	0x5440_0000

Register Name	Offset	Description
ADC_DATA	0x000 ~ 0x01F	Sensing data
ADC_THRHOLD	0x080 ~ 0x09F	FTADCC010 threshold detect programming registers
TS_DATA	0xC0	Temperature data register
TS_THRHOLD	0xC4	Temperature threshold detect programming registers
ADC_CTRL	0x100	FTADCC010 control register
ADC_MISC	0x104	FTADCC010 miscellaneous control register

ADC_INTEN	0x108	FTADCC010 interrupt enable register
ADC_INTR	0x10C	FTADCC010 interrupt status register
ADC_TPARM	0x110	FTADCC010 timing parameter register
ADC_SMPR	0x114	FTADCC010 sampling timing parameter control register
ADC_DISCHTIME	0x118	FTADCC010 sensing discharge time register
ADC_PRESCAL	0x11C	FTADCC010 clock pre-scaler control register
ADC_SQR	0x120	FTADCC010 scan sequence register
ADC_DMA0DAT	0x200	FTADCC010 DMA0 data port register
ADC_DMA0CTRL	0x204	FTADCC010 DMA0 function control register
ADC_DMA0INTEN	0x208	FTADCC010 DMA0 interrupt enable register
ADC_DMA0STS	0x20c	FTADCC010 DMA0 status register
ADC_DMA1DAT	0x210	FTADCC010 DMA1 data port register
ADC_DMA1CTRL	0x214	FTADCC010 DMA1 function control register
ADC_DMA1INTEN	0x218	FTADCC010 DMA1 interrupt enable register
ADC_DMA1STS	0x21c	FTADCC010 DMA1 status register
ADC_DMA2DAT	0x220	FTADCC010 DMA2 data port register
ADC_DMA2CTRL	0x224	FTADCC010 DMA2 function control register
ADC_DMA2INTEN	0x228	FTADCC010 DMA2 interrupt enable register
ADC_DMA2STS	0x22c	FTADCC010 DMA2 status register
ADC_DMA3DAT	0x230	FTADCC010 DMA3 data port register
ADC_DMA3CTRL	0x234	FTADCC010 DMA3 function control register
ADC_DMA3INTEN	0x238	FTADCC010 DMA3 interrupt enable register
ADC_DMA3STS	0x23c	FTADCC010 DMA3 status register

4.13.6. Register Description

4.13.6.1. Sensing Data Register

It is the sensing data read from FTADCC010 channels. There are eight 32-bit words for each channel. For example, 0x0 is for FTADCC010 channel 0 data and so on.

Table 4-334. Sensing Data Register (Offset: 0x000 ~ 0x01F)

Bit	Symbol	Access	Description	Default
[31:12]	-	-	Reserved	0x0
[11:0]	ADC_DATAn	RW	FTADCC010 channel n converter data	0x0

4.13.6.2. ADC Threshold Detect Programming Registers

The FTADCC010 controller supports up to 8 programming registers providing up to 16 threshold values for threshold detection. When AFE_XD_IN[n:0] is over or under the guarded area shown in Figure 4-70, the interrupt events will be triggered. The information of programming register is shown in Table 4-335.

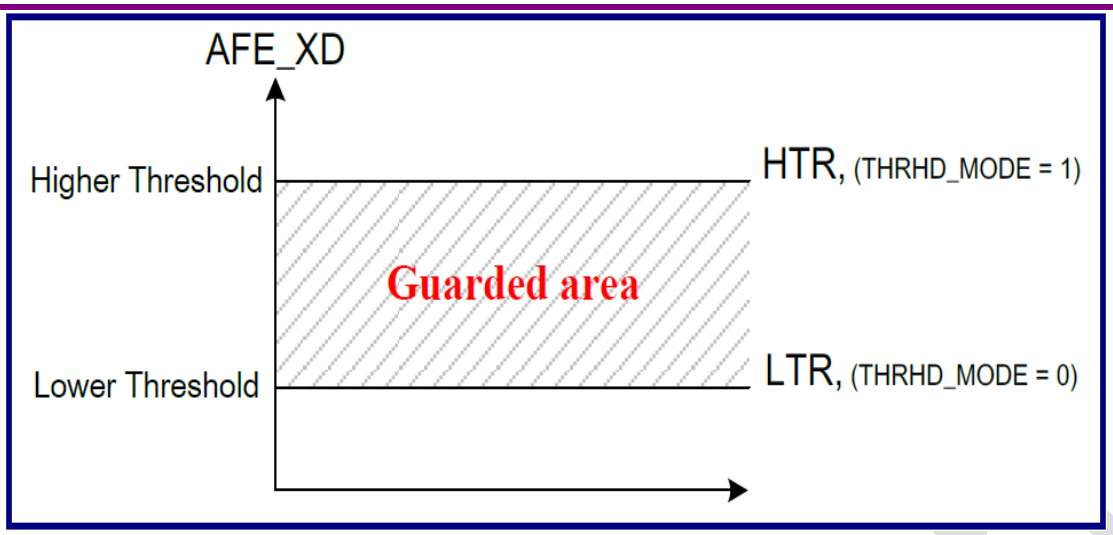


Figure 4-70. Guarded Area between Higher Threshold and Lower Threshold

Table 4-335. ADC Threshold Detect Programming Registers (Offset: 0x080 ~ 0x09F)

Bit	Symbol	Access	Description	Default
31	THRHD1_MODE	RW	Threshold detecting mode 1'b1: Over the threshold mode 1'b0: Under the threshold mode	0x0
[30:28]	THRHD1_CH	RW	Threshold detect for FTADCC010 channels For example, 0x0 is channel 0 and 0x1 is channel 1.	0x0
[27:16]	THRHD1_VALUE	RW	Threshold detect value programming register Higher threshold register (HTR): When THRHD1_MODE = 1, the register (THRHD1_VALUE) is a higher threshold programmable register for over threshold detection function. If all bits of THRHD1_VALUE are 1, it can disable the threshold detection function. Lower threshold register (LTR): When THRHD1_MODE = 0, the register (THRHD1_VALUE) is a lower threshold programmable register for under threshold detection function. If all bits of THRHD1_VALUE are 0, it can disable the threshold detection function.	0x0
15	THRHD0_MODE	RW	Threshold detecting mode 1'b1: Over the threshold mode 1'b0: Under the threshold mode	0x0
[14:12]	THRHD0_CH	RW	Threshold detect for FTADCC010 channels	0x0

Vangotech

			For example, 0x0 is channel 0 and 0x1 is channel 1.	
[11:0]	THRHD0_VALUE	RW	Threshold detect value programming register Higher threshold register (HTR): When THRHD0_MODE = 1, the register (THRHD0_VALUE) is a higher threshold programmable register for over threshold detection function. If all bits of THRHD0_VALUE are 1, it can disable the threshold detection function. Lower threshold register (LTR): When THRHD1_MODE = 0, the register (THRHD0_VALUE) is a lower threshold programmable register for under threshold detection function. If all bits of THRHD0_VALUE are 0, it can disable the threshold detection function.	0x0

4.13.6.3. Temperature Data Register

It presents when ADCC_TMEPSENSOR_ON is defined.

Table 4-336. Temperature Data Register (Offset: 0x0C0)

Bit	Symbol	Access	Description	Default
[31:12]	-	-	Reserved	0x0
[11:0]	TS_DATA	RW	FTADCC010 channel n converter data	0x0

4.13.6.4. Temperature Threshold Detect Programming Register

It presents when ADCC_TMEPSENSOR_ON is defined.

Table 4-337. Temperature Threshold Detect Programming Register (Offset: 0x0C4)

Bit	Symbol	Access	Description	Default
31	TS_HTR_EN	RW	Higher Threshold detecting enable	0x0
[30:28]	-	-	Reserved	0x0
[27:16]	TS_HTR	RW	Higher threshold register (HTR) The register is a higher threshold programmable register for over threshold detection function.	0x0
15	TS_LTR_EN	RW	Lower threshold detecting enable	0x0
[14:12]	-	-	Reserved	0x0
[11:0]	TS_LTR	RW	Lower threshold register (LTR) The register is a lower threshold programmable register for under threshold detection function.	0x0

Vangotech

4.13.6.5. ADC Control Register

Table 4-338. ADC Control Register (ADC_CTRL, Offset: 0x100)

Bit	Symbol	Access	Description	Default
[31:19]	-	-	Reserved	0x0
[18:16]	SCAN_NUM	RW	FTADCC010 scan conversion channel numbers 0x0: One channel in each scan 0x1: Two channels in each scan 0x2: Three channels in each scan … 0x7: Eight channels in each scan The number of the scan channels will be latched to execute conversion after external hardware or software programming trigger. In the latching interval, any register programming will not be immediately updated until next trigger event. After adc_enable is set, the scan channels will be fixed, it will not be changed by re-programming the register. The default value is the FTADCC010 channel number configured in ADCC_ADC_CHNUM_[n].	0x7
15	SAMPL_MODE	RW	Sample mode 1'b0: External sample mode 1'b1: Internal sample mode The sampling mode will be latched to execute conversion after external hardware or software programming trigger. In the latching interval, any register programming will not be immediately updated until next trigger event.	0x0
14	TEMP_EN	RW	Temperature sensor conversion enable 1'b0: General ADC conversion mode 1'b1: Temperature sensor conversion mode The temperature sensor or general conversion mode will be latched to execute conversion after external hardware or software programming trigger. In the latching interval, any register programming will not be immediately updated until next trigger event.	0x0
13	TRIG_MODE	RW	Trigger mode 1'b0: Conversion starts by programming	0x0

Vangotech

			SWSTART. 1'b1: Conversion starts by external hardware trigger.	
12	AUTOPWDN	RW	Auto power-down mode 1'b1: FTADCC010 automatically enters the power-down mode when FTADCC010 scan conversion finishes. FTADCC010 still keeps powering on after finishing FTADCC010 conversion when the bit is programmed to 1'b0. If users want to power down FTADCC010 by software, users can set the bit to1'b1 and clear ADCC_EN to shut FTADCC010 power down. The auto power-down mode will be latched to execute conversion after external hardware or software programming trigger. In the latching interval, any register programming will not be immediately updated until next trigger event.	0x0
[11:10]	-	-	Reserved	0x0
[9:8]	COVN_MODE	RW	FTADCC010 conversion mode 2'b00: Single step conversion mode 2'b01: Single scan conversion mode 2'b10: Continuous scan conversion mode 2'b11: Reserved	0x0
[7:5]	-	-	Reserved	0x0
4	SWSTART	WO	Software programming to trigger FTADCC010 conversion Always 0 for read	0x0
[3:1]	-	-	Reserved	0x0
0	ADCC_EN	RW	FTADCC010 control enable Controller will enable the FTADCC010 hard macro to enter the normal operation mode if it is set. 1'b1: Enable the FTADCC010 conversion function 1'b0: Disable the FTADCC010 function when finishing FTADCC010 conversion	0x0

4.13.6.6. ADC Miscellaneous Control Register

Table 4-339. ADC Miscellaneous Control Register (Offset: 0x104)

Vangotech

Bit	Symbol	Access	Description	Default
[31:0]	MISC_CTRL	RW	AFE_MISC_CTRL output value It is user-defined signal for the FTADCC010 hard macro side-band input signals. It maps to AFE_MISC_CTRL signals.	0x0

4.13.6.7. ADC Interrupt Enable Register

Table 4-340. ADC Interrupt Enable Register (Offset: 0x108)

Bit	Symbol	Access	Description	Default
[31:16]	THRD_INTREN	RW	Threshold detection interrupt enable	0x0
15	CH7_EOC_INTREN	RW	FTADCC010 channel 7 EOC interrupt enable	0x0
14	CH6_EOC_INTREN	RW	FTADCC010 channel 6 EOC interrupt enable	0x0
13	CH5_EOC_INTREN	RW	FTADCC010 channel 5 EOC interrupt enable	0x0
12	CH4_EOC_INTREN	RW	FTADCC010 channel 4 EOC interrupt enable	0x0
11	CH3_EOC_INTREN	RW	FTADCC010 channel 3 EOC interrupt enable	0x0
10	CH2_EOC_INTREN	RW	FTADCC010 channel 2 EOC interrupt enable	0x0
9	CH1_EOC_INTREN	RW	FTADCC010 channel 1 EOC interrupt enable	0x0
8	CH0_EOC_INTREN	RW	FTADCC010 channel 0 EOC interrupt enable	0x0
[7:4]	-	-	Reserved	0x0
3	TS_OVR_INTREN	RW	Temperature sensor over threshold interrupt enable	0x0
2	TS_UDR_INTREN	RW	Temperature sensor under threshold interrupt enable	0x0
1	ADC_STOP_INTREN	RW	FTADCC010 scan conversion stop interrupt enable	0x0
0	ADC_DONE_INTREN	RW	FTADCC010 scan conversion finish interrupt enable	0x0

4.13.6.8. ADC Interrupt Status Register

These bits of register record the interrupt status.

Table 4-341. ADC Interrupt Status Register (Offset: 0x10C)

Vangotech

Bit	Symbol	Access	Description	Default
[31:16]	THRD_INTRSTS	RW1C	Threshold detection interrupt status	0x0
15	CH7_EOC_INTRSTS	RW1C	FTADCC010 channel 7 EOC interrupt status	0x0
14	CH6_EOC_INTRSTS	RW1C	FTADCC010 channel 6 EOC interrupt status	0x0
13	CH5_EOC_INTRSTS	RW1C	FTADCC010 channel 5 EOC interrupt status	0x0
12	CH4_EOC_INTRSTS	RW1C	FTADCC010 channel 4 EOC interrupt status	0x0
11	CH3_EOC_INTRSTS	RW1C	FTADCC010 channel 3 EOC interrupt status	0x0
10	CH2_EOC_INTRSTS	RW1C	FTADCC010 channel 2 EOC interrupt status	0x0
9	CH1_EOC_INTRSTS	RW1C	FTADCC010 channel 1 EOC interrupt status	0x0
8	CH0_EOC_INTRSTS	RW1C	FTADCC010 channel 0 EOC interrupt status	0x0
7	DMACH3_INTRSTS	R	DMA Channel 3 interrupt status	0x0
6	DMACH2_INTRSTS	R	DMA Channel 2 interrupt status	0x0
5	DMACH1_INTRSTS	R	DMA Channel 1 interrupt status	0x0
4	DMACH0_INTRSTS	R	DMA Channel 0 interrupt status	0x0
3	TS_THDOD_INTRSTS	RW1C	Temperature over threshold detection interrupt status	0x0
2	TS_THDUD_INTRSTS	RW1C	Temperature under threshold detection interrupt status	0x0
1	ADC_STOP_INTRSTS	RW1C	FTADCC010 converter stop interrupt status When the ADCC_EN bit of 0x100 is set to 1'b0 during the FTADCC010 conversion, it is asserted when the ADC controller is already at the idle state.	0x0
0	ADC_DONE_INTRSTS	RW1C	FTADCC010 converter done interrupt status It is asserted when the FTADCC010 conversion is done	0x0

4.13.6.9. ADC Timing Parameter Register

The register is used to define the waiting cycles when the analog part is ready based on the input clock. PWRENTIME indicates that the power enable is set, it needs to wait cycles and then it can start to enable

the FTADCC010 sample function.

Table 4-342. ADC Timing Parameter Register (Offset: 0x0110)

Bit	Symbol	Access	Description	Default
[31:27]	-	-	Reserved	0x0
[26:24]	TSWKUPTIME	RW	Wakeup cycles of FTADCC010 temperature sensor conversion 0x0: Disable 0x1: Waiting 1 ADC conversion cycle 0x2: Waiting 2 ADC conversion cycles … 0x7: Waiting 7 ADC conversion cycles The register is valid when ADCC_TEMPSENSOR_ON is defined.	0x0
[23:19]	-	-	Reserved	0x0
[18:16]	WKUPTIME	RW	ADC wakeup cycles of ADC conversion 0x0: Disable 0x1: Waiting 1 ADC conversion cycle 0x2: Waiting 2 ADC conversion cycles … 0x7: Waiting 7 ADC conversion cycles	0x3
[15:12]	-	-	Reserved	0x0
[11:0]	PWRENTIME	RW	FTADCC010 power enable cycle number of mclk It should be programmed before the ADCC_EN bit is active. When the ADCC_EN bit is active, the value should be frozen. The valid bit width is defined by ADCC_TMRBITNUM_[n].	0x020

4.13.6.10. ADC Sampling Timing Parameter Control Register

Table 4-343. ADC Sampling Timing Parameter Control Register (Offset: 0x114)

Vangotech

Bit	Symbol	Access	Description	Default
[31:24]	CHDLYTime	RW	Delay time cycle number of mclk for each channel sampling It should be programmed before the ADCC_EN bit is active. When the ADCC_EN bit is active, the value should be frozen. The valid bit width is defined by ADCC_TMRBITNUM_[n].	0x03
23	-	-	Reserved	-
[22:20]	CHSELTime	RW	Analog input channel change discharge cycle number of mclk It should be programmed before the ADCC_EN bit is active. When the ADCC_EN bit is active, the value should be frozen. The valid bit width is defined by ADCC_TMRBITNUM_[n].	0x1
[19:18]	-	-	Reserved	-
[17:16]	HOLDTime	RW	Analog input channel hold cycle number of mclk It should be programmed before the ADCC_EN bit is active. When the ADCC_EN bit is active, the value should be frozen. The valid bit width is defined by ADCC_TMRBITNUM_[n].	0x2
[15:12]	-	-	Reserved	-
[11:8]	SAMPLTime	RW	Analog input channel sample clock high level cycle number of mclk	0x1
[7:0]	SETTLTime	RW	Analog input channel settling cycle number of mclk It should be programmed before the ADCC_EN bit is active. When the ADCC_EN bit is active, the value should be frozen. The valid bit width is defined by ADCC_TMRBITNUM_[n].	0x01

4.13.6.11. ADC Sensing Discharge Time Register

DISCHTIME is used to wait the discharging time for external analog frontend in each converter cycle.

Table 4-344. ADC Sensing Discharge Time Register (Offset: 0x118)

Bit	Symbol	Access	Description	Default
[31:12]	-	-	Reserved	0x0
[11:0]	DISCHTIME	RW	Sensing Discharge cycle number of mclk It should be programmed before any adc_pwren bit is active. When adc_pwren bit is active, the value should be frozen.	0x1

Vangotech

4.13.6.12. ADC Clock Pre-scaler Control Register

The register is to set MCLK for the FTADCC010 device. It will drive a mclk_div port to control an external divider that is implemented in the clock generation module to create a suitable clock for FTADCC010. Programming the FTADCC010 clock pre-scaler control register will trigger divrst_n to reset the external divider circuit.

Table 4-345. ADC Clock Pre-scaler Control Register (Offset: 0x11C)

Bit	Symbol	Access	Description	Default
[31:8]	-	-	Reserved	0x0
[7:0]	MCLK_DIV	RW	mclk divider factor mclk = (sys_clk/(mclk_div+1)) When mclk_div = 0, it is the bypass mode.	0x0

4.13.6.13. ADC Scan Sequence Register

The limitation is that one channel cannot be programmed in the scan sequence more than once.

Table 4-346. ADC Scan Sequence Register (Offset: 0x120)

Bit	Symbol	Access	Description	Default
[31]	-	-	Reserved	0x0
[30:28]	SQC8	RW	The 8th conversion channel selection for the scan mode	0x7
[27]	-	-	Reserved	0x0
[26:24]	SQC7	RW	The 7th conversion channel selection for the scan mode	0x6
[23]	-	-	Reserved	0x0
[22:20]	SQC6	RW	The 6th conversion channel selection for the scan mode	0x5
[19]	-	-	Reserved	0x0
[18:16]	SQC5	RW	The 5th conversion channel selection for the scan mode	0x4
[15]	-	-	Reserved	0x0
[14:12]	SQC4	RW	The 4th conversion channel selection for the scan mode	0x3
[11]	-	-	Reserved	0x0
[10:8]	SQC3	RW	The 3rd conversion channel selection for the scan mode	0x2
[7]	-	-	Reserved	0x0
[6:4]	SQC2	RW	The 2nd conversion channel selection for the scan mode	0x1
[3]	-	-	Reserved	0x0
[2:0]	SQC1	RW	The 1st conversion channel selection for the scan mode	0x0

4.13.6.14. ADC DMAn Data Port Register

Each data port register consists of two conversion data, which represents the even and odd data in FIFO. For example, if the first data in FIFO is 12'h3f1, the second data in FIFO is 12'h3f2 and so on, the data of the DMA data port register is 32'h03f2_03f1.

It presents when ADCC_DMA_ON is defined.

Table 4-347. ADC DMAn Data Port Register (Offset: 0x200, 0x210, 0x220, and 0x230)

Vangotech

Bit	Symbol	Access	Description	Default
[31:28]	-	-	Reserved	0x0
[27:16]	DMAx_FIFO1	RO	Even conversion data	0x0
[15:12]	-	-	Reserved	0x0
[11:0]	DMAx_FIFO0	RO	Odd conversion data	0x0

4.13.6.15. ADC DMAn Function Control Register

It presents when ADCC_DMA_ON is defined.

Table 4-348. ADC DMAn Function Control Register (Offset: 0x204, 0x214, 0x224, and 0x234)

Bit	Symbol	Access	Description	Default
[31:17]	-	-	Reserved	0x0
[16]	DMA_EN	RW	DMA enable	0x0
[15:10]	-	-	Reserved	0x0
[9:8]	DMA_THD	-	DMA threshold configuration 2'b11: Issue DMA request after full 2'b10: Issue DMA request after 8 data in FIFO 2'b01: Issue DMA request after 4 data in FIFO 2'b00: Issue DMA request after 2 data in FIFO	0x2
[7:6]	-	-	Reserved	0x0
[5:4]	DMA_INJECT	RW	FTADCC010 DMA injected data mode 2'b00: Injected data by DMA_CH select 2'b01: Injected data for all conversion 2'b1x: Injected data for temperature sensor conversion	0x0
3	-	-	Reserved	0x0
[2:0]	DMA_CH	RW	FTADCC010 channel select for DMA	0x0

4.13.6.16. ADC DMAn Interrupt Enable Register

It presents when ADCC_DMA_ON is defined.

Table 4-349. ADC DMAn Interrupt Enable Register (Offset: 0x208, 0x218, 0x228, and 0x238)

Bit	Symbol	Access	Description	Default
[31:2]	-	-	Reserved	0x0
1	DMA_OVR_INTEN	RW	DMA overrun interrupt enable	0x0
0	DMA_UDR_INTEN	RW	DMA underrun interrupt enable	0x0

4.13.6.17. ADC DMAn Interrupt Status Register

It presents when ADCC_DMA_ON is defined.

Table 4-350. ADC DMAn Interrupt Status Register (Offset: 0x20c, 0x21c, 0x22c, and 0x23c)

Bit	Symbol	Access	Description	Default

Vangotech

[31:29]	-	-	Reserved	0x0
[29:24]	FIFO_DAT_CNT	R	DMA FIFO count	0x0
[23:19]	-	-	Reserved	0x0
18	WPTR_FULL	R	DMA FIFO full	0x0
17	FIFO_HFULL	R	DMA FIFO half-full	0x0
16	RPTR_EMPTY	R	DMA FIFO empty	0x0
[15:2]	-	-	Reserved	0x0
1	DMA_OVR_STS	RW1C	DMA FIFO overrun status	0x0
0	DMA_UDR_STS	RW1C	DMA FIFO underrun status	0x0

Vangotech

4.14. TDC

4.14.1. Overview

TDC provides an APB interface wrapper to read temperature data from Temperature-to-Digital Converter (TDC) hard macro. It also supports monitor feature to assert interrupt when temperature is over/under threshold.

4.14.2. Features

- APB 3.0 compatible interface
- Programable timing parameters
- Threshold detection
- Support 12-bit resolution TDC
- Support 1 TDC hard macros
- Fully asynchronous design between TDC clock and PCLK
- Automatically power down TDC after conversion to save power consumption

4.14.3. Block Diagram

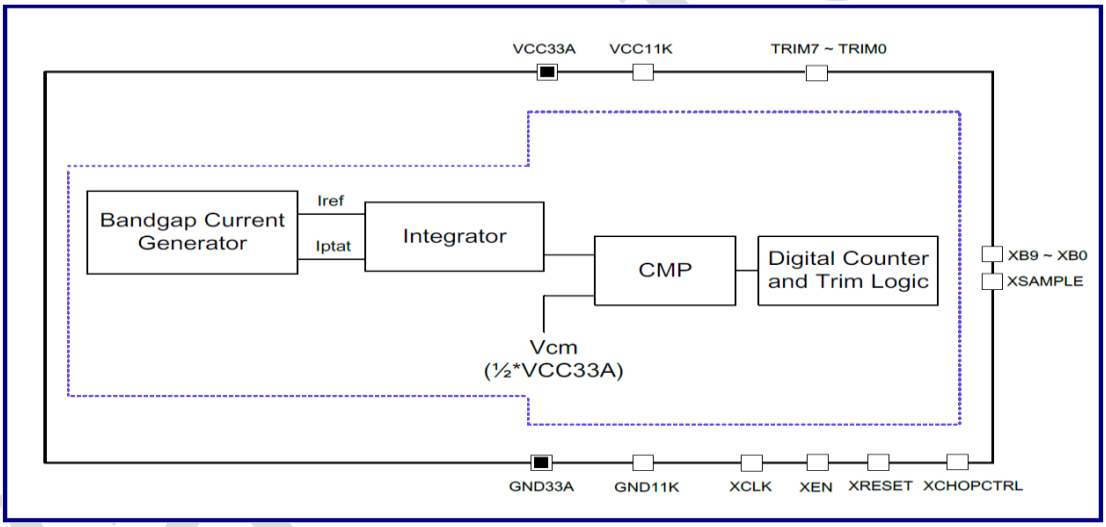


Figure 4-71. Functional Block Diagram

4.14.4. Electrical Specification

4.14.4.1. Electrical Characteristics

All tests are performed under the following conditions, unless stated otherwise:

The clock rate is 1.024 MHz; the typical values are obtained at the operating junction temperature (TJ) = 25 °C; VCC33A = 3.3 V; and VCC11K = 1.1 V.

Table 4-351. Electrical Characteristics

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
VCC33A	Analog power supply voltage	-	2.97	3.3	3.63	V
VCC11K	Internal core power supply voltage	-	0.99	1.1	1.21	V

RES	Resolution	-	-	10	-	bit
XCLK	Clock frequency	-	0.9216	1.024	1.1264	MHz
XSAMPLE	Sampling rate	-	-	1.0	-	KSPS
IVCC33A	Operation current of VCC33A (Simulation)	-	-	0.4	0.8	mA
IVCC11K	Operation current of VCC11K (Simulation)	-	-	0.1	15	μA
Ipd_VCC33A	Power-down current of VCC33A (Simulation)	-	-	0.05	5	μA
Ipd_VCC11K	Power-down current of VCC11K (Simulation)	-	-	0.1	10	μA
Temperature accuracy	Accuracy of the temperature sensor	-40 °C ~ 125 °C (After the trimming stage at 25°C)	-	-	±8.0	°C
Temperature resolution	Resolution of the temperature sensor	-40 °C ~ 125 °C	-	0.25	-	°C/LSB
TJ	Operating junction temperature	-	-40	-	125	°C

4.14.4.2. Timing Diagramof Sampling Clock

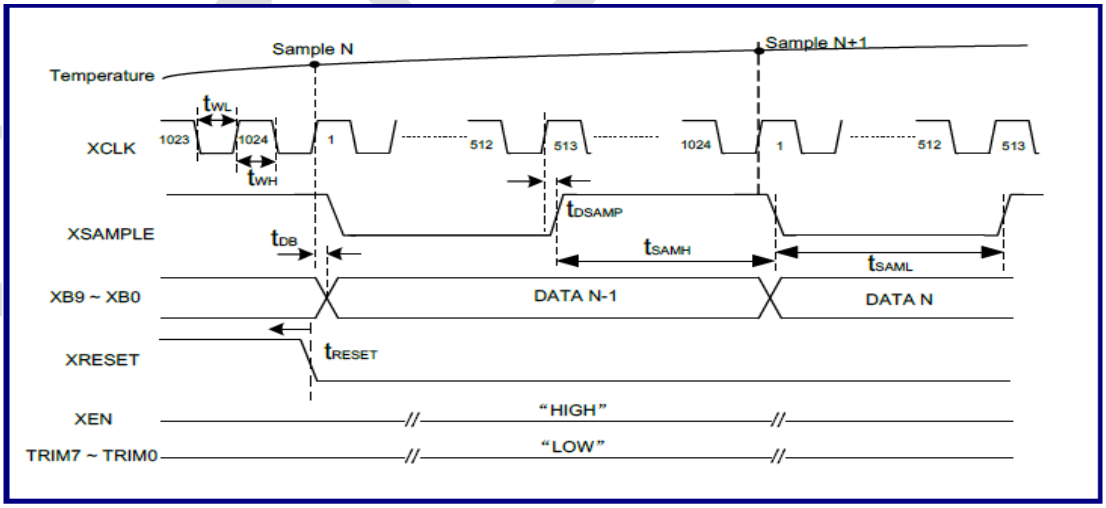


Figure 4-72. Timing Diagram of Sampling Clock

Table 4-352. Timing Parameters of Sampling Clock

Symbol	Description	Min.	Typ.	Max.	Unit
t_{WH}	Clock high-pulse width	-	0.488	-	μs

t_{WL}	Clock low-pulse width	-	0.488	-	μs
t_{SAMH}	High-pulse width of the enable signal (= 512 * (tWH+ tWL))	-	500	-	μs
t_{SAML}	Low-pulse width of the enable signal (= 512 * (tWH+ tWL))	-	500	-	μs
t_{RESET}	Initial reset time	-	0.1	-	μs
t_{DB}	Propagation delay from the XCLK rising edge to XB[9:0]	5.0	10	15	ns
t_{DSAMP}	Propagation delay from the CLK rising edge to XSAMPLE	5.0	10	15	ns

4.14.4.3. Wake-up Timing Requirement

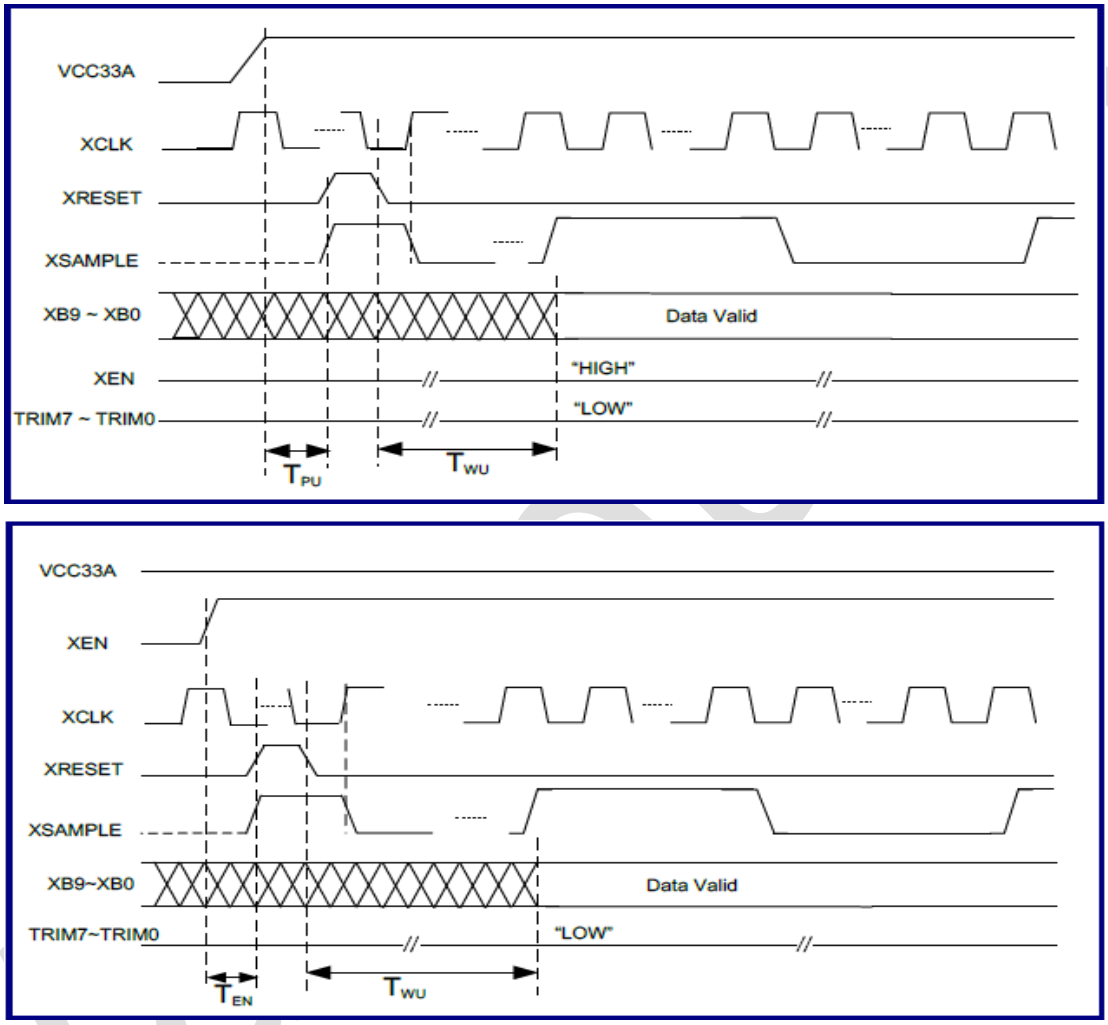


Figure 4-73. Wake-up Timing Requirement

Table 4-353. Wake-up Timing Specifications

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
TWU	Total wake-up time	-	-	3.0	-	ms
TPU	Delay time from power-on to XRESET	-	-	10	-	μs
TEN	Delay time from XEN to XRESET	-	-	1	-	μs

4.14.4.4. Trimming Table

Users should make sure that the temperature for the testing environment is at 25 °C, and the power supply provided to the chip is as clean as possible. The trimming input signals, TRIM7 ~ TRIM0, must be set to logic 0.

The digital outputs, XB9 ~ XB0, will be fetched by the logic analyzer and triggered by the XSAMPLE signal. At least 100 samples of XB9 ~ XB0 must be used to get the average value, named XB_avg[9:0]. First, compare XB_avg[9:0] and 10'b 0111011100 (Ideal output code is at 25°C). If XB_avg[9:0] 10'b 0111011100, please use the data in Table 4-354 for trimming; otherwise, use the data in Table 4-355 for trimming. Second, get the absolute difference between XB_avg[9:0] and 10'b 0111011100, named TRIM7 ~ TRIM0, and use the corresponding table for trimming.

Table 4-354. Trimming Table (1)

TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
0	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0

Table 4-355. Trimming Table (2)

TRIM7	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0
1	TRIM6	TRIM5	TRIM4	TRIM3	TRIM2	TRIM1	TRIM0

4.14.5. Functional Descriptions

The relationship between the temperature and output of TDC and the data format are listed in Table 4-356 as an example of an ideal case.

Table 4-356. Relationship between Temperature & Output of TDC and Data Format

Temperature (°C)	Output Code
-40	0011011000
-39.75	0011011001
....	.....
....	.....
24.75	0111011011
25	0111011100
25.25	0111011101
....	.....
....	.....
89.75	1011011111
90	1011100000
90.25	1011100001
....	.....
114.75	1101000011
115	1101000100
....	.....
....	.....
124.75	1101101011
125	1101101100

Vangotech

4.14.6. Register List

Module Name	Base Address
TDC	0x5450_0000

Register Name	Offset	Description
0x000	TDC_DATA0	TDC sensing data registers
0x080	TDC_THRHOLD0	Threshold detect registers
0x100	TDC_CTRL	TDC control register
0x104	TDC_TRIM	Trim register
0x108	TDC_INTEN	Interrupt enable register
0x10c	TDC_INTST	Interrupt status register
0x110	TDC_TPARAM0	Timing parameter 0 register
0x114	TDC_TPARAM1	Timing parameter 1 register
0x118	Reserved	Reserved
0x11c	TDC_PRESCAL	TDC clock pre-scaler register
0x13c	IP_REVISION	IP revision register

4.14.7. Register Description

4.14.7.1. TDC sensing data registers

Table 4-357. TDC sensing data registers (TDC_DATA0, offset: 0x000)

Bit	Symbol	Access	Description	Default
31:0	TDC_DATA	RW	TDC sensing data register	0x0

4.14.7.2. TDC Threshold registers

Table 4-358. Threshold registers (TDC_THRHOLD0, offset: 0x080)

Bit	Symbol	Access	Description	Default
31	HTHR_EN	RW	Higher threshold detecting enable 1'b0: disable 1'b1: enable	0x0
30:28	-	RO	Reserved	0x0
27:16	HTHR	RW	Higher threshold register (HTHR): The register is a higher threshold programmable register for over threshold detection function	0x0
15	LTHR_EN	RW	Lower Threshold detecting enable 1'b0: disable 1'b1: enable	0x0
14:12	-	RO	Reserved	0x0
11:0	LTHR	RW	Lower threshold register (LTHR) The register is a lower threshold programmable register for	0x0

Vangotech

			under threshold detection function.	

4.14.7.3. TDC control register

Table 4-359. TDC control register (TDC_CTRL, offset: 0x100)

Bit	Symbol	Access	Description	Default
31:24	CHOPCTRL	RW	Fixed to 1'b0 in normal mode	0x0
13:2	Reserved	RO	Reserved	0x0
1	SCANMODE	RW	1'b0: Single scan	0x0
0	TDCC_EN	RW	1'b1: enable TDC conversion function 1'b0: disable TDC function when TDC conversion finish	0x0

4.14.7.4. TDC Trim register

Table 4-360. Trim register (TDC_TRIM, offset: 0x104)

Trim register is reset by tdc_por_n.

Bit	Symbol	Access	Description	Default
31:8	Reserved	RO	Reserved	0x0
7:0	TRIM	RW	tdc_trim value	0x0

4.14.7.5. TDC Interrupt enable register

Table 4-361. Interrupt enable register (TDC_INTEN, offset: 0x108)

Bit	Symbol	Access	Description	Default
31:24	TDC_OVR_INTEN	RW	Over threshold interrupt enable	0x0
23:16	TDC_UNDR_INTEN	RW	Under threshold interrupt enable	0x0
15:8	TDC_CHDONE_INTEN	RW	TDC channel done interrupt enable	0x0
7:0	Reserved	RO	Reserved	0x0

4.14.7.6. TDC Interrupt status register

Table 4-362. Interrupt status register (TDC_INTST, offset: 0x10c)

Bit	Symbol	Access	Description	Default
31:24	TDC_OVR_INTST	RW1C	Over threshold interrupt status It is write one clear register.	0x0
23:16	TDC_UNDR_INTST	RW1C	Under threshold interrupt status It is write one clear register.	0x0
15:8	TDC_CHDONE_INTST	RW1C	TDC channel done interrupt status It is write one clear register.	0x0
7:3	Reserved	RO	Reserved	0x0
2	TDC_TH_INTST	RO	Threshold interrupt It is read only register. It is high when at least one bit of [31:16] is high.	0x0

Vangotech

1	TDC_DONE_INTST	RO	TDC done Interrupt It is read only register. It is high when at least one bit of [15:8] is high.	0x0
0	TDC_INTST	RO	Global TDC interrupt status It is read only register. It is high when one of the interrupt status is high.	0x0

4.14.7.7. TDC Timing Parameter 0 control register

Table 4-363. Timing Parameter 0 control register (TDC_TPARAM0, offset: 0x0110)

Bit	Symbol	Access	Description	Default
31:16	TWU	RW	Wakeup delay time	0x1003
15:12	Reserved	RO	Reserved	0x0
11:8	TRESET	RW	Reset time	0x01
7:4	TEN	RW	TDC enable delay time	0x2
3:0	TPU	RW	TDC power up delay time	0xB

4.14.7.8. TDC Timing Parameter 1 control register

Table 4-364. Timing Parameter 1 control register (TDC_TPARAM1, offset: 0x114)

Bit	Symbol	Access	Description	Default
31:26	Reserved	RO	Reserved	0x0
25:16	THOLD	RW	Hold time	0x2
15:10	Reserved	RO	Reserved	0x0
9:0	TSAMPLE	RW	Sample time	0x200

4.14.7.9. TDC clock pre-scaler register

Table 4-365. TDC clock pre-scaler register (TDC_PRESCAL, offset: 0x11c)

Bit	Symbol	Access	Description	Default
31:8	Reserved	RO	Reserved	0x0
7:0	XCLKDIV	RW	XCLK divider ratio	0x09

4.14.7.10. IP revision register

Table 4-366. IP revision register (IP_REVISION , offset: 0x13c)

Bit	Symbol	Access	Description	Default
31:0	IP_REVISION	RO	IP revision number	0x0001_0200

Vangotech

4.15. eFUSE

4.15.1. Overview

The controller is design for UMC eFuse hardcore to provide a program and read to/from eFuse hardcore.

4.15.2. Features

- APB 3.0 compatible interface
- Programable timing parameters
- On-chip program support without tester
- Write protected function
- eFuse page data read while power on

4.15.3. Register List

Module Name	Base Address
EFUSE	0x2200_0000

Name	Offset	Description
eFUSE_DATA	0x00~0xFFF	eFuse data register
TIMPARAM1	0x1000	Timing parameter 1 register
TIMPARAM2	0x1004	Timing parameter 2 register
PWRCTRL	0x1008	eFuse control and timing parameter 3 register
PAGE_STS	0x100c	Page data status register
PROTECT_KEY	0x1010	Protect key and status register
UNLOCK	0x1014	Unlock register (Write-only)

4.15.4. Register Description

4.15.4.1. eFuse data register

It is the eFuse page data register. The register number depends on MBITS parameter. For example, if MBITS = 512, it means 512-bit and the eFuse data registers will be 512/32 = 16 (0x00~0x3c).

Table 4-367. eFuse data register (Offset: 0x000~0xFFF.)

Bit	Symbol	Access	Description	Default
31:0	eFuse_DATA	RW	eFuse data register	0x0

Table 4-368. electronic signature

Address	Description	Default

0x044	Electronic signature rule version (ASCII)	0x30313031
0x048~ 0x050	corporate name(ASCII) 0x048:bit0-7:"S";bit8-31: Space character; 0x04C: bit24-31:"m";bit16-23:"a";bit8-15:"r" ;bit0-7:"t"; 0x050: bit24-31:"C";bit16-23:"h";bit8-15:"i" ;bit0-7:"p";	0x20202053 0x6D617274 0x43686970
0x054	Product version number(ASCII) bit8-15:"C" ;bit0-7:"0";bit16~31: Space character;	0x20204330
0x058~ 0x064	Product model (ASCII) 0x058: Space character; 0x05C: Space character; 0x060: bit16-23:"S";bit24-31:Space haracter;bit8-15:"C" ;bit0-7:"M" ; 0x064: bit24-31:"7";bit16-23:"0";bit8-15:"1" ;bit0-7:"D";	0x20202020 0x20202020 0x2053434D 0x37303144
0x068	Checksum 1	0x4700DB96
0x06C	Reserved	0x00000000
0x06E~ 0x073	Unique product identification 0x06E: Lot number(3 bytes) 0x071: Wafer number(1byte) 0x072: X coordinate(1byte) 0x073: Y coordinate(1byte)	
0x074	date of production(hex) 0x74:bit16~bit31:year; bit8~bit15:month; bit0~bit7: day; 0x078:bit16~bit23: hour;bit8~bit15: minute; bit0~bit7: second;	0x00000000
0x078		0x00000000
0x07C	Checksum 2 Invert the sum 0x06C to 0x078	

4.15.4.2. Timing parameter 1 register

Table 4-369. Timing parameter 1 register (Offset: 0x1000)

Bit	Symbol	Access	Description	Default
31:24	tEPS	RW	CSB falling to STROBE rising in program mode (tSUP_CS)	0x00
23:16	tPI	RW	STROBE pulse interval in program mode	0x1A
15:0	tPP	RW	STROBE pulse width in program mode	0x006A

4.15.4.3. Timing parameter 2 register

Table 4-370. Timing parameter 2 register (Offset: 0x1004)

Bit	Symbol	Access	Description	Default
31:16	tRC	RW	STROBE cycle time in read mode	0x0000
15:8	tRAS	RW	A[n:0] to STROBE setup time in read mode (tSUR_A)	0x00
7:0	tRPW	RW	STROBE pulse width in read mode (tRD)	0x01

4.15.4.4. eFuse control and timing parameter 3 register

Table 4-371. eFuse control and timing parameter 3 register (0x1008)

Bit	Symbol	Access	Description	Default
31:16	tPD	RW	PD to PS setup time in program mode (tSUP_PD_PS)	0x000E
15:12	tPS_CS	RW	PS to CSB setup time in program mode (tSUP_PS_CS)	0x10
11:10	rvsd	RO	Reserved	0x0
9:8	PWRMD	RW	Power Control mode Please fix to 1'b0 if not used.	0x0
7:4	tPD_PS	RW	PD to PS hold time in program mode (tHP_PD_PS)	0xD
3:2	rvsd	RO	Reserved	0x0
1	PWROFF	RW	Write 1 to disable eFuse power, write 0 don't care Please fix to 1'b0 if not used. Read: 1'b1: eFuse power down is in progress and will be clear to 1'b0 when finished.	0x0
0	PWREN_ST	RW	Write 1 to enable eFuse power, write 0 don't care Read for power enable pins status Please fix to 1'b0 if not used.	0x0

4.15.4.5. Page data status register

Table 4-372. Page data status register (Offset: 0x100c)

Bit	Symbol	Access	Description	Default
31:1	rvsd	RO	Reserved	0x0
0	FIFO_FLUSH	RW	For read: 1'b1: page data is not up-to-date. 1'b0: eFuse write operation is done. When eFuse is in write operation, the page data is not up-to-date. User needs to read eFuse data to update page data out bus. The page data out is valid when read eFuse or trigger initial read. For write: 1'b1: controller will force a page read in next eFuse read operation to update page data out bus. 1'b0: no effect.	0x0

4.15.4.6. Protect key and status register

Table 4-373. Protect key and status register (Offset: CTRL_REG_BASE+0x10) For write operation.

Bit	Symbol	Access	Description	Default
31:8	rvsd	RO	Reserved	0x0
7:0	PROTECT_KEY	WO	Protect key	n/a

Vangotech

			Write 0x55 and then write 0xAA to unprotect efuse in each programming.	

Table 4-374. Protect key and status register (Offset: CTRL_REG_BASE+0x10) For read operation.

Bit	Symbol	Access	Description	Default
31:1	rvsd	RO	Reserved	0x0
0	UNPROTECT	RO	Unprotect status. 0x1: Unprotect state 0x0: Protected state	0x0

4.15.4.7. Unlock register

Table 4-375. Unlock register (Offset: CTRL_REG_BASE+0x14)

Bit	Symbol	Access	Description	Default
31:8	rvsd	RO	Reserved	0x0
7:0	UNLOCK	WO	Unlock For write only. User needs to write 0x40 in unprotected state once before one eFuse programming.	n/a

Vangotech

4.16. Port Controller

Table 4-376. Extension Register Summary Table (Base Address = 0x2800_0000)

Offset	Default Value	Description
SPI		
0x0000_8500	0x00000040	SPI_CS_N IO control register
0x0000_8504	0x00000040	SPI_CS_CLK IO control register
0x0000_8508	0x00000040	SPI_DO IO control register
0x0000_850C	0x00000040	SPI_DI IO control register
0x0000_8510	0x00000040	SPI_WP_N IO control register
0x0000_8514	0x00000040	SPI_HOLD_N IO control register
EMMC0		
0x0000_8518	0x000000C8	EMMC0_DATA_7 IO control register
0x0000_851C	0x000000C8	EMMC0_DATA_6 IO control register
0x0000_8520	0x000000C8	EMMC0_DATA_5 IO control register
0x0000_8524	0x000000C8	EMMC0_DATA_4 IO control register
0x0000_8528	0x000000C8	EMMC0_DATA_3 IO control register
0x0000_852C	0x000000C8	EMMC0_DATA_2 IO control register
0x0000_8530	0x000000C8	EMMC0_DATA_1 IO control register
0x0000_8534	0x000000C8	EMMC0_DATA_0 IO control register
0x0000_8538	0x000000C8	EMMC0_RSTN IO control register
0x0000_853C	0x000000C0	EMMC0_CMD IO control register
0x0000_8540	0x000000C0	EMMC0_CLK IO control register
USB		
0x0000_8544	0x00000040	USB0_DRVBUS IO control register
0x0000_8548	0x00000000	USB0_VBUS IO control register
0x0000_854C	0x00000040	USB1_DRVBUS IO control register
0x0000_8550	0x00000000	USB1_VBUS IO control register
0x0000_8554	0x00000040	USB2_DRVBUS IO control register
0x0000_8558	0x00000000	USB2_VBUS IO control register
JTAG		
0x0000_855C	0x00000050	JTAG_TRST_N IO control register
0x0000_8560	0x00000040	JTAG_TDI IO control register
0x0000_8564	0x00000040	JTAG_SWDITMS IO control register
0x0000_8568	0x00000040	JTAG_SWCLKTCK IO control register
0x0000_856C	0x00000040	JTAG_TDO IO control register
SPI2AHB		
0x0000_8570	0x00000040	HSPI_CLK IO control register

Vangotech

0x0000_8574	0x00000040	HSPI_CS IO control register
0x0000_8578	0x00000040	HSPI_DI IO control register
0x0000_857C	0x00000040	HSPI_DO IO control register
I2C		
0x0000_8580	0x00000040	I2C0_CLK IO control register
0x0000_8584	0x00000040	I2C0_DATA IO control register
0x0000_8588	0x00000040	I2C1_CLK IO control register
0x0000_858C	0x00000040	I2C1_DATA IO control register
0x0000_8590	0x00000040	I2C2_CLK IO control register
0x0000_8594	0x00000040	I2C2_DATA IO control register
0x0000_8598	0x00000040	I2C3_CLK IO control register
0x0000_859C	0x00000040	I2C3_DATA IO control register
SSP		
0x0000_85A0	0x00000040	SPI0_CLK IO control register
0x0000_85A4	0x00000040	SPI0_CS IO control register
0x0000_85A8	0x00000040	SPI0_DI IO control register
0x0000_85AC	0x00000040	SPI0_DO IO control register
0x0000_85B0	0x00000040	SPI1_CLK IO control register
0x0000_85B4	0x00000040	SPI1_CS IO control register
0x0000_85B8	0x00000040	SPI1_DI IO control register
0x0000_85BC	0x00000040	SPI1_DO IO control register
0x0000_85C0	0x00000040	SPI2_CLK IO control register
0x0000_85C4	0x00000040	SPI2_CS IO control register
0x0000_85C8	0x00000040	SPI2_DI IO control register
0x0000_85CC	0x00000040	SPI2_DO IO control register
0x0000_85D0	0x00000040	SPI3_CLK IO control register
0x0000_85D4	0x00000040	SPI3_CS IO control register
0x0000_85D8	0x00000040	SPI3_DI IO control register
0x0000_85DC	0x00000040	SPI3_DO IO control register
0x0000_85E0	0x00000040	SPI4_CLK IO control register
0x0000_85E4	0x00000040	SPI4_CS IO control register
0x0000_85E8	0x00000040	SPI4_DI IO control register
0x0000_85EC	0x00000040	SPI4_DO IO control register
0x0000_85F0	0x00000040	SPI5_CLK IO control register
0x0000_85F4	0x00000040	SPI5_CS IO control register
0x0000_85F8	0x00000040	SPI5_DI IO control register
0x0000_85FC	0x00000040	SPI5_DO IO control register
0x0000_8600	0x00000040	SPI6_CLK IO control register
0x0000_8604	0x00000040	SPI6_CS IO control register

vangotech

0x0000_8608	0x00000040	SPI6_DI IO control register
0x0000_860C	0x00000040	SPI6_DO IO control register
UART		
0x0000_8610	0x00000040	UART0_TX IO control register
0x0000_8614	0x00000040	UART0_RX IO control register
0x0000_8618	0x00000040	UART1_TX IO control register
0x0000_861C	0x00000040	UART1_RX IO control register
0x0000_8620	0x00000040	UART2_TX IO control register
0x0000_8624	0x00000040	UART2_RX IO control register
0x0000_8628	0x00000040	UART2_NRTS IO control register
0x0000_862C	0x00000040	UART2_NCTS IO control register
0x0000_8630	0x00000040	UART3_TX IO control register
0x0000_8634	0x00000040	UART3_RX IO control register
0x0000_8638	0x00000040	UART3_NRTS IO control register
0x0000_863C	0x00000040	UART3_NCTS IO control register
0x0000_8640	0x00000040	UART4_TX IO control register
0x0000_8644	0x00000040	UART4_RX IO control register
SPI7(ESAM)		
0x0000_8648	0x00000040	SPI7_CLK IO control register
0x0000_864C	0x00000040	SPI7_CS IO control register
0x0000_8650	0x00000040	SPI7_DI IO control register
0x0000_8654	0x00000040	SPI7_DO IO control register
EMMC1		
0x0000_8658	0x000000C8	EMMC1_DATA_7 IO control register
0x0000_865C	0x000000C8	EMMC1_DATA_6 IO control register
0x0000_8660	0x000000C8	EMMC1_DATA_5 IO control register
0x0000_8664	0x000000C8	EMMC1_DATA_4 IO control register
0x0000_8668	0x000000C8	EMMC1_DATA_3 IO control register
0x0000_866C	0x000000C8	EMMC1_DATA_2 IO control register
0x0000_8670	0x000000C8	EMMC1_DATA_1 IO control register
0x0000_8674	0x000000C8	EMMC1_DATA_0 IO control register
0x0000_8678	0x000000C0	EMMC1_RSTN IO control register
0x0000_867C	0x000000C8	EMMC1_CMD IO control register
0x0000_8680	0x000000C0	EMMC1_CLK IO control register
I2S		
0x0000_8684	0x00000040	I2S0_FS IO control register
0x0000_8688	0x00000040	I2S0_RXD IO control register
0x0000_868C	0x00000040	I2S0_SCLK IO control register
0x0000_8690	0x00000040	I2S0_TXD IO control register

vangotech

irDA		
0x0000_8694	0x00000040	UART5_TX IO control register
0x0000_8698	0x00000040	UART5_RX IO control register
0x0000_869C	0x00000040	UART5_RX_h IO control register
0x0000_86A0	0x00000040	UART6_TX IO control register
0x0000_86A4	0x00000040	UART6_RX IO control register
0x0000_86A8	0x00000040	UART7_TX IO control register
0x0000_86AC	0x00000040	UART7_RX IO control register
0x0000_86B0	0x00000040	UART8_TX IO control register
0x0000_86B4	0x00000040	UART8_RX IO control register
0x0000_86B8	0x00000040	UART8_RX_h IO control register
0x0000_86BC	0x00000040	UART9_RX IO control register
0x0000_86C0	0x00000040	UART9_TX IO control register
RGMII		
0x0000_86C4	0x00000088	RGMII_RX_CK IO control register
0x0000_86C8	0x00000080	RGMII_RXD0 IO control register
0x0000_86CC	0x00000080	RGMII_RXD1 IO control register
0x0000_86D0	0x00000080	RGMII_RXD2 IO control register
0x0000_86D4	0x00000080	RGMII_RXD3 IO control register
0x0000_86D8	0x00000080	RGMII_RXCTL IO control register
0x0000_86DC	0x00000080	RGMII_GTX_CK IO control register
0x0000_86E0	0x00000080	RGMII_TXCTL IO control register
0x0000_86E4	0x00000080	RGMII_TXD0 IO control register
0x0000_86E8	0x00000080	RGMII_TXD1 IO control register
0x0000_86EC	0x00000080	RGMII_TXD2 IO control register
0x0000_86F0	0x00000080	RGMII_TXD3 IO control register
0x0000_86F4	0x00000080	RGMII_MDC IO control register
0x0000_86F8	0x00000080	RGMII_MDIO IO control register
RMII		
0x0000_86FC	0x00000040	RMII_TXD0 IO control register
0x0000_8700	0x00000040	RMII_TXD1 IO control register
0x0000_8704	0x00000040	RMII_TX_EN IO control register
0x0000_8708	0x00000040	RMII_RX_CRSDV IO control register
0x0000_870C	0x00000040	RMII_RX_ER IO control register
0x0000_8710	0x00000040	RMII_RXD0 IO control register
0x0000_8714	0x00000040	RMII_RXD1 IO control register
0x0000_8718	0x00000040	RMII_REF_CLK IO control register
0x0000_871C	0x00000040	RMII_MDC IO control register
0x0000_8720	0x00000040	RMII_MDIO IO control register

vangotech

MII		
0x0000_8724	0x00000040	MII_TXD0 IO control register
0x0000_8728	0x00000040	MII_TXD1 IO control register
0x0000_872C	0x00000040	MII_TXD2 IO control register
0x0000_8730	0x00000040	MII_TXD3 IO control register
0x0000_8734	0x00000040	MII_TX_EN IO control register
0x0000_8738	0x00000040	MII_RX_DV IO control register
0x0000_873C	0x00000040	MII_RX_ER IO control register
0x0000_8740	0x00000040	MII_RXD0 IO control register
0x0000_8744	0x00000040	MII_RXD1 IO control register
0x0000_8748	0x00000040	MII_RXD2 IO control register
0x0000_874C	0x00000040	MII_RXD3 IO control register
0x0000_8750	0x00000040	MII_TX_ER IO control register
0x0000_8754	0x00000040	MII_CRS IO control register
0x0000_8758	0x00000040	MII_COL IO control register
0x0000_875C	0x00000040	MII_TX_CK IO control register
0x0000_8760	0x00000040	MII_MDC IO control register
0x0000_8764	0x00000040	MII_MDIO IO control register
0x0000_8768	0x00000040	MII_RXCK control register
LCDC		
0x0000_876C	0x000000C0	LC_PCLK IO control register
0x0000_8770	0x000000C0	LC_VS IO control register
0x0000_8774	0x000000C0	LC_HS IO control register
0x0000_8778	0x000000C0	LC_DE IO control register
0x0000_877C	0x000000C0	LC_DAT0 IO control register
0x0000_8780	0x000000C0	LC_DAT1 IO control register
0x0000_8784	0x000000C0	LC_DAT2 IO control register
0x0000_8788	0x000000C0	LC_DAT3 IO control register
0x0000_878C	0x000000C0	LC_DAT4 IO control register
0x0000_8790	0x000000C0	LC_DAT5 IO control register
0x0000_8794	0x000000C0	LC_DAT6 IO control register
0x0000_8798	0x000000C0	LC_DAT7 IO control register
0x0000_879C	0x000000C0	LC_DAT8 IO control register
0x0000_87A0	0x000000C0	LC_DAT9 IO control register
0x0000_87A4	0x000000C0	LC_DAT10 IO control register
0x0000_87A8	0x000000C0	LC_DAT11 IO control register
0x0000_87AC	0x000000C0	LC_DAT12 IO control register
0x0000_87B0	0x000000C0	LC_DAT13 IO control register
0x0000_87B4	0x000000C0	LC_DAT14 IO control register

vangotech

0x0000_87B8	0x000000C0	LC_DAT15 IO control register
0x0000_87BC	0x000000C0	LC_DAT16 IO control register
0x0000_87C0	0x000000C0	LC_DAT17 IO control register
0x0000_87C4	0x000000C0	LC_DAT18 IO control register
0x0000_87C8	0x000000C0	LC_DAT19 IO control register
0x0000_87CC	0x000000C0	LC_DAT20 IO control register
0x0000_87D0	0x000000C0	LC_DAT21 IO control register
0x0000_87D4	0x000000C0	LC_DAT22 IO control register
0x0000_87D8	0x000000C0	LC_DAT23 IO control register
0x0000_87DC	0x00000040	I2S_MCLK IO control register
0x0000_87E0	0x00000040	SPI_DCX1 IO control register
0x0000_87E4	0x00000040	SPI_DCX2 IO control register
0x0000_87E8	0x00000040	DDRCK IO control register
0x0000_87EC	0x00000040	PSW0 IO control register
0x0000_87F4	0x00000001	SYS_XTAL IO control register

Table 4-377. IO Control Register (Offset: 0x0000_8500~0x0000_87F4)

Bit	Symbol	Access	Description	Default
31:10	-	-	Reserved	-
9	SMT	RW	1: Schmitt-trigger	Table 4-198
8:5	DCSR	RW	[8]: Slew rate 1: Slow [7:6]: Output driving capability for EMMC pads 00: 10 mA at3.3 V , 4 mA at 1.8 V 01: 15 mA at 3.3 V , 6 mA at 1.8 V 10: 7.5 mA at 3.3 V, 3 mA at 1.8 V 11: 5 mA at 3.3 V, 2 mA at 1.8 V [7:6]: Output driving capability for normal pads 00: 4 mA 01: 8 mA 10: 12 mA 11: 16 mA [5]: Reserved	Table 4-198
4	PD	RW	1: 75K at 3.3 V Pull Down	Table 4-198
3	PU	RW	1: 75K at 3.3 V Pull Up	Table 4-198
2:0	SEL	RW	Pin mux mode	Table 4-198

Vangotech

4.17. TRNG

4.17.1. Block diagram

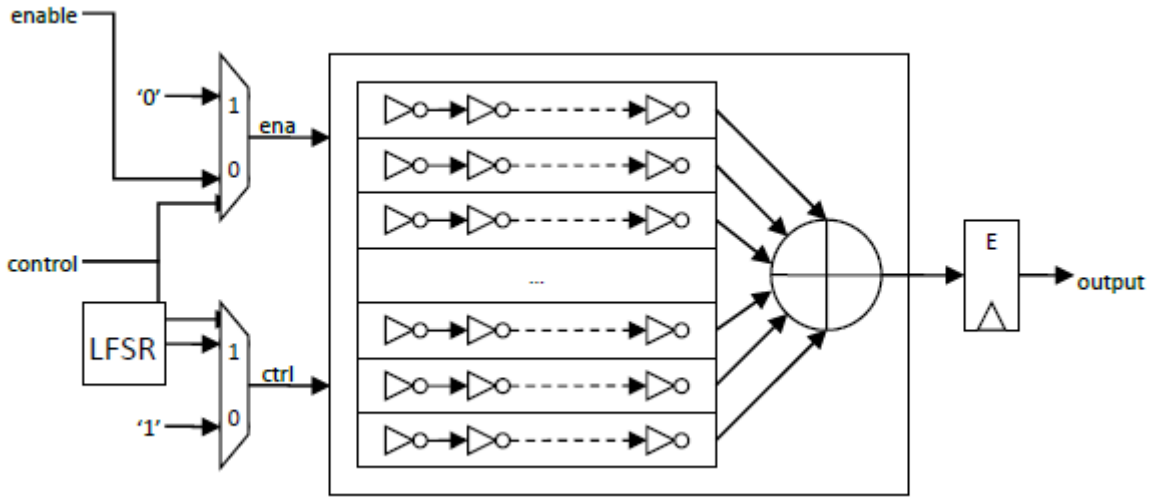


Figure 4-74. Entropy source block diagram

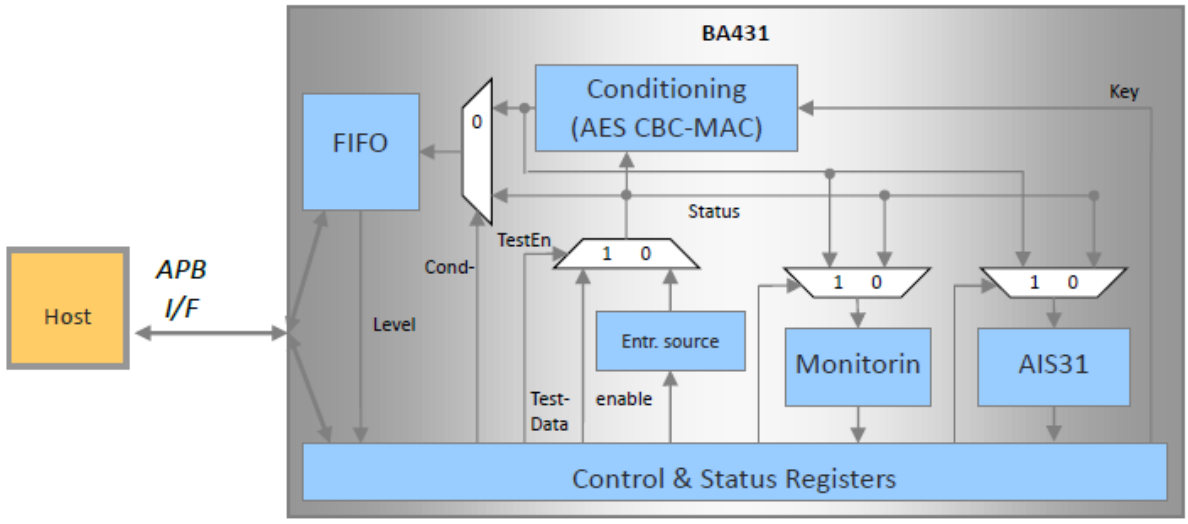


Figure 4-75. BA431 block diagram

4.17.2. Register map

The TRNG is accessed on 32-bit word boundaries and the lowest two bits of the address are not used.

Table 4-378 Register map

Offset	Name	Access	Description	Default Value
0x0000	ControlReg	R/W	Control register	0x00040000
0x0004	FIFOLevelReg	R/W	FIFO level register	0x00000000

0x0008	FIFOThreshReg	R/W	FIFO threshold register	$2^{g\_log2fifodepth-2}-1$
0x000C	FIFODepthReg	R	FIFO depth register	$2^{g\_log2fifodepth}$
0x0010	Key0Reg	R/W	Key register (MSB)	0x00000000
0x0014	Key1Reg	R/W	Key register	0x00000000
0x0018	Key2Reg	R/W	Key register	0x00000000
0x001C	Key3Reg	R/W	Key register (LSB)	0x00000000
0x0020	TestDataReg	W	Test data register	-
0x0024	RepThresReg	R/W	Repetition Count Threshold register	0x00000029
0x0028	PropThresReg	R/W	Adaptive Proportion Threshold register (1024-sample window)	0x00000319
0x002C	-	-	Reserved	-
0x0030	StatusReg	R/W	Status register	0x00000000
0x0034	InitWaitVal	R/W	Initial wait counter value	0x0000FFFF
0x0038	DisableOsc0	R/W	Disable oscillator rings	0x00000000
0x003C	DisableOsc1	R/W	Disable oscillator rings	0x00000000
0x0040	SwOffTmrVal	R/W	Switch off timer value	0x0000FFFF
0x0044	ClkDiv	R/W	Sample clock divider	0x00000000
0x0048	AIS31Conf0	R/W	AIS31 configuration register 0	0x43401040
0x004C	AIS31Conf1	R/W	AIS31 configuration register 1	0x03C00680
0x0050	AIS31Conf2	R/W	AIS31 configuration register 2	0x04400340
0x0054	AIS31Status	R/W	AIS31 status register	0x00000000
0x0058	HwConfig	R	Hardware configuration register	0x00000137

4.17.3. Register Description

4.17.3.1. Control register

Table 4-379 Control register

Bits	Name	Access	Description
[31:21]		-	Reserved
20	FifoWriteStartUp	R/W	Enable write of the samples in the FIFO during start-up
[19:16]	Nb128BitBlocks	R/W	Number of 128 bit blocks used in AES-CBCMAC post-processing. This value cannot be zero.
15	AIS31TestSel	R/W	Select input to the AIS31 test module 0: Before conditioning 1: After conditioning
14	HealthTestSel	R/W	Select input to health test module 0: Before conditioning 1: After conditioning

Vangotech

13	AIS31Bypass	R/W	Bypass AIS31 tests such that the results of the start-up and online tests do not affect the FSM state
12	HealthTestBypass	R/W	Bypass NIST tests such that the results of the start-up and online test do not affect the FSM state
11	ForceRun	R/W	Force oscillators to run when FIFO is full
10	IntEnAlm	R/W	Interrupt enable for AIS31 noise alarm
9	IntEnPre	R/W	Interrupt enable for AIS31 preliminary noise alarm
8	SoftReset	R/W	Software reset 0: Normal mode 1: The continuous test, the conditioning function and the FIFO are reset. This bit is not cleared automatically.
7	IntEnFull	R/W	Interrupt enable for FIFO full
6	-	-	Reserved
5	IntEnProp	R/W	Interrupt enable for Adaptive Proportion Test failure (1024-sample window)
4	IntEnRep	R/W	Interrupt enable for Repetition Count Test failure
3	CondBypass	R/W	0: the conditioning function is used (normal mode) 1: the conditioning function is bypassed (to observe entropy source directly).
2	TestEn	R/W	Select input for conditioning function and continuous tests: 0: Noise source (normal mode) 1: Test data register (test mode)
1	LFSREn	R/W	Select between the NDRNG with asynchronous free running oscillators (when '0') and the Pseudo-Random generator with synchronous oscillators for simulation purpose (when '1').
0	Enable	R/W	Enable the TRNG.

4.17.3.2. FIFOLevel register

Table 4-380 FifoLevel register

Bits	Name	Access	Description
[31:0]	FIFOLevel	R	Number of 32 bits words of random available in the FIFO. Any write to this register clears the FullInt flag in the Status register.

4.17.3.3. FIFOThresh register

Table 4-381 FIFOThresh register

Bits	Name	Access	Description

[log2fifodepth-2:0]	FIFOLevel	R	FIFO level below which the module leaves the idle state to refill the FIFO, expressed in number of 128bit blocks.

4.17.3.4. FIFODepth register

Table 4-382 FIFODepth register

Bits	Name	Access	Description
[31:0]	FIFO Depth	R	Maximum number of 32 bits words that can be stored in the FIFO: $2^{g\_fifodepth}$.

4.17.3.5. Key registers

This set of registers bits form the 128-bit AES key used for conditioning function. The first byte (MSB of 128-bit word) is at address 0x0010, the second byte at address 0x0011...

Table 4-383 Key register

Offset	Name	Access	Description
0x0010	Key0Reg	R/W	Key (MSB)
0x0014	Key1Reg	R/W	Key
0x0018	Key2Reg	R/W	Key
0x001C	Key3Reg	R/W	Key (LSB)

4.17.3.6. TestData register

This register is used to feed known data to the conditioning function or to the continuous tests. When one data is written into this register, the 32-bit are sent to those modules. Since some time is needed for processing, there is one busy flag (TestDataBusy in status register) going high as soon as one data is written, and going low when the next data can be written.

Write access to this register is ignored when TestEn bit in control register is low. Test data written through this interface are expected to be a multiple of 128 bits.

This register is write-only.

4.17.3.7. Threshold registers

The threshold registers contain the threshold values for the continuous tests.

Table 4-384 Threshold registers

Offset	Name	Access	Description
0x0024	RepThresReg	R(W)	Repetition Test Count Cut-Off value

Vangotech

0x0028	PropThresReg	R(W)	Adaptive Proportion Test (1024-sample window) Cut-Off value

4.17.3.8. Status register

Table 4-385 Control register

Bits	Name	Access	Description
[31:12]	-	-	Unused
11	FifoAccFail	R/W	Set when a FIFO data read is performed while the NDRNG is disabled AND has its FIFO empty (FIFOLevel = 0)
10	StartUpFail	R	Start-up test failure
9	AlmInt	R	AIS31 noise alarm interrupt status
8	PreInt	R/W	AIS31 preliminary noise alarm interrupt status
7	FullInt	R/W	FIFO full status
6	-	-	Reserved
5	PropFail	R/W	NIST-800-90B adaptive Proportion Test (1024-sample window) interrupt status
4	RepFail	R/W	NIST-800-90B repetition Count Test interrupt status
[3:1]	State	R	State of the control FSM 000: Reset 001: Startup 010: Idle (Rings On) 011: Idle (Rings Off) 100: Fill FIFO 101: Error 110: Unused 111: Unused
0	TestDataBusy	R	High when data written to TestData register is being processed.

4.17.3.9. InitWaitVal register

Table 4-386 InitWaitVal register

Bits	Name	Access	Description
[31:16]	-	-	Unused
[15:0]	InitWaitVal	R/W	Number of clock cycles to wait before sampling data from the noise source. All 16 bits are reset high (reset value = 0xFFFF).

4.17.3.10. DisableOsc registers

Disable oscillator rings independently, one control bit per ring. Allow test and optimization on entropy and

Vangotech

power consumption.

Table 4-387 DisableOsc0 register

Bits	Name	Access	Description
[31:0]	DisableOsc0	R/W	Disable oscillator rings #0 to #31

Table 4-388 DisableOsc1 register

Bits	Name	Access	Description
[31:0]	DisableOsc1	R/W	Disable oscillator rings #32 to #63

4.17.3.11. SwOffTmrVal register

Table 4-389 SwOffTmrVal register

Bits	Name	Access	Description
[31:16]	-	-	Unused
[15:0]	SwOffTmrVal	R/W	Number of clk cycles to wait before stopping the rings after the FIFO is full.

4.17.3.12. ClkDiv register

Table 4-390 ClkDiv register

Bits	Name	Access	Description
[31:8]	-	-	Unused
[7:0]	ClkDiv	R/W	Sample clock divider. The frequency at which the outputs of the rings are sampled is given by Fs = Fpclk/(ClkDiv + 1).

4.17.3.13. AIS31Conf registers

Configure the AIS31 online and start-up test thresholds. These registers are only accessible when g_AIS31=True. Some registers can only be written when g_AIS31Full=True.

Table 4-391 AIS31 configuration register 0

Bits	Name	Access	Description
[31:16]	OnlineThresh	R/(W)	Online threshold
[15:0]	StartUpThresh	R/(W)	Start-up test threshold

Table 4-392 AIS31 configuration register 1

Vangotech

Bits	Name	Access	Description
[31:16]	HExpectedValue	R/(W)	Expected history value
[15:0]	OnlineRepThresh	R/W	Online repeat threshold

Table 4-393 AIS31 configuration register 2

Bits	Name	Access	Description
[31:16]	HMax	R/(W)	Maximum allowed history valu
[15:0]	HMin	R/(W)	Minimum allowed history value

4.17.3.14. AIS31Status register

This register is used to obtain diagnostic information about the AIS31 start-up and online tests when g_AIS31=True. Writing to this register clears the NumPrelimAlarms value. The PrelimNoiseAlarmRep and PrelimNoiseAlarmRng flags are cleared by clearing the PreInt flag in the Status register.

Table 4-394 AIS31Status register

Bits	Name	Access	Description
[31:18]	-	-	Unused
17	PrelimNoiseAlarmRep	R	Last preliminary noise alarm occurred due to consecutive high X^2
16	PrelimNoiseAlarmRng	R	Last preliminary noise alarm occurred due to history value out of range
[15:0]	NumPrelimAlarms	R/(W)	Number of preliminary noise alarms since counter was last cleared

4.17.3.15. HwConf register

Hardware configuration parameters can be read via this register.

Table 4-395 HwConf register

Bits	Name	Access	Description
[31:10]	-	-	Unused
9	AIS31Full	R	Generic g_AIS31Full value
8	AIS31	R	Generic g_AIS31 value
[7:0]	NumbOfRings	R	Generic g_NumRings value

Vangotech

Vangotech

4.18. Crypto Engine

4.18.1. Overview

The Crypto Engine (CE) module is one encryption/decryption algorithms accelerator. It supports kinds of symmetric, asymmetric, HASH, and RNG algorithms. There are two software interfaces for secure and non-secure world. The software interface is simple for configuration, only setting interrupt control, task description address and load tag. Algorithm control information is written in memory in task descriptor, then CE automatically reads it when executing request.

4.18.2. Features

AES

- NIST FIPS 197
- Key size: 128-bit, 192-bit, and 256-bit
- Modes based on NIST SP 800-38: ECB, CTR, CBC, CFB, OFB, CMAC, CCM, GCM

• HASH

- FIPS 180-3: MD-5, SHA-1, SHA-256, SHA-224, SHA-384 and SHA-512
- 66 cycles per 512-bits blocks for MD5, SHA-224, and SHA-256
- 82 cycles per 512/1024-bits blocks for SHA-1, SHA-384, and SHA-512

4.18.3. Block Diagram

The following figure shows the block diagram of Crypto Engine.

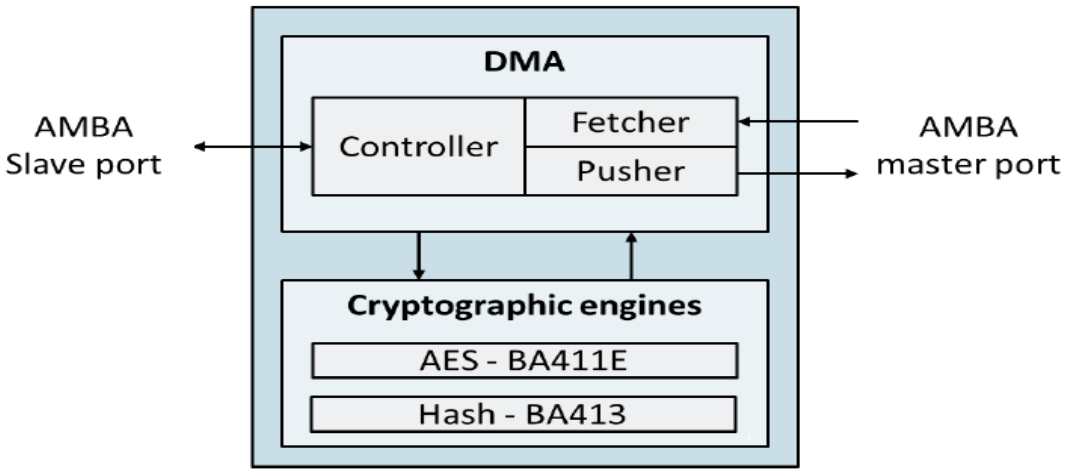


Figure 4-76. CE Block Diagram

4.18.4. Functional Descriptions

4.18.4.1. Control bits

4.1.1.1 Constant address bit

When the constant address bit is set, the address is not incremented. This allows accessing a memory mapped FIFO for example. All transfers (including the first one and the last one) are done using full bus width (no narrow transfer):

- The address must be aligned on bus width.

- If the transfer length is not a multiple of bus width, the DMA fetcher will read more bytes than necessary (those bytes will be discarded).
- If the transfer length is not a multiple of bus width, the DMA pusher will write more bytes than necessary (the value of those additional bytes is undefined).

4.1.1.2 Discard bit

When the discard bit is set, the data coming from the IP cores is discarded by the pusher instead of being written to memory. In this case the destination address is ignored.

4.1.1.3 Realign bit

The realign bit is used to realign the data stream on FIFO width boundary.

Assuming a 32-bit FIFO (c_DataWidth = 32):

- If the total number of bytes transferred (since previous descriptor with realign=1) is a multiple of 4 (32-bit), then this bit has no effect.
- If it is not a multiple of 4, then the effect is the following:
  - The fetcher will add 1, 2 or 3 bytes to the stream (sent to the IP cores) to reach 32-bit boundary.
  - The pusher will discard 1, 2 or 3 bytes from the stream (coming from the IP core) to reach 32-bit boundary.

This bit is usually set in the last descriptor for one item (config, key, payload…) to be transferred.

Note that it's not mandatory to use the realign bit: the same effect could be achieved by other means:

- with an additional descriptor with the required number dummy bytes for the fetcher;
- with an additional descriptor with discard bit for the pusher.

The above description assumes scatter-gather mode, but the principle remains valid in direct mode.

4.18.4.2. Using the DMA

The fetcher and the pusher can be controlled independently. The sequence to use the pusher or the fetcher is the following:

- Write all the configuration registers.
- Write the start bit in the control register.
- Wait for transfer completion:
  - Either by using interrupt
  - Or by polling busy bit in the status register.
- Clear the interrupt (when using interrupt).

A new transfer can be started when the previous one has been completed. The above flow is valid for both direct and scatter-gather modes.

The usual way to use the DMA is to program the fetcher with the operation inputs and to program the pusher with the corresponding outputs. In this case the software needs only to wait the pusher transfer completion since the fetcher will always finish before the pusher in that case.

4.18.5. Register List

Module Name	Base Address
CE	0x2830_0000

Vangotech

Register Name	Offset	Description
Fetcher / pusher registers		
FETCH_ADDR_LSB	0x00	Fetch Address
FETCH_ADDR_MSB	0x04	Fetch Address
FETCH_LEN	0x08	Fetch length
FETCH_TAG	0x0C	User tag
PUSH_ADDR_LSB	0x10	Push address
PUSH_ADDR_MSB	0x14	Push address
PUSH_LEN	0x18	Push length
Interrupts		
INT_EN	0x1C	Interrupt enable
INT_ENSET	0x20	Interrupt enable set
INT_ENCLR	0x24	Interrupt enable clear
INT_STATRAW	0x28	Interrupt raw status
INT_STAT	0x2C	Interrupt status
INT_STATCLR	0x30	Interrupt status clear
Control / status register		
CONFIG	0x34	Fetch/Push config
START	0x38	Fetch/Push start
STATUS	0x3C	Fetch/Push status

4.18.6. Register Description

4.18.6.1. Fetcher / pusher registers

Table 4-396. Fetcher / pusher registers

Register	Bits	Access	Description
FETCH_ADDR_LSB	31:0	R/W	Address Direct mode: This register is written by the software (address of the first data). Scatter-gather mode: This register is written by software (address of the first descriptor). Afterwards, it is updated by the hardware after each processed descriptor. 32-bit address The ADDR_LSB register and ADDR_MSB register both give access to the same 32-bit register. 64-bit address One register gives access to the 32
PUSH_ADDR_LSB			
FETCH_ADDR_MSB	31:0	R/W	
PUSH_ADDR_MSB			

Vangotech

			most significant bits and one register gives access to the 32 least significant bits.
FETCH_LEN PUSH_LEN	27:0	*R/W*	Length of data block In direct mode, this register is written by the software. In scatter-gather mode, this register is not used.
	28	*R/W*	Constant address In direct mode, this register is written by the software. In scatter-gather mode, this register is not used.
	29	*R/W*	Realign length In direct mode, this register is written by the software. In scatter-gather mode, this register is not used.
	30	*R/W*	Discard data In direct mode, this register is written by the software. In scatter-gather mode, this register is not used.
FETCH_TAG	31:0	*R/W*	User tag In direct mode, this register is written by the software. In scatter-gather mode, this register is not used.

4.18.6.2. Interrupts

The table below shows all interrupts that can be generated by the DMA module.

Table 4-397. Interrupt sources

#	Module	Name	Description
0	AXI2FIFO (fetcher)	End of block	Triggered at the end of each block (if enabled in the descriptor - scatter-gather only)
1		Stopped	Triggered when reaching a block with Stop=1 (or end of direct transfer)
2		Error	Triggered when an error response is received from AXI
3	FIFO2AXI (pusher)	End of block	Triggered at the end of each block (if enabled in the descriptor -

Vangotech

			scatter-gather only)
4		Stopped	Triggered when reaching a block with Stop=1 (or end of direct transfer)
5		Error	Triggered when an error response is received from AXI

The table below describes the registers controlling the interrupts. Each register has one bit per interrupt.

Table 4-398. Interrupt registers

Register	Bits	Access	Description
INT_EN	5:0	R/W	Interrupt enable
INT_ENSET	5:0	W	Interrupt enable set This register allows modifying only a few bits of INT_EN in a single write access. Writing a '1' enables the interrupt. Writing a '0' has no effect.
INT_ENCLR	5:0	W	Interrupt enable clear This register allows modifying only a few bits of INT_EN in a single write access. Writing a '1' disables the interrupt. Writing a '0' has no effect.
INT_STATRAW	5:0	R	Interrupt raw status Reading returns the raw status (independently of interrupt enable).
INT_STAT	5:0	R	Interrupt status Reading returns the interrupt status (depending on interrupt enable).
INT_STATCLR	5:0	W	Interrupt status clear Writing a '1' clears the interrupt status. Writing a '0' has no effect.

The module generates an active-high interrupt signal.

- When an interrupt condition occurs, the corresponding bit in INT_STATRAW is set.
- This bit can be cleared by software (INT_STATCLR).
- The INT_STAT register contains a bitwise AND of INT_STATRAW and INT_EN.
- If INT_STAT is different than zero, then the interrupt signal (to the interrupt controller of the system) is high.

As long as INT_STAT[0] (End of block) is high or INT_STAT[1] (stopped) is high or INT_STAT[2] (Error) is high, the registers FETCH_ADDR, FETCH_DESCR, FETCH_LEN and FETCH_TAG will stay stable and contain the information related to the descriptor that triggered the interrupt. This allows the interrupt routine to read stable value. When those interrupt bits are low (cleared or disabled), those registers are updated after each processed block. The mechanism is the same for the pusher.

If the software does not handle the interrupt fast enough, there is a back-pressure mechanism that will

stall the DMA, so that not interrupt will be lost.

4.18.6.3. Control / status register

The table below describes the control/status registers.

Table 4-399. Control/status register

Register	Bits	Access	Description
CONFIG	0	R/W	Fetcher scatter/gather When this bit is zero, the fetcher runs in direct mode. When this bit is one, the fetcher runs in scatter-gather mode.
	1	R/W	Pusher scatter/gather When this bit is zero, the pusher runs in direct mode. When this bit is one, the pusher runs in scatter-gather mode.
	2	R/W	Stop fetcher When this bit is high, the fetcher will stop at the end of the current block (even if the STOP bit in the descriptor is low).
	3	R/W	Stop pusher When this bit is high, the pusher will stop at the end of the current block (even if the STOP bit in the descriptor is low).
START	0	W	Start fetch Writing a '1' starts the fetcher (AXI2FIFO). Writing a '0' has no effect.
	1	W	Start push Writing a '1' starts the pusher (FIFO2AXI). Writing a '0' has no effect.
STATUS	0	R	Fetcher busy This bit is high as long as the fetcher is busy.
	1	R	Pusher busy This bit is high as long as the pusher is busy.
	2	R	Reserved
	3	R	Reserved
	4	R	Not empty flag from input FIFO (fetcher)
	5	R	Pusher Waiting FIFO This bit is high when the pusher if waiting for

Vangotech

			more data in output FIFO.
	6	R	SoftRst_busy This bit is high when the soft reset is "on going"
	15:7	-	Not used
	31:16	R	Number of data in output FIFO (pusher)

4.18.6.4. Scatter-gather descriptor

With 32-bit address, each descriptor entry is made of three (FIFO2AXI) or four (AXI2FIFO) 32-bit words, as described in the table below.

Table 4-400. Descriptor (32-bit address)

Offset	Bits	Name	Description
0x00	31:0	Address LSB	Address of first data of the block
0x04	0	Stop	Flag indicating that this is the last descriptor or that the next descriptor is not ready yet.
	1	-	Not used
	31:2	Next LSB	Address of next descriptor Since the address must be aligned on 32-bit, the two LSB are not used.
0x08	27:0	Length	Number of bytes inside the block
	28	CstAddr	When high, address is not incremented (to access a FIFO)
	29	Realign	When high, block length will be realigned on FIFO width).
	30	Discard	When high, data is not written to memory (pusher only).
	31	IntEn	Interrupt enable: when high, an interrupt is generated at the end of the block.
0x0C	31:0	Tag	User tag (only for AXI2FIFO)

With 64-bit address, the descriptor size is 5 (FIFO2AXI) or 6 (AXI2FIFO) words.

Table 4-401. Descriptor (64-bit address)

Offset	Bits	Name	Description
0x00	31:0	Address LSB	Address of first data of the block
0x04	31:0	Address MSB	
0x08	0	Stop	Flag indicating that this is the last descriptor or that the next descriptor is not ready yet.
	1	-	Not used
	31:2	Next LSB	Address of next descriptor

Vangotech

0x0C	31:0	Next MSB	Since the address must be aligned on 32-bit, the two LSB are not used.
0x10	27:0	Length	Number of bytes inside the block
	28	CstAddr	When high, address is not incremented (to access a FIFO)
	29	Realign	When high, block length will be realigned on FIFO width.
	30	Discard	When high, data is not written to memory (pusher only).
	31	IntEn	Interrupt enable: when high, an interrupt is generated at the end of the block.
0x14	31:0	Tag	User tag (only for AXI2FIFO)

4.19. Secure Design

The following subsections describe the secure design in SCM701. There are three special secure designs in V8130L: JTAG, SPI Host, and Efuse Controller.

4.19.1. Secure JTAG

Figure 7-5 describes V8130L's JTAG secure scheme:

1) eFuse bit [67] enable/disable JTAG function

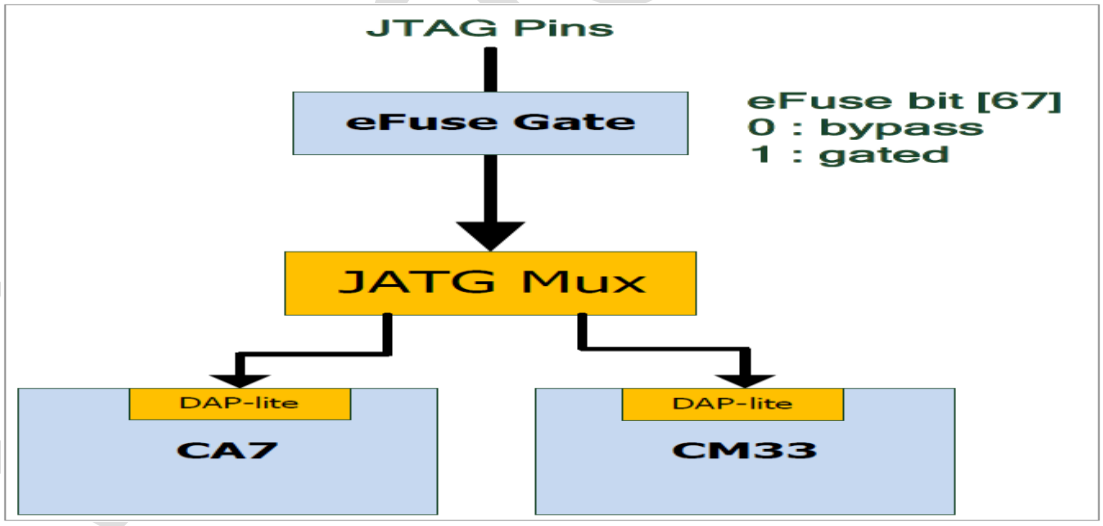


Figure 4-77. Secure JATG Scheme

4.19.2. Secure SPI Host

Figure 4-78 describes V8130L's SPI Host secure scheme:

1) eFuse bit [64] enable/disable SPI Host function

2) Software Password to enable/disable SPI Host function if eFuse bit [64] is not gated (disable) SPI pins.

3) SPI Host's AHB interface always acts as secure master which can access all slaves.

4) After power-on, the access path of secure SPI is workable. Users need to modify the password to

disable.

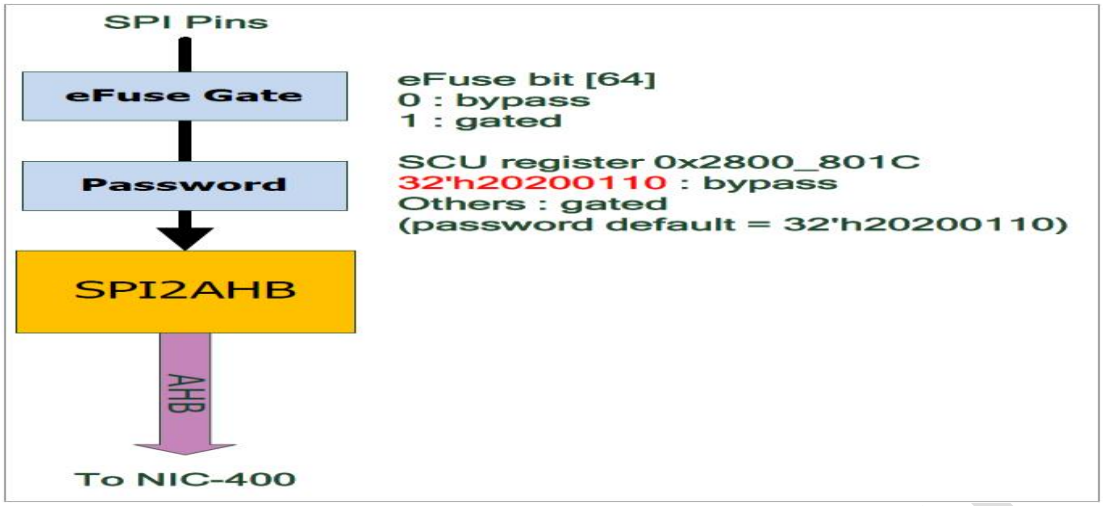


Figure 4-78. Secure SPI Host Scheme

4.19.3. Secure Efuse Controller

Figure 4-79 describes V8130L's Efuse Controller secure scheme:

1) Software Password to enable/disable whether Efuse Controller can read correct data or not. If the password is not match matched , the read data will all 0'

2) If user users would like to program eFuse's content through Efuse Controller, it need needs to unlock the write

protect function. Please refer eFuse Controller_v7 .pdf for detail detailed eFuse programming flow.

3) After power power-on, the access path of efuse is workable. User Users need to modify the password to disable.

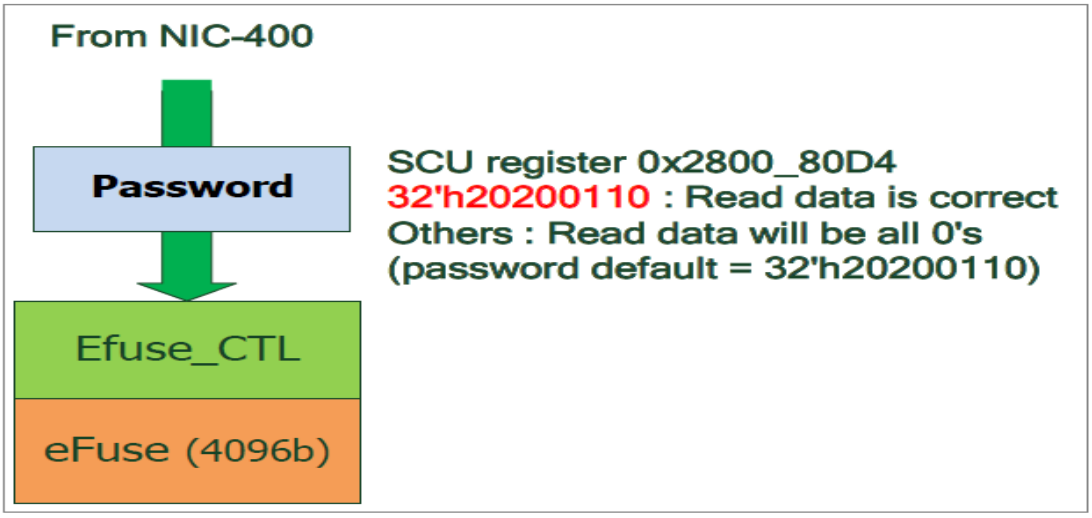


Figure 4-79. Secure Efuse Controller Scheme

4.19.4. Application Note

• For V8130L system application, users need to confirm whether the external input pad is driven on low when V8130L is power-off. If the external pad is not driven on low, there may be leakage current on system.

• Partial slave are defined as boot secured attribute, and users need to configure the GPV 0/1 to modify by application. (Secure or Non-Secure attribute)

• System will boot up by the internal OSC and enable PLL0/1/2/4/6 by default. Users can disable them by application. When fast boot from the internal OSC and change system into PLL clock source, there are some notices about switching mechanism:

- PLL0/PLL4 controlled by FTSCU100 unit
- The frequency change of CPU/SYSTEM BUS CLK is needed by FCS mechanism.
- "scu_clkin_mux" are controlled by FCS and execute when FCS is complete. (These mux are all system bus clock and well controlled by SCU.)
- The suggestion flow of switching clock is mux select register setting after enabling the clock source.
- When any clock mux switching in clock schematic, mux input clocks need exist on system. After clock switching completion, it is permitted to gate clock.

• If it is necessary to reset NIC bus, please follow the rule:

- All master and slave on NIC bus need to be reset at the same time, and they cannot be reset independently.

Bit	Symbol	Access	Default	Description
18	cm33bus_rstn	RW	0x0	CM33 bus reset
17	nic_0_bus_rstn	RW	0x0	NIC_0 bus reset
16	nic_1_bus_rstn	RW	0x0	NIC_1 bus reset

• When users need to enter SoftOff/Sleep mode and wait for external event trigger

- Before entering SoftOff/Sleep mode, users need to configure 0x2800_0000[2] as "0" to change RTC pads into the input mode and wait for external trigger.

2	GPO[2]	R/W	0x1	WAKEUP_IN_ENABLE: (Users need to enable before entering the wake-up mode.) 1: Disable the wakeup by external pad 0: Enable the wakeup by external pad

4.19.5. TZC NSAID Mapping Table

For TZC400 security mechanism, the filter can be passed by setting NSAID for specified master. The mapping table of Master is listed as below table for reference.

Master	Mapping NSAID for TZC
DMA_0	1
DMA_1	2
OTG	3
OTG_1	4
OTG_2	5
MAC	6
MAC_1	7
MAC_2	8
NIC_0_TO_1	9
eMMC_0	10
eMMC_1	11
AES	12
SPI2AHB	13

4.20. Internal Interrupt Vector Generation

The complete list of interrupt sources and the interrupt controller inputs to which they are assigned is shown in Table 4-402.

Table 4-402. Internal Interrupt Vectors for CA7

IRQ Source	IRQ Source Name	IRQ Controller Instance	IRQ Controller Pin Number	Inv?	IRQ #	IRQ Trigger Mode	IRQ Trigger Level
u_SYSC	scu_intr	u_Glue_sync_ca7	IRQS[0]	-	IRQ 0	-	-
u_FTGPIO010	gpio_intr	u_Glue_sync_ca7	IRQS[1]	-	IRQ 1	-	-
u_FTGPIO010_u1	gpio_intr	u_Glue_sync_ca7	IRQS[2]	-	IRQ 2	-	-
u_FTWDT011	wd_intr	u_Glue_sync_ca7	IRQS[3]	-	IRQ 3	-	-
u_FTWDT011_u1	wd_intr	u_Glue_sync_ca7	IRQS[4]	-	IRQ 4	-	-
u_FTWDT011_u2	wd_intr	u_Glue_sync_ca7	IRQS[5]	-	IRQ 5	-	-
u_FTWDT011_u3	wd_intr	u_Glue_sync_ca7	IRQS[6]	-	IRQ 6	-	-
u_FTGMAC030	mac_intr	u_Glue_sync_ca7	IRQS[7]	-	IRQ 7	-	-

u_FTGMAC030_u1	mac_intr	u_Glue_sync_ca7	IRQS[8]	-	IRQ 8	-	-
u_FTGMAC030_u2	mac_intr	u_Glue_sync_ca7	IRQS[9]	-	IRQ 9	-	-
u_FTSSP010_I2S	ssp_intr	u_Glue_sync_ca7	IRQS[10]	-	IRQ 10	-	-
u_FTSSP010_I2S_u1	ssp_intr	u_Glue_sync_ca7	IRQS[11]	-	IRQ 11	-	-
u_FTSSP010_SPI	ssp_intr	u_Glue_sync_ca7	IRQS[12]	-	IRQ 12	-	-
u_FTSSP010_SPI_u1	ssp_intr	u_Glue_sync_ca7	IRQS[13]	-	IRQ 13	-	-
u_FTSSP010_SPI_u2	ssp_intr	u_Glue_sync_ca7	IRQS[14]	-	IRQ 14	-	-
u_FTSSP010_SPI_u3	ssp_intr	u_Glue_sync_ca7	IRQS[15]	-	IRQ 15	-	-
u_FTSSP010_SPI_u4	ssp_intr	u_Glue_sync_ca7	IRQS[16]	-	IRQ 16	-	-
u_FTSSP010_SPI_u5	ssp_intr	u_Glue_sync_ca7	IRQS[17]	-	IRQ 17	-	-
u_FTSSP010_SPI_u6	ssp_intr	u_Glue_sync_ca7	IRQS[18]	-	IRQ 18	-	-
u_FTSSP010_SPI_u7	ssp_intr	u_Glue_sync_ca7	IRQS[19]	-	IRQ 19	-	-
u_FTPWMTMR010	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[20]	-	IRQ 20	-	-
u_FTPWMTMR010	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[21]	-	IRQ 21	-	-
u_FTPWMTMR010	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[22]	-	IRQ 22	-	-
u_FTPWMTMR010	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[23]	-	IRQ 23	-	-
u_FTPWMTMR010	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[24]	-	IRQ 24	-	-
u_FTPWMTMR010_u1	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[25]		IRQ 25	-	-
u_FTPWMTMR010_u1	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[26]	-	IRQ 26	-	-
u_FTPWMTMR010_u1	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[27]	-	IRQ 27	-	-
u_FTPWMTMR010	ptmr_tmr	u_Glue_sync_ca	IRQS[28]	-	IRQ	-	-

Vangotech

_u1	3_intr	7			28		
u_FTPWMTMR010_u1	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[29]	-	IRQ 29	-	-
u_FTPWMTMR010_u2	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[30]	-	IRQ 30	-	-
u_FTPWMTMR010_u2	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[31]	-	IRQ 31	-	-
u_FTPWMTMR010_u2	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[32]	-	IRQ 32	-	-
u_FTPWMTMR010_u2	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[33]	-	IRQ 33	-	-
u_FTPWMTMR010_u2	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[34]	-	IRQ 34	-	-
u_FTPWMTMR010_u3	ptmr_tmr_intr	u_Glue_sync_ca7	IRQS[35]	-	IRQ 35	-	-
u_FTPWMTMR010_u3	ptmr_tmr1_intr	u_Glue_sync_ca7	IRQS[36]	-	IRQ 36	-	-
u_FTPWMTMR010_u3	ptmr_tmr2_intr	u_Glue_sync_ca7	IRQS[37]	-	IRQ 37	-	-
u_FTPWMTMR010_u3	ptmr_tmr3_intr	u_Glue_sync_ca7	IRQS[38]	-	IRQ 38	-	-
u_FTPWMTMR010_u3	ptmr_tmr4_intr	u_Glue_sync_ca7	IRQS[39]	-	IRQ 39	-	-
u_FTIIC010	isi2c	u_Glue_sync_ca7	IRQS[40]	-	IRQ 40	-	-
u_FTIIC010_u1	isi2c	u_Glue_sync_ca7	IRQS[41]	-	IRQ 41	-	-
u_FTIIC010_u2	isi2c	u_Glue_sync_ca7	IRQS[42]	-	IRQ 42	-	-
u_FTIIC010_u3	isi2c	u_Glue_sync_ca7	IRQS[43]	-	IRQ 43	-	-
u_FTIIC010_u4	isi2c	u_Glue_sync_ca7	IRQS[44]	-	IRQ 44	-	-
u_FTUART010	irda_intr1	u_Glue_sync_ca7	IRQS[45]	-	IRQ 45	-	-
u_FTUART010	irda_intr2	u_Glue_sync_ca7	IRQS[46]	-	IRQ 46	-	-
u_FTUART010_u1	irda_intr1	u_Glue_sync_ca7	IRQS[47]	-	IRQ 47	-	-
u_FTUART010_u1	irda_intr2	u_Glue_sync_ca7	IRQS[48]	-	IRQ 48	-	-

Vangotech

u_FTUART010_u2	irda_intr 1	u_Glue_sync_ca 7	IRQS[49]	-	IRQ 49	-	-
u_FTUART010_u2	irda_intr 2	u_Glue_sync_ca 7	IRQS[50]	-	IRQ 50	-	-
u_FTUART010_u3	irda_intr 1	u_Glue_sync_ca 7	IRQS[51]	-	IRQ 51	-	-
u_FTUART010_u3	irda_intr 2	u_Glue_sync_ca 7	IRQS[52]	-	IRQ 52	-	-
u_FTUART010_u4	irda_intr 1	u_Glue_sync_ca 7	IRQS[53]	-	IRQ 53	-	-
u_FTUART010_u4	irda_intr 2	u_Glue_sync_ca 7	IRQS[54]	-	IRQ 54	-	-
u_FTUART010_u5	irda_intr 1	u_Glue_sync_ca 7	IRQS[55]	-	IRQ 55	-	-
u_FTUART010_u5	irda_intr 2	u_Glue_sync_ca 7	IRQS[56]	-	IRQ 56	-	-
u_FTUART010_u6	irda_intr 1	u_Glue_sync_ca 7	IRQS[57]	-	IRQ 57	-	-
u_FTUART010_u6	irda_intr 2	u_Glue_sync_ca 7	IRQS[58]	-	IRQ 58	-	-
u_FTUART010_u7	irda_intr 1	u_Glue_sync_ca 7	IRQS[59]	-	IRQ 59	-	-
u_FTUART010_u7	irda_intr 2	u_Glue_sync_ca 7	IRQS[60]	-	IRQ 60	-	-
u_FTUART010_u8	irda_intr 1	u_Glue_sync_ca 7	IRQS[61]	-	IRQ 61	-	-
u_FTUART010_u8	irda_intr 2	u_Glue_sync_ca 7	IRQS[62]	-	IRQ 62	-	-
u_FTUART010_u9	irda_intr 1	u_Glue_sync_ca 7	IRQS[63]	-	IRQ 63	-	-
u_FTUART010_u9	irda_intr 2	u_Glue_sync_ca 7	IRQS[64]	-	IRQ 64	-	-
u_FTDDR3030	ddr_int	u_Glue_sync_ca 7	IRQS[65]	-	IRQ 65	-	-
u_FTDDR3030	ecc_int	u_Glue_sync_ca 7	IRQS[66]	-	IRQ 66	-	-
u_FTSDC021_EM MC	sdc_int	u_Glue_sync_ca 7	IRQS[67]	-	IRQ 67	-	-
u_FTSDC021_EM MC_u1	sdc_int	u_Glue_sync_ca 7	IRQS[68]	-	IRQ 68	-	-
u_FTSPI020	intr	u_Glue_sync_ca	IRQS[69]	-	IRQ	-	-

Vangotech

		7			69		
u_FTCAN010_u0	ir_n	u_Glue_sync_ca 7	IRQS[70]	Y	IRQ 70	-	-
u_FTCAN010_u1	ir_n	u_Glue_sync_ca 7	IRQS[71]	Y	IRQ 71	-	-
u_FTKBC010	kb_int	u_Glue_sync_ca 7	IRQS[72]	-	IRQ 72	-	-
u_FTKBC010	kb_padin t	u_Glue_sync_ca 7	IRQS[73]	-	IRQ 73	-	-
-	-	u_Glue_sync_ca 7	IRQS[74]	-	IRQ 74	-	-
-	-	u_Glue_sync_ca 7	IRQS[75]	-	IRQ 75	-	-
u_FTADCC010	adc_int	u_Glue_sync_ca 7	IRQS[76]	-	IRQ 76	-	-
u_FTTDCC010	adc_int	u_Glue_sync_ca 7	IRQS[77]	-	IRQ 77	-	-
-	-	u_Glue_sync_ca 7	IRQS[78]	-	IRQ 78	-	-
-	-	u_Glue_sync_ca 7	IRQS[79]	-	IRQ 79	-	-
-	-	u_Glue_sync_ca 7	IRQS[80]	-	IRQ 80	-	-
u_FOTG210	sys_int_n	u_Glue_sync_ca 7	IRQS[81]	Y	IRQ 81	-	-
u_FOTG210_u1	sys_int_n	u_Glue_sync_ca 7	IRQS[82]	Y	IRQ 82	-	-
u_FOTG210_u2	sys_int_n	u_Glue_sync_ca 7	IRQS[83]	Y	IRQ 83	-	-
u_FTLCDC210	LC_INTR	u_Glue_sync_ca 7	IRQS[84]	-	IRQ 84	-	-
u_FTLCDC210	LC_MERR INTR	u_Glue_sync_ca 7	IRQS[85]	-	IRQ 85	-	-
u_FTLCDC210	LC_FURI NTR	u_Glue_sync_ca 7	IRQS[86]	-	IRQ 86	-	-
u_FTLCDC210	LC_BAUP DINTR	u_Glue_sync_ca 7	IRQS[87]	-	IRQ 87	-	-
u_FTLCDC210	LC_VSTA TUSINTR	u_Glue_sync_ca 7	IRQS[88]	-	IRQ 88	-	-
u_FTEMC030_RO M	intr	u_Glue_sync_ca 7	IRQS[89]	-	IRQ 89	-	-

Vangotech

u_FTEMC030_SRAM	intr	u_Glue_sync_ca7	IRQS[90]	-	IRQ 90	-	-
u_TZC400_1F_ca7_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[91]	-	IRQ 91	-	-
u_TZC400_1F_lcdc_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[92]	-	IRQ 92	-	-
u_TZC400_2F_bus_ddr	TZCINT	u_Glue_sync_ca7	IRQS[93]	-	IRQ 93	-	-
u_ba431	IRQ_TRNG	u_Glue_sync_ca7	IRQS[94]	-	IRQ 94	-	-
u_cryptomaster_ahb	IRQ	u_Glue_sync_ca7	IRQS[95]	-	IRQ 95	-	-
u_CA7Q	COMMRX[0]	u_Glue_sync_ca7	IRQS[96]	-	IRQ 96	-	-
u_CA7Q	COMMRX[1]	u_Glue_sync_ca7	IRQS[97]	-	IRQ 97	-	-
u_CA7Q	COMMRX[2]	u_Glue_sync_ca7	IRQS[98]	-	IRQ 98	-	-
u_CA7Q	COMMRX[3]	u_Glue_sync_ca7	IRQS[99]	-	IRQ 99	-	-
u_CA7Q	COMMTX[0]	u_Glue_sync_ca7	IRQS[100]	Y	IRQ 100	-	-
u_CA7Q	COMMTX[1]	u_Glue_sync_ca7	IRQS[101]	Y	IRQ 101	-	-
u_CA7Q	COMMTX[2]	u_Glue_sync_ca7	IRQS[102]	Y	IRQ 102	-	-
u_CA7Q	COMMTX[3]	u_Glue_sync_ca7	IRQS[103]	Y	IRQ 103	-	-
u_CA7Q	nAXIERRIRQ	u_Glue_sync_ca7	IRQS[104]	Y	IRQ 104	-	-
u_CA7Q	nCTIIRQ[0]	u_Glue_sync_ca7	IRQS[105]	Y	IRQ 105	-	-
u_CA7Q	nCTIIRQ[1]	u_Glue_sync_ca7	IRQS[106]	Y	IRQ 106	-	-
u_CA7Q	nCTIIRQ[2]	u_Glue_sync_ca7	IRQS[107]	Y	IRQ 107	-	-
u_CA7Q	nCTIIRQ[3]	u_Glue_sync_ca7	IRQS[108]	Y	IRQ 108	-	-
u_CA7Q	nPMUIRQ[0]	u_Glue_sync_ca7	IRQS[109]	Y	IRQ 109	-	-
u_CA7Q	nPMUIRQ	u_Glue_sync_ca	IRQS[110]	Y	IRQ	-	-

Vangotech

	[1]	7			110		
u_CA7Q	nPMUIRQ [2]	u_Glue_sync_ca 7	IRQS[111]	Y	IRQ 111	-	-
u_CA7Q	nPMUIRQ [3]	u_Glue_sync_ca 7	IRQS[112]	Y	IRQ 112	-	-
u_FTDMAC020	dmaint	u_Glue_sync_ca 7	IRQS[113]	-	IRQ 113	-	-
u_FTDMAC020	dmaint_t c	u_Glue_sync_ca 7	IRQS[114]	-	IRQ 114	-	-
u_FTDMAC020	dmaint_e rr	u_Glue_sync_ca 7	IRQS[115]	-	IRQ 115	-	-
u_FTDMAC030	DMACIntr	u_Glue_sync_ca 7	IRQS[116]	-	IRQ 116	-	-
u_FTDMAC030	DMACIntr _err	u_Glue_sync_ca 7	IRQS[117]	-	IRQ 117	-	-
u_FTDMAC030	DMACIntr _tc	u_Glue_sync_ca 7	IRQS[118]	-	IRQ 118	-	-
u_FTDMAC030_u 1	DMACIntr	u_Glue_sync_ca 7	IRQS[119]	-	IRQ 119	-	-
u_FTDMAC030_u 1	DMACIntr _err	u_Glue_sync_ca 7	IRQS[120]	-	IRQ 120	-	-
u_FTDMAC030_u 1	DMACIntr _tc	u_Glue_sync_ca 7	IRQS[121]	-	IRQ 121	-	-

Vangotech

Chapter 5. Memory

This chapter details the V8130L external memory.

•

Vangotech

EMC

- eMMC

Vangotech

5.1. EMC

5.1.1. Overview

EMC is an embedded synchronous single-port SRAM/ROM controller designed to directly access the AXI slave port. The AXI slave port does not support the Exclusive and Locked atomic accesses. When the master issues the Exclusive/Locked read command to EMC, the response of the read operation indicated as "OKAY" means that the AXI slave port does not support the Exclusive access. Therefore, the AXI master should not issue the Exclusive/Locked write transfer. Once the master issues the Exclusive/Locked write transfer, EMC will treat as "Normal write transfer" and overwrite the data of the target address location. Besides, EMC can be configured up to two memory banks for reducing the memory read latency for the read access timing larger than one AXI clock. EMC includes the hamming code ECC function with few ECC bases. EMC can correct 1-bit error, but cannot correct 2-bit or more bit error. EMC will response 2-bit or more bit error when it occurs.

5.1.2. Features

- Two instances, EMC0 for ROM, EMC1 for SRAM.
- Supports 32-bit address bus width
- Supports AXI 64-bit data bus widths
- Supports 64-bit hamming code ECC base for EMC1, not for EMC0
- Supports 19 ID width for AXI channel
- Supports one memory bank for EMC0, two memory banks for EMC1
- Supports one AXI clock latency for EMC0, two AXI clock latency for EMC1
- Automatically enters retention mode
- Programmable SRAM retention function and wake-up cycle
- Supports three lever synchronous single-port SRAM/ROM

5.1.3. Block Diagram

Figure 5-1 shows the functional block diagram of the EMC.

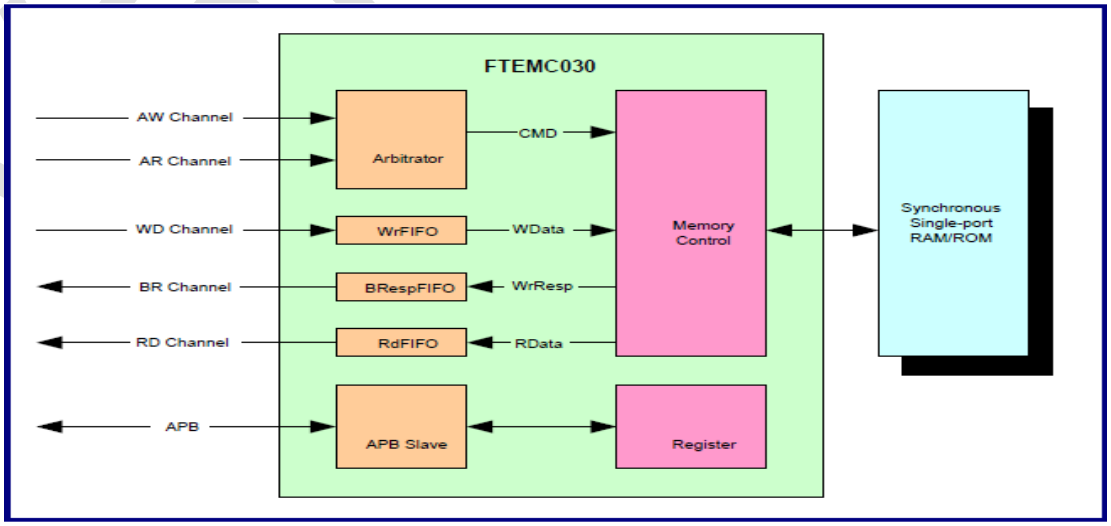


Figure 5-1. Functional Block Diagram

5.1.4. Operations and Functional Descriptions

5.1.4.1. Read Un-pack

The main function of the read un-pack is to handle the AXI read data flow. When AXI issues a read command to EMC, it will calculate the total read memory counts, and then read memory data times depending on the count value. The read data will be placed in the read data FIFO. When the read data FIFO is not empty, EMC will pop the data times to the AXI read data channel depending on the AXI bursts needed.

5.1.4.2. Memory Retention Function Control

The default of the memory retention function is off. Users can turn it on by setting bit[2:1] of register offset 0x04. When the memory retention function is on, EMC will automatically assert the mRET pin to control the memory to enter the retention mode when EMC is idle.

5.1.5. Register List

Module Name	Base Address
EMC_0 (for SRAM) CSR	0x28A0 0000
EMC_1 (for ROM) CSR	0X28B0 0000

Table 5-1. Register Summary Table

Base Offset (Hex)	Name	Type	Description
00	EEA	RO	ECC Error Address
04	MCR	RW	Memory Control Register
08	PCR	RW	APB Control Register
F8	FEAR	RO	Feature Register
FC	IPR	RO	Version register

5.1.6. Register Definition

5.1.6.1. ECC Error Address Register

This register will report the address if uncorrectable error during read data occurs. The address is aligned with the AXI bus data width (Please note that if EMC_INTERLEAVE_2 is configured, the address is aligned with 2 times of the AXI bus data width). This register is valid if EMC_ECC_ON is configured.

Table 5-2. ECC Error Address Register (EEAR, Offset = 0x0)

Bit	Name	Type	Description	Default Value
[31:0]	err_addr	RO	ECC Error Address Location This register provides the ECC error location, which is aligned with the AXI bus data width and interleaving mode. err_addr = Start address / (EMC_DATA_WIDTH/8) / EMC_INTERLEAVE EMC_DATA_WIDTH and EMC_INTERLEAVE can be read	0

			from register offset of 0xF8. Please note that this bit is only valid when EMC_ECC_ON is configured.	

5.1.6.2. Memory Control Register

This register includes the read latency, retention wake-up cycle, and ECC relation setting.

Table 5-3. Memory Control Register (MCR, Offset = 0x4)

Bit	Name	Type	Description	Default Value
31	-	Rsvd	Reserved	0
30	err_int_en	RW	ECC Error Interrupt Enable Bit When this bit set to 1, EMC will trigger the interrupt while read data has 2-bit or more bit error that is uncorrectable. Please note that this bit is only valid when EMC_ECC_ON is configured.	0
29	bus_err_en	RW	Bus Error Response Enable Bit When this bit set to 1, AXI will response Slverr while read data has 2-bit or more bit error that is uncorrectable. Please note that if EMC_ECC_ON is not configured, this bit is invalid and RResp is always OK.	1
28	ecc_err_sts	RW1C	ECC Error Status while Reading Data 0: No error or 1-bit error, correctable 1: 2 or more bit error, cannot be corrected Please note that this bit is only valid when EMC_ECC_ON is configured.	0
[27:24]	Init_range	RW	Initial Range Initial range that users desire to initialize from the initial entry 2^{Init_size} Kbytes 0: 1 Kbytes 1: 2 Kbytes 2: 4 Kbytes ... 7: 128 Kbytes Please note that this bit is only valid when EMC_ECC_ON is configured, and users cannot set Init_range larger than the memory size. The memory size of EMC0 is 32 Kbytes, the memory size of EMC1 is 128 Kbytes	5 for EMC0 7 for EMC1
23	Init_en	RW	Initializing SRAM Content Enable Bit The content of SRAM will be filled with 0 based on	0

Vangotech

			the initial range and from initial entry when this bit is set. This bit will automatically clear to 0 when initialization is finished. Please note that this bit is only valid when EMC_ECC_ON is configured.	
[22:4]	Init_entry	RW	Initial Entry Initial entry that is calculated by start address aligned with the AXI: bus data width and interleaving mode. The calculation formula is shown below Init_entry = Start address/(EMC_DATA_WIDTH/8)/ EMC_INTERLEAVE EMC_DATA_WIDTH and EMC_INTERLEAVE can be read from register offset of 0xF8. Examples: • When start address is 0x8, EMC_DATA_WIDTH is 64, EMC_INTERLEAVE is 1, initial entry is 1. The exact initial address is from 0x8. • When start address is 0x8, EMC_DATA_WIDTH is 64, EMC_INTERLEAVE is 2, initial entry is 0. The exact initial address is from 0x0. • When start address is 0x30, EMC_DATA_WIDTH is 128, EMC_INTERLEAVE is 1, initial entry is 3. The exact initial address is from 0x30. • When start address is 0x30, EMC_DATA_WIDTH is 128, EMC_INTERLEAVE is 2, initial entry is 1. The exact initial address is from 0x20. Please note that this bit is only valid when EMC_ECC_ON is configured.	0
[3:1]	RET_CYC	RW	Wake-up cycle for the retention mode 3’b000: Disable retention function 3’b001 ~ 3'b111: Wake-up cycles 1 ~ 7	0
0	RdLatency	RW	SRAM Taa latency. 0: Taa is smaller than one clock cycle. 1: Taa is smaller than two clock cycles.	0 for EMC0 1 for EMC1

Vangotech

5.1.6.3. APB Control Register

When users read the register offset over a legal range (Offset: 0x0 and 0xC ~ 0xF4), EMC will response "Slave Error" from the APB port. The APB slave error mask register can mask the response, and always response "Slave Okay" from the APB port. If users write the register offset over a legal range, the APB port will always response "Slave Error".

Table 5-4. APB Controller Register (PCR, Offset = 0x8)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	ErrMask	RW	APB slave error indication masking 1'b0: Slave error response output 1'b1: Slave okay response output	1

5.1.6.4. IP Feature Register

This register is the hardware configured item.

Table 5-5. IP Feature Register (FEAR, Offset = 0xF8)

Bit	Name	Type	Description	Default Value
[31:24]	EMC_DATA_WIDTH	RO	AXI data bus width	64
[23:20]	-	Rsvd	Reserved	0
[19:16]	EMC_INTERLEAVE	RO	Memory bank number	1 for EMC0 2 for EMC1
[15:8]	EMC_AXI_ID	RO	AXI ID width	19
7	-	Rsvd	Reserved	0
[6:5]	EMC_ECC_BASE	RO	ECC base 1: 64	1
4	EMC_AXI_PROTOCOL	RO	AXI protocol 0: AXI3	0
[3:0]	EMC_SRAM	RO	SRAM size 2^{Init_size} Kbytes	5 for EMC0 7 for EMC1

5.1.6.5. IP Revision Register

Table 5-6. IP Revision Register (IVR, Offset = 0xFC)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:16]	MAJOR_VER	RO	Major version number	1
[15:8]	MINOR_VER	RO	Minor version number	3
[7:0]	REV_VER	RO	Revision number	0

Vangotech

5.2. eMMC

5.2.1. Overview

eMMC controller provides the programmable I/O and DMA for data transfers. DMA supports the data transfers between the memory and the eMMC through the AHB Master interface without being interrupted by CPU. DMA also supports the single-block transfer, multi-block transfer, and infinite transfer. The system address register points to the address of the descriptor table and sequentially accesses the data until the end of a transfer.

5.2.2. Features

eMMC Specification

- • Supports both DMA and non-DMA data transfers
- • Compliant with eMMC card specification, version 5.1 mandatory part
- Supports eMMC HS200, not support eMMC High Speed DDR and eMMC HS400
- Supports 4K SRAM for data FIFO
- Supports 8-bit eMMC card bus
- Supports CPRM function for security
- Built-in generation and check for 7-bit and 16-bit CRC data
- Card detection (Insertion/Removal)
- Supports Read Wait mechanism for SDIO function

CPRM Function

- Compliant with SD security specification in SD specifications
- Supports encryption using C2 cipher in Electronic CodeBook (ECB) mode, C2_E (k,d)
- Supports decryption using C2 cipher in ECB mode, C2_D (k,d)
- Supports encryption using C2 cipher in Converted-Cipher Block Chaining (C-CBC) mode, C2_ECBC (k,d)
- Supports decryption using C2 cipher in C-CBC mode, C2_DCBC (k,d)
- Supports C2 one-way function, C2_G (d1,d2)
- Supports C2 random number generator, RNGC2_G_EN (k,d)
- Supports automatic encryption/decryption in C-CBC mode

5.2.3. Block Diagram

Figure 5-2 shows the block diagram of the SD host controller with an AHB interface.

Vangotech

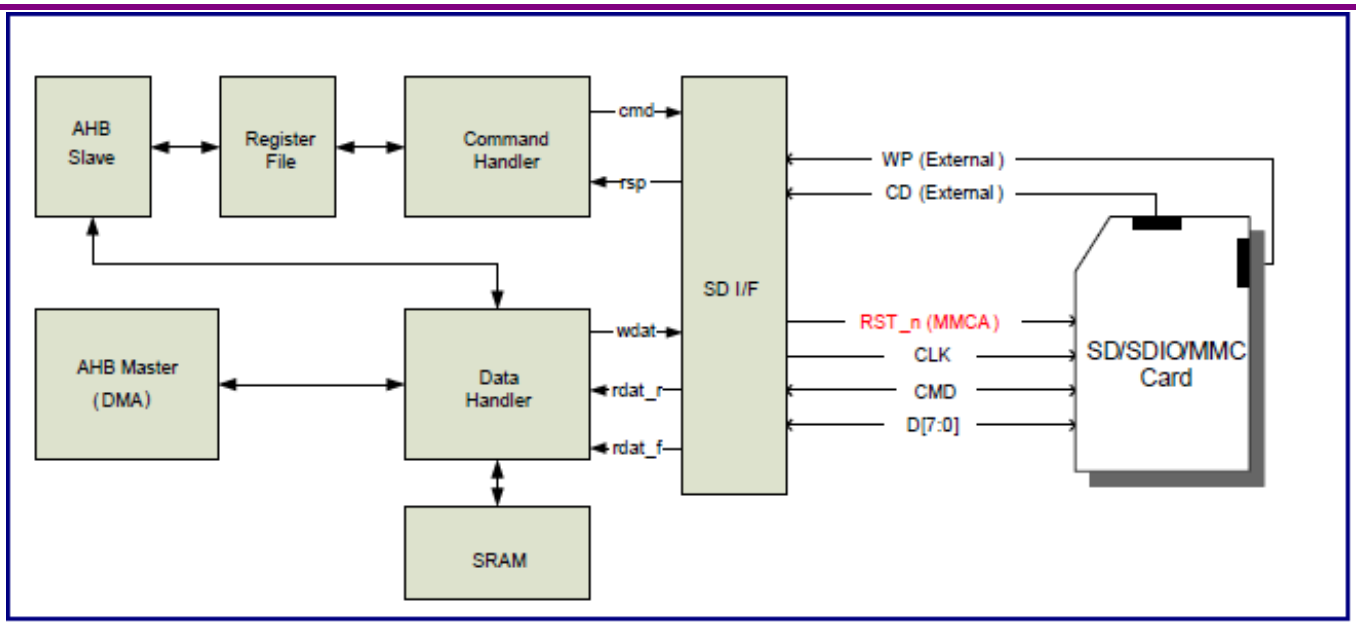


Figure 5-2. Block Diagram

5.2.4. Functional Descriptions

5.2.4.1. Block Description

The eMMC host controller includes the AHB master interface, AHB slave interface, Command Unit, DATA Unit, Control Register, Configurable FIFO, Clock Divider, and Configurable CPRM function.

AHB Master

If the Host IF is AHB, the AHB master module should be added to this IP. When the DMA transfer is used, the AHB Master will initiate a read/write transfer with the memory and the minimum data size will be limited to four bytes. The DMA algorithms defined in the SD host controller standard specification include SDMA and ADMA (Advanced DMA). SDMA has a disadvantage that the DMA interrupt generated at the every page boundary will disturb CPU in reprogramming the new system address. The new ADMA uses the link-list mechanism to provide a higher data transfer speed.

AHB Slave

The main function of the APB slave port is to handle the APB read and write transfers for accessing theregisters.

If the Host IF is AHB, the AHB slave module should be added to this IP. The AHB slave is implemented to read/write the host control registers by using the pRCessor through the slave interface. The AHB slave provides the byte, half-word, or word data access. Please note that the address must be 0x20 no matter the HSIZE is BYTE, HWORD, or WORD when accessing data port though the register.

The DMA handshake protocol is supported for the flow control when using the external DMA. Please note that the setting of the burst size in the DMA should match the setting of the block size if the FIFO size is in 256-word, 512-word, or 1024-word.

Command Unit

The command unit is implemented to transmit command to the SD card and receive the response from the SD card. The built-in 7-bit CRC check and generation are involved in the command unit.

Data Unit

The data unit is implemented to transmit data to the SD card and receive data from the SD card. The built-in 16-bit CRC check and generation are involved in the data unit.

Control Register

The control register can be divided into two groups: Command register and status register. To activate a command or data transfer with the card, the command register should be programmed properly. The failure, success, or received command/data CRC status will be reported in the status register under any condition.

Configuarable Data FIFO

The data FIFO is in the full-duplex mode and is configurable for a 32-bit data width with an 8-word or 16-word register file or the 256-word, 512-word, or 1024-word data depth SRAM.

Clock Divider

In the card identification mode, the maximum clock frequency is 400 kHz. During the data transfer, the maximum clock frequency rate should not exceed 25 MHz in the default speed mode, 50 MHz in the high-speed mode for the SD card, and 52 MHz for the eMMC card. When the card clock is stopped, the activated command, the received response, or the transfer data may be temporarily suspended in the command unit the sufficient clock cycles re-start for the card operation.

Configurable CPRM Function

The control register can be divided into two groups: Command register and status register. To activate a command or data transfer with the card, the command register should be programmed properly. The failure, success, or received command/data CRC status will be reported in the status register under any condition.

5.2.4.2. DMA Transaction

The DMA algorithms (Also called SDAM and ADMA) are implemented in this host controller. SDMA has a disadvantage that the DMA interrupt generated at every page boundary will disturb CPU in reprogramming the new system addresses. Since the host driver is required to reprogram the next system address to restart the transfer, the performance of the DMA transfer will be affected. The new ADMA implements the link-list mechanism to provide the higher data transfer speed. The host driver can program a list of descriptor line to transfer data without interrupting the host driver. The block diagram of ADMA is shown in Figure 5-3. The descriptor table is created in the system memory by the host driver.

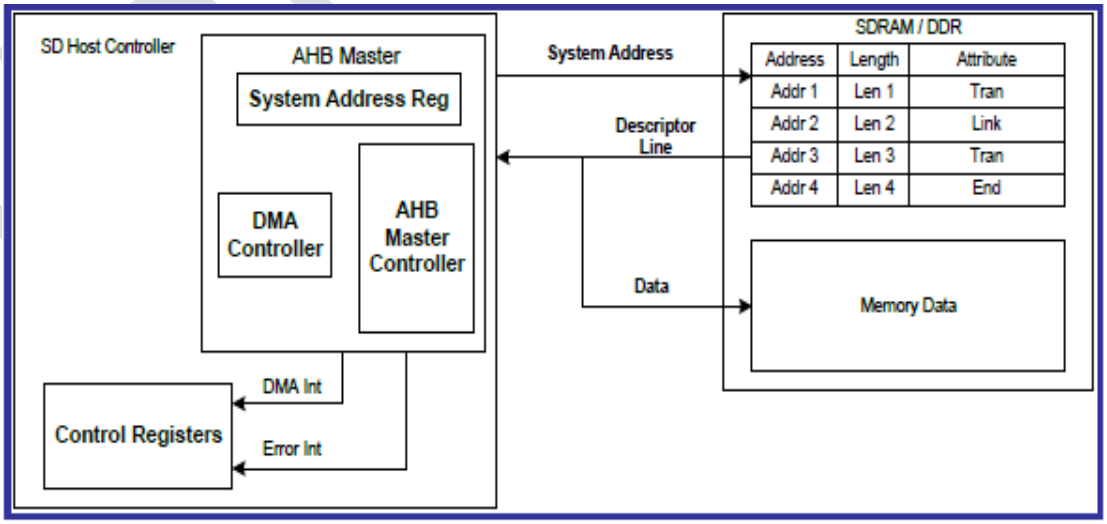


Figure 5-3. Block Diagram of ADMA

Data Address and Data Length

The minimum unit of an address is four bytes and the data address should be word-aligned. The maximum

data length of each descriptor line should be less than 64K bytes. The total data length should be equal to the block count multiplied by the block size. The block count register is defined as a 16-bit register, which will limit the maximum transfer of 65535 blocks.

(1) The address should be word-aligned (4-byte).

(2) The data length of each descriptor line should be word-aligned except for the last descriptor.

(3) The data length of each descriptor line is less than 64K bytes.

(4) Total data length = Len 1 + Len 2 + …… + Len n = Multiple block size

(5) If ADMA is used in the CPRM auto mode, the data length of the descriptor line should be the multiple of the block sizes.

Descriptor Table

Figure 5-4 shows the 32-bit address descriptor table. Each descriptor line requires the 64-bit memory space. The address field stores the 32-bit data address. The length field defines the data length of each descriptor line.

Attribute

The data in the attribute field is used to control each descriptor line.

Symbol

Three action symbols are specified:

• The "Nop" operation indicates that the current descriptor line is not in operation and fetches the next descriptor line

• The "Tran" operation indicates that the current descriptor line is the data transfer operation, which is designated by the data address and data length.

• The "Link" operation indicates the connection to the other descriptor lines, including the next system address and the other system addresses. The next descriptor line of the system address is generated by the data address of the current descriptor line.

Table 5-7 lists the defined data length according to the length field.

Address	Length	Reserved	Attribute					
63:32	31:16	15:6	5	4	3	2	1	0
32-bit data address	16-bit data length	000000	Act2	Act1	0	Int	End	Valid

Act2	Act1	Symbol	Description
0	0	Nop	Current descriptor is not executed and go to the next descriptor line
0	1	Reserved	-
1	0	Tran	Current descriptor line is executed to transfer data
1	1	Link	Link to the other descriptor line

Int	Int = 1 will generate **dma_interrupt_r** interrupt when the current descriptor line is done.
End	End indicates that the current descriptor line is the end of descriptor. When the current descriptor line is done, the **tran_complete_r** interrupt is generated.
Valid	Valid indicates that the current descriptor line is valid. If the bit is set to 0, that will generate **adma_err_r** interrupt and stop transaction.

Figure 5-4. 32-bit Descriptor Table

Table 5-7. ADMA Data Length Field

Length Field	Data Length
0xFFFF	65535 bytes
….	….

0x0002	2 bytes
0x0001	1 bytes
0x0000	65536 bytes

ADMA States

Figure 5-5 depicts the state diagram of the ADMA transfer. Four states are addressed in this state diagram: Stop transfer state, fetch descriptor state, change address state, and transfer data state. The state operations are defined in Table 5-8.

The ADMA transfer can neither be stopped at the block gap nor continue the available requests. If an error occurs during the ADMA transfer, the ADMA operation may stop and generate the **adma_err_r** interrupt. The **adma_err_st_r** registers hold the ADMA state and **adma_sys_addr_r** holds the system address around the error descriptor. The host driver can read the **adma_err_st_r** registers to check the state of the ADMA operation.

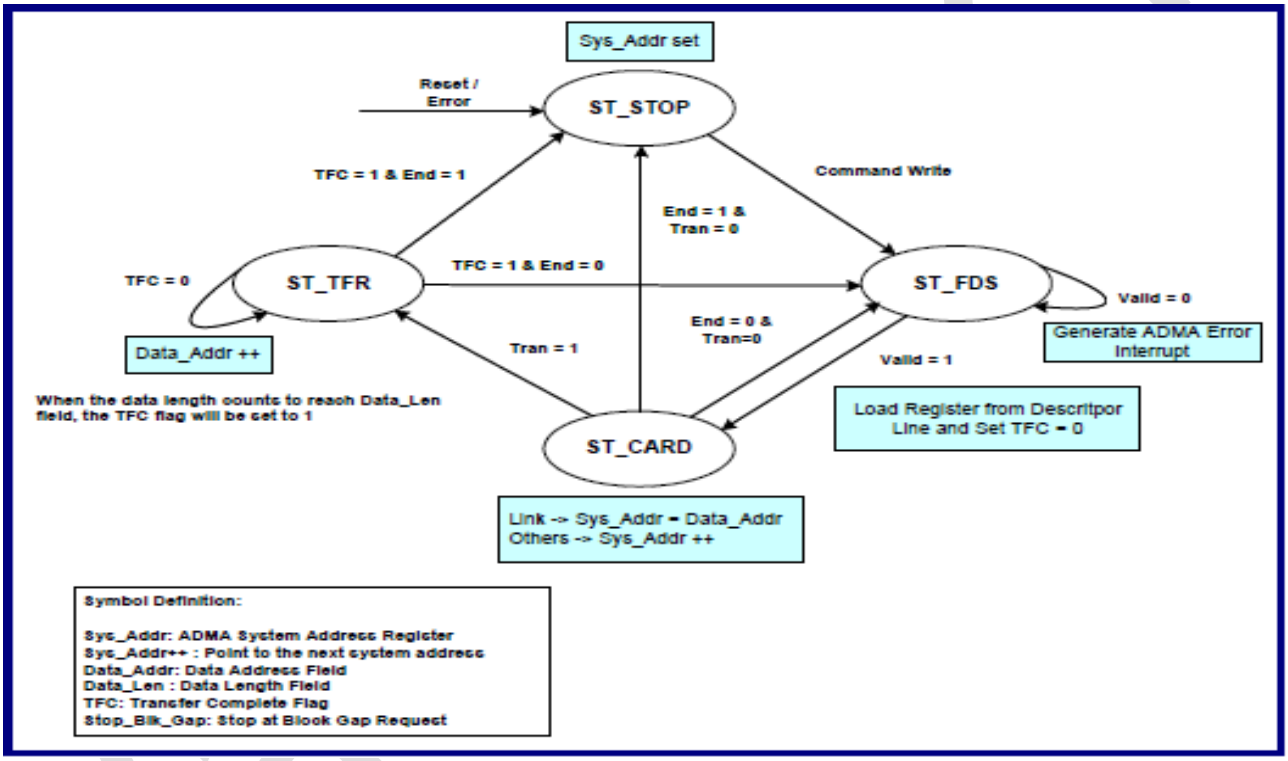


Figure 5-5. State Diagram of ADMA

Table 5-8. ADMA States

State	Description
ST_STOP (Stop DMA)	ADMA will stay at this state under the following conditions: (1) When Reset or Error occurs. (2) When all descriptor line data transfers are completed. If the command register is written, the state will go to the ST_FDS state.
ST_FDS (Fetch Descriptor)	In this state, a descriptor line will be fetched from the system memory to the host controller. After fetching a descriptor line, go to the ST_CADR state.
ST_CADR (Change Address)	In this state, the line system of the next descriptor address will be acquired. In the link operation, the next system address will be loaded by the data address

State	Description
	of current descriptor line and the state will change to the ST_FDS state to fetch the next descriptor line. In the other operations, the current system address will be incremented to point to the next descriptor line and the state will change to the ST_TFR state.
ST_TFR (Transfer Data)	The data transfer of one descriptor line will be executed between the system memory and the SD card. If the data transfer is not completed (End = '0'), it will go to the ST_FDS state to fetch the next descriptor line. If the data transfer is completed, it will go to the ST_STOP state.

NOTE: When an ADMA interrupt occurs, the asynchronous abort sequence should be issued.

5.2.4.3. SD Command and Data Input/Output Timing

For the SD interface timing, users can adjust the input/output timing to meet the timing margin.

Command and Data Output

For the SD interface output timing, the timing adjustment is designed in SDC . Table 5-9 lists different cases of the timing adjustment.

Table 5-9. Timing Adjustment of SD Interface Output

Clock Divided is Zero	Clock Divided is One	DDR Mode Enable	Int. Edge Enable	Output Timing Case
No	No	No	No	Case 1
No	No	No	Yes	Case 2
No	No	Yes	No	Case 2
No	No	Yes	Yes	Case 2
No	Yes	No	No	Case 1
No	Yes	No	Yes	Case 2
No	Yes	Yes	No	Case 3
No	Yes	Yes	Yes	Case 3
Yes	No	No	No	Case 3
Yes	No	No	Yes	Case 2
Others		Inhibit		

Where "clock divided is zero" means that the value of the 10-bit clock divider is zero; "clock divided is one" means that the value of the 10-bit clock divider is one; "DDR mode enable" means that UHS_mode_sel is set to the DDR50 mode (Host Control 2 register, 0x3E, bits[2:0]); and "Int. Edge Enable" means that bit[16] of the vendor register 0 (0x100) is set to '1'.

The output timing adjustment has three cases. Please refer to Figure 5-6 for details.

Vangotech

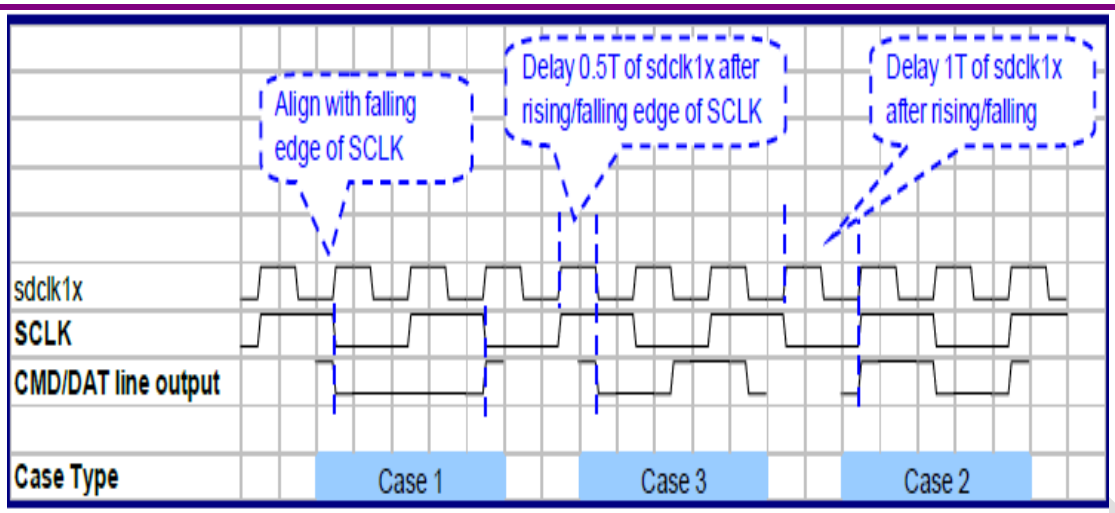


Figure 5-6. SD Output Timing

Command and Data Input Latching

For the SD input timing, users need to select the latching mechanism to avoid incorrect latching points. SDC provide two data latching mechanisms: Multiphase DLL latching and pulse latching.

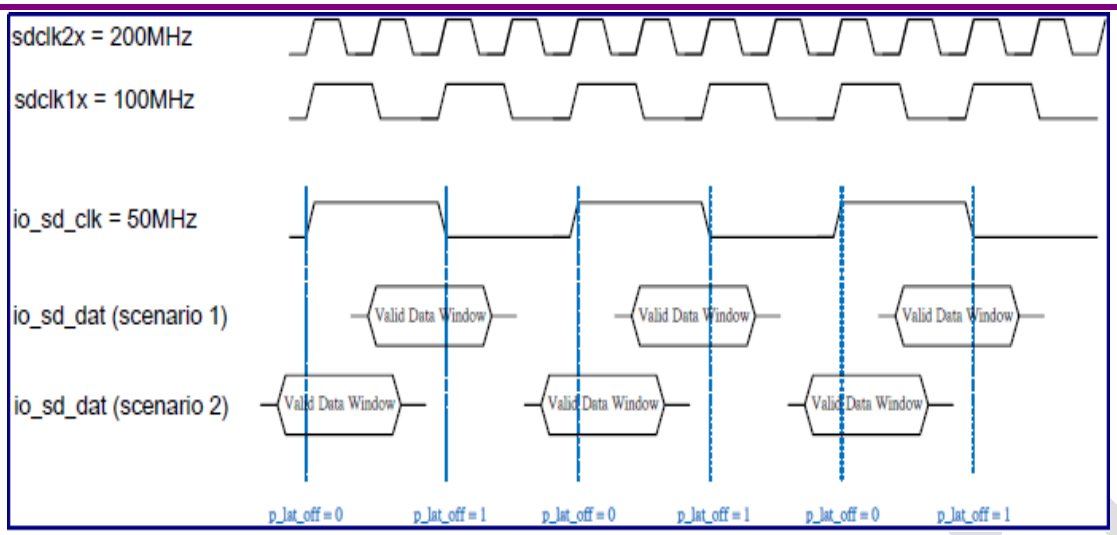
- Multiphase DLL latching mechanism: The system provides one multiphase DLL to adjust the SD clock delay for latching. The tuning/re-tuning pRCedure is defined by the SD standard specification, version 3.00, to acquire one correct delay value for the multiphase DLL. This mechanism applies at the high speed protocol, especially for SDR50/104.
- Pulse latching mechanism: This mechanism does not require extra multiphase DLL in the system. In general, the system interface has the round-trip effect between the chip and device. To eliminate the round-trip latency, SDC designs one adjustable latching pointer to avoid the issue. Users can set bits[13:8] of the vendor register 0 (0x100) to select the latching pointer. The default value is zero and the latching pointer is at the rising edge of the SD clock (SCLK). The latching pointer delays one sdclk1x period once the value increases by one.

Besides, the latching pointer number depends on the clock divider number (offset 0x2C). Users can compute the number by the following equation:

Latching pointer Range = 0 to 2N-1,where N is divider value at Clock Control Register (offset 0x2C).

The following figure shows one example, sdclk2x is 200 MHz andsdclk1x is 100 MHz. For SD High Speed mode, users have to run SD clock at 50 MHz. So, Clock Control Register N needs to set to 1 and the latching pointer (p_lat_off) can be 0 or 1.

Vangotech



Due to the round-chip effect and SD Card compatibility, the "Valid Data Window" of io_sd_dat can occur at scenario 1 or scenario 2 in the above figure. In that case, users need to adjust the latching pointer (p_lat_off) value to get correct read data.

Besides, for the latching pointer number equation, if the N is zeros then only one latching pointer and the value cannot be adjusted.

So, it is strongly recommended that the frequency of sdclk1x should be twice or more than the user proposed io_sd_clk. And then, users can adjust the latching pointer value when different scenario cases occur.

5.2.4.4. CPRM Operation

The CPRM operation defines a renewable method for protecting the contents recorded in a number of physical media types.

The CPRM operation is designed to satisfy the following applications:

- It is applicable for both the audio and video contents.
- It is suitable for implementing the PCs devices.
- It is applicable to different media types.

Figure 5-7 depicts the block diagram of the SD host controller with the CPRM function. Table 5-10 lists the basic characteristics of the C2 cipher. Each operation needs the 64-bit data to encrypt or decrypt. There are two basic operational modes: Electronic Codebook (ECB) mode and Converted Cipher Block Chaining (C-CBC) mode.

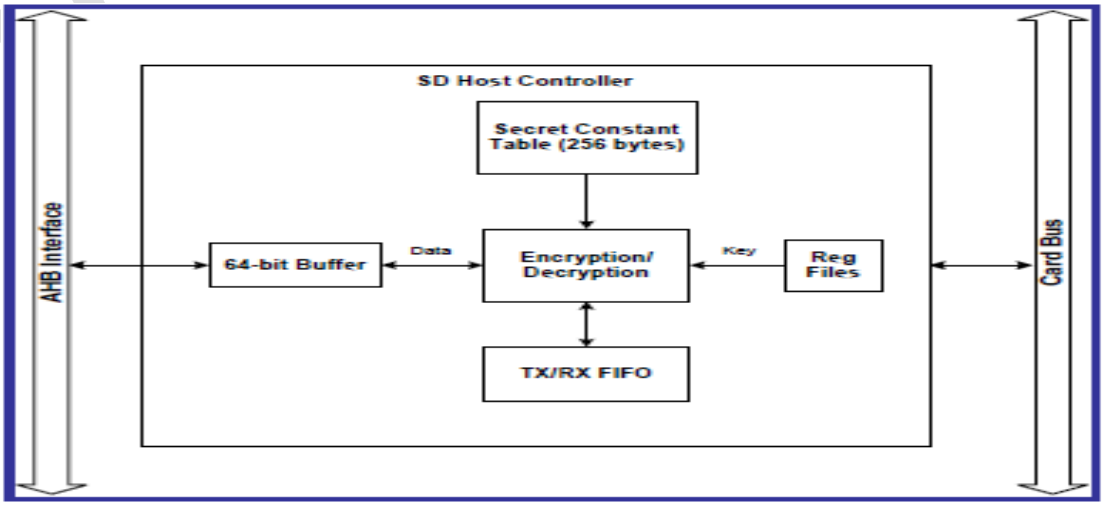


Figure 5-7. Host Controller with CPRM Function

Table 5-10. C2 Cipher Characteristics

Input block size	64 bits
Output block size	64 bits
Input key size	56 bits
Number of rounds	10 rounds

C2 Block Cipher in Electronic CodeBook (ECB) Mode

The encryption with the C2 cipher in the Electronic CodeBook (ECB) mode is represented by using the following function:

C2_E (k,d)

where *k* is a 56-bit key, *d* is the 64-bit data to be encrypted, and C2_E returns the 64-bit result.

The decryption with the C2 cipher in the ECB mode is represented by using the following function:

C2_D (k,d)

where *k* is a 56-bit key, *d* is the 64-bit data to be decrypted, and C2_D returns the 64-bit result.

Figure 5-8 depicts the encryption operation in the ECB mode. The decryption operation is similar to the one for encryption.

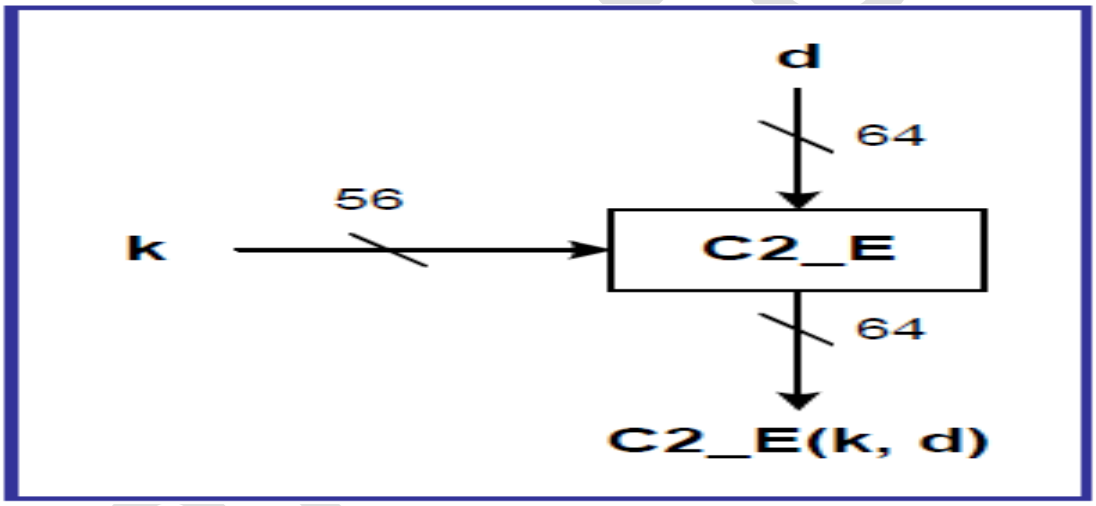


Figure 5-8. Encryption in ECB Mode

C2 Block Cipher in Converted Cipher Block Chaining (C-CBC) Mode

The encryption operation with the C2 cipher in the C-CBC mode is represented by using the following function:

C2_ECBC (k,d)

where *k* is a 56-bit key, *d* is a frame of data to be encrypted, and C2_ECBC returns the encrypted frame.

The decryption operation in the C-CBC mode is represented by using the following function:

C2_DCBC (k,d)

where *k* is a 56-bit key, *d* is a frame of data to be decrypted, and C2_DCBC returns the decrypted frame.

Figure 5-9 depicts the encryption of the C-CBC mode. The decryption operation is similar to the encryption.

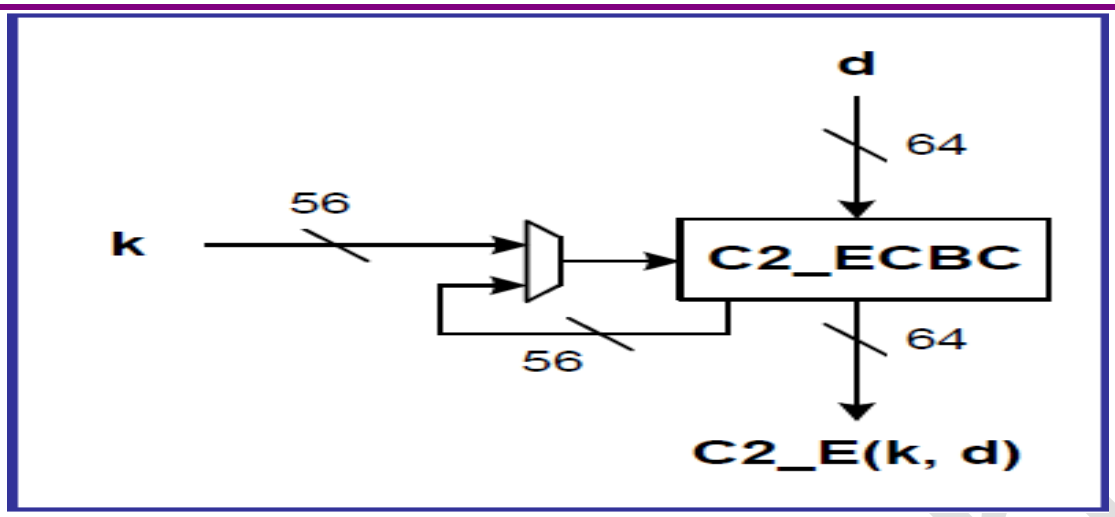


Figure 5-9. Encryption in C-CBC Mode

5.2.5. Register List

Module Name	Base Address
eMMC_0	0x1210 0000
eMMC_1	0x1220 0000

Register Name	Offset	Description
SDMADR	00	SDMA system address register/Argument 2
BSR	04	Block Size Register
BCR	06	Block Count Register
ARG1	08	Argument 1 Register
TMR	0C	Transfer Mode Register
CR	0E	Command Register
RESP0	10	Response 0 Register
RESP1	14	Response 1 Register
RESP2	18	Response 2 Register
RESP3	1C	Response 3 Register
BDP	20	Buffer Data Port
PSR	24	Present State Register
HC1	28	Host Control 1 Register
PCR	29	Power Control Register
BGCR	2A	Block Gap Control Register
CCR	2C	Clock Control Register
TCR	2E	Timeout Control Register
SRR	2F	Software Reset Register
NISR	30	Normal Interrupt Status Register
EISR	32	Error Interrupt Status Register
NISER	34	Normal Interrupt Status Enable Register
EISER	36	Error Interrupt Status Enable Register

Vangotech

Register Name	Offset	Description
NISEN	38	Normal Interrupt Signal Enable Register
EISEN	3A	Error Interrupt Signal Enable Register
AC12ES	3C	Auto CMD12 Error Status Register
HC2	3E	Host Control 2 Register
CAP0	40	Capabilities 0 Register
CAP1	44	Capabilities 1 Register
MCC0	48	Maximum Current Capabilities 0 Register
MCC1	4C	Maximum Current Capabilities 1 Register
FACERR	50	Force Event for the Auto CMD12 Error Status
FERR	52	Force Event for the Error Interrupt Status
AESR	54	ADMA Error Status Register
ALSAR	58	ADMA Low System Address Register
AHSAR	5C	ADMA High System Address Register
PV0	60	Preset Value 0
PV1	64	Preset Value 1
PV2	68	Preset Value 2
PV3	6C	Preset Value 3
HCVR	FE	Host Controller Version Register
VR0	100	Vendor-defined Register 0
VR1	104	Vendor-defined Register 1
VR2	108	Vendor-defined Register 2
VR3	10C	Vendor defined Register 3
VR4	110	Vendor-defined Register 4
VR5	114	Vendor-defined Register 5
VR6	118	Vendor-defined Register 6
VR7	11C	Vendor-defined Register 7
VR8	120	Vendor-defined Register 8
VR9	124	Vendor-defined Register 9
DHER	128	DMA Handshake Enable Register
HWA	178	Hardware Attributes
IPRR	17C	IP Revision Register
CMCR	180	Cipher Mode Control Register
CMSR	184	Cipher Mode Status Register
CMSE	188	Cipher Mode Signal Enable Register
LWID	18C	Low Word of Input Data Register
HWID	190	High Word of Input Data Register
LWIK	194	Low Word of Input Key Register
HWIK	198	High Word of Input Key Register
LWOD	19C	Low Word of Output Data Register
HWOD	1A0	High Word of Output Data Register

Vangotech

Register Name	Offset	Description
SCTDP	1A4	Secret Constant Table Data Port

5.2.6. Register Description

5.2.6.1. SDMA System Address Register

Table 5-11 lists the bit assignment of the SDMA system address register. This register is used to set the system memory address of an SDMA transfer or the second argument for Auto CMD23.

- **For the SDMA transfer**

  The host driver sets the register before issuing a command to start the SDMA transfer. After the SDMA buffer boundary reaches and stops an SDMA transfer, the next system address of the next contiguous data address can be read from this register. The SDMA transfer waits at every boundary specified by **sdma_buf_bound** in the block size register. When the host driver sets the next system address to this register, the host controller will restart the SDMA transfer.

- **For the Auto CMD23 argument**

  This register set a 32-bit block count to the argument of CMD23 while executing Auto CMD23. If Auto CMD23 is used with ADMA, the full 32-bit block will be used. If Auto CMD23 is used without ADMA, the available block count will be limited by **blk_cnt_r**. In this case, 65535 blocks is the maximum value.

Table 5-11. SDMA System Address Register (SDMADR, Offset: 0x00)

Bit	Name	Type	Description	Default Value
[31:0]	sdma_sys_addr_r	RW	SDMA system address register or Auto CMD23 argument 2	0x0

5.2.6.2. Block Size Register

Table 5-12 lists the bit assignments of the block size register. This register is used to configure the number of bytes in a data block. The SDMA system address register is updated at each system memory boundary during the SDMA transfer. When the SDMA transfer reaches each boundary, the host controller will trigger the **dma_interrupt_r** interrupt to request the host driver to update the SDMA system address.

Table 5-12. Block Size Register (BSR, Offset: 0x04)

Bit	Name	Type	Description	Default Value
15	-	Rsvd	Reserved	0x0
[14:12]	sdma_buf_bound	RW	Buffer boundary of the SDMA host 0b111 — 512K bytes 0b110 — 256K bytes 0b101 — 128K bytes 0b100 — 64K bytes 0b011 — 32K bytes 0b010 — 16K bytes 0b001 — 8K bytes	0x0

			0b000 — 4K bytes	
[11:0]	blk_size_r	RW	This register specifies the block size of data transfers for CMD17, CMD18, CMD24, CMD25, and CMD53 and can be set with values ranging from 1 up to the maximum buffer size. In the memory, this register should be set up to 512 bytes. It can be accessed only when no transaction is executed (i.e., After a transaction has been stopped). The read operations during the transfers may return an invalid value and the write operations should be ignored. 0x800 — 2048 bytes …. — - 0x200 — 512 bytes 0x1FF — 511 bytes …. — - 0x002 — 2 bytes 0x001 — 1 bytes 0x000 — No transfer	0x0

5.2.6.3. Block Count Register

Table 5-13 lists the bit assignment of the current block counts. The block count registeris set when the **blk_cnt_en**register is set to '1'.This register is used only for the multi-block transfers. The host controller will decrease the counting number during the data transfer and stop counting when it counts down to zero.

Table 5-13. Block Count Register (BCR, Offset: 0x06)

Bit	Name	Type	Description	Default Value
[15:0]	blk_cnt_r	RW	Block count of the current transfer 0Xffff — 65535 blocks …. — - 0x0002 — 2 blocks 0x0001 — 1 block 0x0000 — Stop counting	0x0

5.2.6.4. Argument 1 Register

Table 5-14 lists the bit assignments of the SD command argument. This register is assigned to bits[39:8] of the command field.

Table 5-14. Argument 1 Register (ARG1, Offset: 0x08)

Bit	Name	Type	Description	Default Value
[31:0]	arg1_r	RW	Command argument	0x0

5.2.6.5. Transfer Mode Register

Table 5-15 lists the bit assignments of the transfer mode register. The host driver should set this register before issuing the data transfer command or resume command. Table 5-16 lists the transfer mode command setting.

Table 5-15. Transfer Mode Register (TMR, Offset: 0x0C)

Bit	Name	Type	Description	Default Value
[15:6]	-	Rsvd	Reserved	0x0
5	multi_blk_rw	RW	Single/Multi-block selection 1: Multiple blocks 0: Single block	0x0
4	tran_dir_sel	RW	Data transfer direction selection 1: Read from the card to host 0: Write from the host to card	0x0
[3:2]	auto_cmd_en	RW	Auto CMD enable There are two methods to stop the read and write operations of multiple blocks: (1) Auto CMD12 Enable When this field is set to '01b', the Host Controller will automatically issue CMD12 when the last block transfer is completed. The Auto CMD12 error is indicated to the Auto CMD Error Status register. This bit should not be set by the Host Driver if CMD12 is not required by the command. User cannot issue ACMD before receiving transfer complete interrupt of Auto CMD12. (2) Auto CMD23 Enable When this bit field is set to '10b', the Host Controller will automatically issue a CMD23 before issuing a command specified in the Command Register. The following conditions are required to use the Auto CMD23. Auto CMD23 Supported (Host Controller Version is 3.00 or later) A memory card that supports CMD23 (SCR[33]=1)	0x0

Vangotech

			If DMA is used, it shall be ADMA. Only when CMD18 or CMD25 is issued (Note, the Host Controller does not check command index.) Auto CMD23 can be used with or without ADMA. By writing the Command Register, the Host Controller will issue CMD23 first and then issue a command specified by the Command Index in the Command Register. If the response errors of CMD23 are detected, the second command will not be issued. The Auto CMD23 error is indicated in the Auto CMD Error Status register. CMD23 with a block count value of 32-bit will be set to the SDMA System Address register 0b11 Reserved 0b10 Auto CMD23 Enable 0b01 Auto CMD12 Enable 0b00 Auto Command Disable	
1	blk_cnt_en	RW	Block count enable This bit is only valid for a multi-block transfer. When this bit is set to '0', the blk_cnt_r register will be disabled. The multi-block transfer will be an infinite transfer.	0x0
0	dma_en	RW	DMA enable This bit can enable the DMA transfer. The DMA mode can be selected by **dma_type** in the host control register. When a data transfer command is issued, the DMA transfer will begin the operation. 1: DMA data transfer 0: No data transfer or Non-DMA data transfer	0x0

Table 5-16. Transfer Command Setting

Single/Multi-block Select	**Block Count Enable**	**Block Count**	**Transfer Type**
0	Don't care	Don't care	Single transfer
1	0	Don't care	Infinite block transfer
1	1	Non-zero	Multi-block transfer
1	1	0	Stop Multi-block transfer

5.2.6.6. Command Register

Table 5-17 lists the bit assignments of the command register. The host driver should check the **Command Inhibit (CMD)** and **Command Inhibit (DAT)** bits in the present state register to determine whether the SD bus is free to transfer.

Table 5-17. Command Register (CR, Offset: 0x0E)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Description</th><th>Default Value</th></tr>
<tr><td>[15:14]</td><td>-</td><td>Rsvd</td><td>Reserved</td><td>0x0</td></tr>
<tr><td>[13:8]</td><td>cmd_idx</td><td>RW</td><td>Command Index
These bits should be assigned to bits[45:40] of the command field.</td><td>0x0</td></tr>
<tr><td>[7:6]</td><td>cmd_type</td><td>RW</td><td>Command Type
There are two types of special commands: Normal and Abort. Abort Command
If this command is set when executing a read transfer, the Host Controller should stop reading to the buffer. If this command is set when executing a write transfer, the Host Controller should stop driving the DAT line. After issuing the Abort command, the Host Driver should issue a software reset.
These bits should be set to ‘00b’ for all other commands.
<table>
<tr><td>0b11</td><td>Abort</td><td>CMD12, CMD52 for writing “I/O Abort” in CCCR</td></tr>
<tr><td>0b10</td><td>Reserved</td><td>Reserved for SDIO resume command</td></tr>
<tr><td>0b01</td><td>Reserved</td><td>Reserved for SDIO suspend command</td></tr>
<tr><td>0b00</td><td>Normal</td><td>Other commands</td></tr>
</table></td><td>0x0</td></tr>
<tr><td>5</td><td>data_pres_sel</td><td>RW</td><td>Data Present Select
This bit is set to ‘1’ to indicate that data is present and data transfer is enabled.
This bit is set to ‘0’ under the following conditions:
(1) Commands only using the CMD line (ex. CMD52)
(2) Commands with no data transfer but using the busy signal on DAT[0] line (R1b or R5b ex. CMD38)
(3) Resume command</td><td>0x0</td></tr>
<tr><td>4</td><td>cmd_idx_chk_en</td><td>RW</td><td>Command Index Check Enable
If this bit is set to ‘1’, the host controller will check the index field response to determine if the values</td><td>0x0</td></tr>
</table>

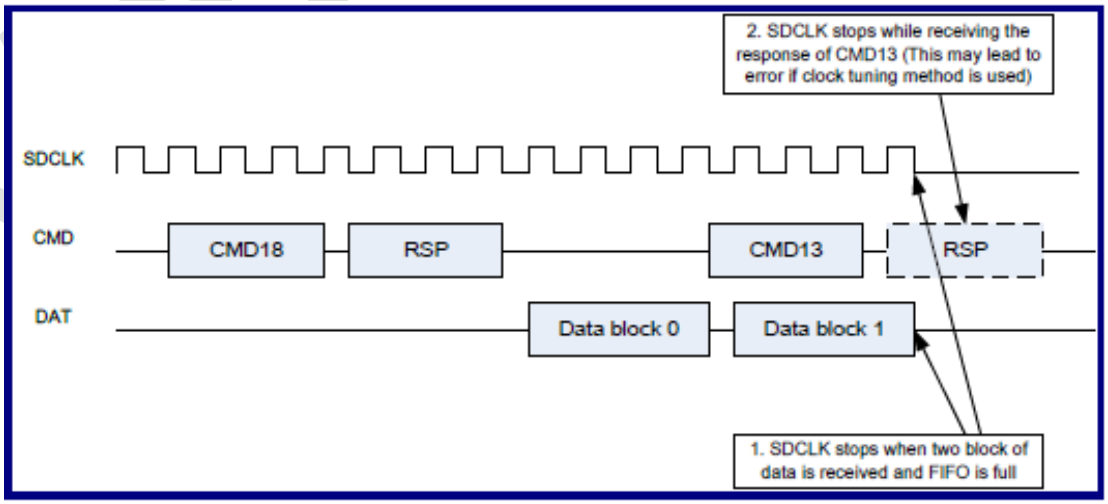
Vangotech

			are **cmd_idx**. If they are not the same, **cmd_idx_err** will be triggered.	
3	cmd_crc_chk_en	RW	Command CRC Check Enable If this bit is set to '1', the Host Controller will check the CRC field response to determine whether CRC is correct. If an error is detected, **cmd_crc_err** will be triggered.	0x0
2	-	Rsvd	Reserved	0x0
[1:0]	rsp_type_sel	RW	Response Type Select 0b11: Response length 48 with busy check after response 0b10: Response length 48 0b01: Response length 136 0b00: No response	0x0

Table 5-18. Relationship between Parameters and Names of Response Types

Response Type	**Command Index Check Enable**	**Command CRC Check Enable**	**Response**
00	0	0	No response
01	0	1	R2
10	0	0	R3, R4
10	1	1	R1, R5, R6, R7
11	1	1	R1b, R5b

When using the clock tuning method (In the SDR104/SDR50/DDR50 mode), it is forbidden to issue any command (Except CMD12) before a data transfer command is finished. If SDCLK stops during receiving a response, errors may occur in this response once the clock is resumed. Please refer to the following figure.



5.2.6.7. Response Registers 0 ~ 3

Table **5-20** lists the mapping of the command responses of each response type. The Host Controllerstores the Auto CMD12 response in the upper word of the Response Register to avoid the Auto

CMD12 response, which tends to be overwritten by the other command.

Table 5-19. Response Register 0 ~ 3 (RESPn, Offset: 0x10 ~ 0x1C, n = 0 ~ 3)

Bit	Name	Type	Description	Default Value
[31:0]	rsp_r	ROC	Command Response	0x0

Table 5-20. Response Bit Definition of Each Response Type

Response Type	Meaning of Response	Response Field	Response Register
R1, R1b (Normal response)	Card status	R[39:8]	rsp_0_r (rsp_r[31:0])
R1b (Auto CMD12 response)	Card status for Auto CMD12	R[39:8]	rsp_3_r (rsp_r[127:96])
R1 (Auto CMD23 response)	Card status for Auto CMD23	R[39:8]	rsp_3_r (rsp_r[127:96])
R2 (CID, CSD register)	CID or CSD register	R[127:8]	rsp_0_r ~ rsp_3_r (rsp_r[119:0])
R3 (OCR register)	OCR register for memory	R[39:8]	rsp_0_r (rsp_r[31:0])
R4 (OCR register)	OCR register for I/O	R[39:8]	rsp_0_r (rsp_r[31:0])
R5, R5b	SDIO response	R[39:8]	rsp_0_r (rsp_r[31:0])
R6 (RCA response)	RCA[31:16]	R[39:8]	rsp_0_r (rsp_r[31:0])

5.2.6.8. Buffer Data Port Register

Table 5-21 lists the Buffer Data Port Register. This register uses the 32-bit Data Port Register to access the internal buffer (Always access through the offset address 0x20 even when the transfer size is byte or half-word.)

Table 5-21. Buffer Data Port Register (BDP, Offset: 0x20)

Bit	Name	Type	Description	Default Value
[31:0]	data_port_r	RW	Buffer Data Port Register	0x0000_0000

5.2.6.9. Present State Register

Table 5-22 lists the present status of the host controller. The host driver can access the status from the 32-bit read-only register.

Table 5-22. Present State Register (PSR, Offset: 0x24)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0x0
24	cmd_lin_lv	RO	Command Line Signal Level This status bit is used to check the CMD line level.	0x0
[23:20]	data_lin_lv	RO	DATA[3:0] Line Signal Level	0x0

Vangotech

			This status bit is used to check the DATA[3:0] line level	
19	wr_prot_lv	RO	Write Protect Pin Level This status bit is used to reflect the write protect pin level. 1: Write enabled 0: Write protected	0x0
18	cd_pin_lv	RO	Card Detect Pin Level This status bit is used to reflect the inverse value of the card detect pin. It is stable when sys_card_stable is set to '1'. 1: Card is detected. 0: Card is not detected.	0x0
17	sys_card_stable	RO	Card State Stable This status bit is used to reflect whether the card detect pin is stable. 1: No card or card is inserted. 0: Reset or de-bounce	0x1
16	sys_card_insert	RO	Card Inserted This status bit indicates whether a card has been inserted. After de-bounced by the Host Controller, the host driver can check whether the card is inserted. The de-bouncing time can be determined by setting db_timeout in the Vendor-defined Register 5.This bit can be changed from 0 to 1 to generate a card_insert_r interrupt in the normal interrupt status register. This bit can be changed from 1 to 0 to generate a card_remove_r interrupt in the normal interrupt status register. 1: Card inserted 0: Reset, de-bouncing, or no card is detected.	0x0
[15:12]	-	Rsvd	Reserved	0x0
11	buf_ren_r	ROC	Buffer Read Enable This status bit is used for the non-DMA read transfer. This bit indicates that there are valid data exist in the Rx buffer and is ready for read. This bit can be changed from 1 to 0 to indicate that the Rx buffer data have been read from the buffer. This bit can be changed from 0 to 1 when all block data are ready in the buffer for read or when the Rx buffer is full and generates a buf_r_rdy_r interrupt in the normal interrupt status register. 1: Read enable	0x0

Vangotech

			0: Read disable Note: The buffer read enable is set when the amount of data that equals the FIFO depth is ready for read when the data FIFO is implemented by the register (Please check the register offset 0x178 for the hardware attribute). When the data FIFO is implemented by SRAM, this bit will be set when at least one block of data is ready for read.	
10	buf_wen_r	ROC	Buffer Write Enable This bit is used for the non-DMA write transfer. This bit indicates that the TX buffer is ready to receive data. If the bit is '1', data can be written to the TX buffer. This bit changes from 1 to 0 when all block data are written to the TX buffer or when the Tx buffer is full. This bit changes from 0 to 1 when all block data are written to the Tx buffer. 1: Write enable 0: Write disable Note: The buffer write enable is set when the amount of space that equals the FIFO depth is available for write if the data FIFO is implemented by the register (Please check the register offset 0x178 for the hardware attribute). When the data FIFO is implemented by SRAM, this bit will be set when at least one block of space is available for write. Note: This bit is still work when stop at block gap is set. But user should not base on this bit to write data to TX buffer when stop at block gap is set.	0x0
9	rd_tran_act_r	ROC	Read Transfer Active This status bit is used to detect the completion of a read transfer. This bit is set to '1' under the following conditions: (1) After the end bit of a read command (2) When cont_req in the block gap control register is set to restart a transfer. This bit is set to '0' under the following conditions: (1) When all data blocks specified by the block length are transferred to the system. (2) When sp_blk_gap_req in the block gap control register is set to '1' and the host controller has	0x0

Vangotech

			transferred all the valid data blocks to the system. The tran_complete_r interrupt is generated when this bit changes from '1' to '0'.	
8	wr_tran_act_r	ROC	Write Transfer Active This status bit indicates that a write transfer is active. This bit is set to '1' under the following conditions: (1) After the end bit of a write command (2) When cont_req in the block gap control register is set to restart a transfer. This bit is set to '0' under the following conditions: (1) After getting the CRC status of the last data block specified by the transfer count. (2) After getting the CRC status of any block where data transmission is stopped by sp_blk_gap_req. A blk_gap_evt_r interrupt will be generated when sp_blk_gap_req is set to '1' and this bit changes to '0.' This bit is useful in the command with a busy data line.	0x0
[7:3]	-	Rsvd	Reserved	0x0
2	data_lin_act_r	ROC	Data Line Active This bit is used to determine whether the Data Line is in use. In a read transfer, this status bit is used to check whether a read transfer is executing on the bus. Changing this bit from 1 to 0 will generate a blk_gap_evt_r interrupt when sp_blk_gap_req is set to '1'. This bit is set to '1' under the following conditions: (1) After the end bit of a read command (2) When cont_req in the block gap control register is set to restart a transfer. This bit is set to '0' under the following conditions: (1) When the end bit of the last data block is sent from the SD bus to the host controller. (2) When sp_blk_gap_req is set to '1' and a read transfer is stopped at the block gap. In a write transfer, this status bit is used to check whether a write transfer is executing on the bus. Changing this bit from 1 to 0 will generate a tran_complete_r interrupt in the normal interrupt status register. This bit is set to '1' under the following conditions: (1) After the end bit of a read command	0x0

Vangotech

			(2) When cont_req in the block gap control register is set to restart a transfer. This bit is set to '0' under the following conditions: (1) When the card release the busy signal of the last data block. (2) When sp_blk_gap_req is set to '1' and the card releases the write busy at the block gap. In the command with busy data line, this bit indicates whether a command with busy is executing on the bus. This bit will be set after the end bit of the command with busy and will be cleared when busy is de-asserted or busy is not detected after the end of a response.	
1	cmd_inhibit_d	ROC	Command Inhibit (DAT) This bit will be generated if data_lin_act_r or rd_tran_act_r is set to '1'. When this bit is set to '0', the host driver can issue a new command by using the data line. Command with busy data line belongs to cmd_inhibit_d. 1: Cannot issue new commands to use the data line 0: Issue new commands to use the data line	0x0
0	cmd_inhibit_c	ROC	Command Inhibit (CMD) When this bit is set to '0', the host driver will issue a new command by using the command line. When this bit is set to '1', the command register will be written. This bit is cleared when the command response is received. Even if cmd_inhibit_d is set to '1', the commands using the command line can be issued if this bit is set to '0'. CMD0, CMD12, CMD13, and CMD52 can be issued when the data lines are in use during a data transfer. The cmd_complete_r interrupt in the normal interrupt register is issued when this bit is changed from 1 to 0. However, if the present command suffer some errors (CRC error, command index error, and so on), this bit will be remained '1'. Note: This bit will not be set by Auto CMD12. 1: Cannot issue command 0: Issue command only with the command line	0x0

5.2.6.10. Host Control 1 Register

Table 5-23 lists the bit assignments of the host control 1 register.

Table 5-23. Host Control 1 Register (HC1, Offset: 0x28)

Vangotech

Bit	Name	Type	Description	Default Value
7	cd_sel	RW	Card Detect Signal Selection This bit is used to select the source of card detection. 1: The test level for the card detection 0: The card detect pin is selected.	0x0
6	cd_test_lv	RW	Card Detect Test Level This bit is used to set the test level when cd_sel is set to '1'. 1: Card is inserted. 0: Card cannot be found.	0x0
5	ext_data_width	RW	Extended Data Transfer Width This bit controls the 8-bit bus width mode for the embedded device, which is indicated in the 8-bit Support for the Embedded Device in the Capabilities register. When this bit is set to '1', it indicates that the 8-bit bus mode is supported by the device. When this bit is set to '0', it indicates that the bus width is controlled by the Data Transfer Width in the Host Control 1 register. This bit will not be effective if multiple devices are installed on a bus slot (Slot Type is set to '10b' in the Capabilities register). In this case, each device bus width will be controlled by the Bus Width Preset field in the Shared Bus Control register. 1: 8-bit bus width 0: Bus width is selected by the data transfer width.	0x0
[4:3]	dma_type	RW	DMA Type Select 0b11 Reserved 0b10 32-bit address ADMA2 is selected 0b01 Reserved 0b00 SDMA is selected	0x0
2	hi_speed	RW	High Speed Enable If this bit is set to '0', the host controller can output a command and data at the falling edge of io_sd_clk at a speed of up to 25 MHz. If this bit is set to '1', the host controller can output a command and data at the rising edge of io_sd_clk at a speed of up to 50 MHz.	0x0

Vangotech

Bit	Name	Type	Description	Default Value
[1:0]	-	Rsvd	Reserved	0x0
0	led_ctrl	RW	LED Control This bit is used to notify users that the SD card has been accessed. 1: LED on 0: LED off	0x0

5.2.6.11. Power Control Register

Table 5-24 lists the bit assignment of the power control register.

Table 5-24. Power Control Register (PCR, Offset: 0x29)

Bit	Name	Type	Description	Default Value
[7:4]	-	Rsvd	Reserved	0x0
[3:1]	sd_bus_vol	RW	SD Bus Voltage Select By setting these bits, the Host Driver will select the voltage level for the SD card. Before setting this register, the Host Driver should check the Voltage Support bits in the Capabilities Register. If an unsupported voltage is selected, the Host System should not supply the SD bus voltage.	0x0
0	sd_bus_pow	RW	SD Bus Power This bit is used to enable the SD bus power. If no card is detected by the Host Controller, this bit will be set to '0'. The io_sd_clk signal cannot be driven to '0' by setting this bit. It should be implemented at the system level.	0x0

5.2.6.12. Block Gap Control Register

Table 5-25 lists the bit assignments of the Block Gap Control Register.

Table 5-25. Block Gap Control Register (BGCR, Offset: 0x2A)

Bit	Name	Type	Description	Default Value
[7:4]	-	Rsvd	Reserved	0x0
3	Int_at_blk_gap	RW	Interrupt at Block Gap This bit is only useful in the 4-bit mode of the SDIO card. Setting this bit to '1' will enable the interrupt detection at the block gap for the multiple block transfers. This bit should be set to '0' if the SDIO card cannot	0x0

Vangotech

			issue an interrupt during the block gap. 1: Check the interrupt at block gap enable 0: Check the interrupt at block gap disable	
2	read_wait	RW	Read Wait Control This bit is useful for the SDIO card. When the SDIO card supports the read wait function, the host controller can use the read wait protocol to control the SDIO bus. When the card does not support the read wait function, the host controller will stop io_sd_clk to hold the read transfer. 1: Enable the read wait function 0: Disable the read wait function	0x0
1	cont_req	RWAC	Continue Request This bit is used to restart a transaction, which can be stopped by using sp_blk_gap_req. To restart a transaction, this bit should be set to '1' and sp_blk_gap_req should be set to '0'. The host controller will automatically clear this bit under the following conditions: (1) In a read transfer, data_lin_act_r will be changed from 0 to 1 to start a read transfer. (2) In a write transfer, wr_tran_act_r will be changed from 0 to 1 to start a write transfer. 1: Restart 0: No effect	0x0
0	sp_blk_gap_req	RW	Stop at Block Gap Request This bit is used to stop executing the read or write transfer at the next block gap. This bit is useful for the non-DMA and SDMA, but is not useful for ADMA. The host driver should keep the setting of this bit to '1' until the tran_complete_r interrupt is set to '1'. If this bit is set to '1', the host controller will stop at the block gap by using read_wait or stop io_sd_clk in a read transaction. If this bit is set to '0', the host controller will not write data to data_port_r. 1: Stop 0: Transfer	0x0

Vangotech

5.2.6.13. Clock Control Register

Table 5-26 lists the bit assignments of the Clock Control Register.

Table 5-26. Clock Control Register (CCR, Offset: 0x2C)

<table>
<tr><th>Bit</th><th>Name</th><th>Type</th><th>Description</th><th>Default Value</th></tr>
<tr><td>[15:8]</td><td>low_bit_sd_clk_sel</td><td>RW</td><td>SD clock frequency value [7:0] for the 10-bit divided clock mode
This bit is used to select the frequency of the io_sd_clk pin. The base clock is specified by sdc_clk (Base clock = sdclk1x).
<table>
<tr><td>10`f3FF</td><td>Base clock divided by 2046</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>N</td><td>Base clock divided by 2N (Duty 50%)</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td>0x002</td><td>Base clock divided by 4</td></tr>
<tr><td>0x001</td><td>Base clock divided by 2</td></tr>
<tr><td>0x000</td><td>Base clock (sdclk1x)</td></tr>
</table></td><td>0x0</td></tr>
<tr><td>[7:6]</td><td>Upper_bit__sd_clk_sel</td><td>RW</td><td>SD clock frequency value [9:8] for the 10-bit divided clock mode</td><td>0x0</td></tr>
<tr><td>5</td><td>clk_gen_sel</td><td>ROC</td><td>This bit is always set to zero.
1: Programmable clock mode
0: 10-bit divided clock mode</td><td>0x0</td></tr>
<tr><td>[4:3]</td><td>-</td><td>Rsvd</td><td>Reserved</td><td>0x0</td></tr>
<tr><td>2</td><td>sd_clk_en</td><td>RW</td><td>SD Clock Enable
When this bit is set to ‘1’, io_sd_clk will be outputted to the SD card.
When this bit is set to ‘0’, io_sd_clk will be stopped at ‘0’.</td><td>0x0</td></tr>
<tr><td>1</td><td>clk_stable</td><td>ROC</td><td>Internal Clock Stable
This bit is set to ‘1’ when the internal clock is stable. After inter_clk_en is set to ‘1’, the internal clock will begin oscillating.</td><td>0x0</td></tr>
<tr><td>0</td><td>Inter_clk_en</td><td>RW</td><td>Internal Clock Enable
When the Host Controller is not use by the Host Driver, this bit can be set to ‘0’ to stop the internal clock to be in the low-power state. When this bit is set to ‘1’, the internal clock will begin oscillating.</td><td>0x0</td></tr>
</table>

Vangotech

5.2.6.14. Timeout Control Register

Table **5-27** lists the bit assignmentsof the Timeout Control Register. This host driver should set the timeout value according to the **cap_r** register. The value of data_timer indicates the data line timeout times.

Table 5-27. Timeout Control Register (TCR, Offset: 0x2E)

Bit	Name	Type	Description	Default Value
[7:4]	-	Rsvd	Reserved	0x0
[3:0]	data_timer	RW	**Data Timeout Counter Value** 0b1111 Reserved 0b1110 sys_hclk x 2^{27} 0b1101 sys_hclk x 2^{26} …. … 0b0000 sys_hclk x 2^{13}	0b1110

5.2.6.15. Software Reset Register

Table 5-28 lists the bit assignments of the Software Reset Register. A reset pulse will be generated when this bit is set to '1'. This bit will be automatically cleared once the reset pulse is issued. Please note that these bits cannot be set again until the auto clear of last time setting has been enabled.

Table 5-28. Software Reset Register (SRR, Offset: 0x2F)

Bit	Name	Type	Description	Default Value
[7:3]	0	Rsvd	Reserved	0x0
2	soft_rst_dat	RWAC	Software Reset for Data Line The following registers and bits will be cleared by this reset bit: (1) Buffer Data Port Register data_port_r (2) Present State Registers buf_ren_r, buf_wen_r, rd_tran_act_r, wr_tran_act_r, data_lin_act_r, and cmd_inhibit_d (3) Block Gap Control Registers cont_req and sp_blk_gap_req (4) Normal Interrupt Status Registers buf_r_rdy_r, buf_w_rdy_r, dma_interrupt_r, blk_gap_evt_r, and tran_complete_r	0x0
1	soft_rst_cmd	RWAC	Software Reset for Command Line The following registers and bits will be cleared by this reset bit: (1) Present State Register cmd_inhibit_c	0x0

Vangotech

Bit	Name	Type	Description	Default Value
			(2) Normal Interrupt Status Register cmd_complete_r	
0	soft_rst_all	RWAC	Software Reset for All The reset signal resets all registers except for the card detection and capability register.	0x0

5.2.6.16. Normal Interrupt Status Register

Table 5-29 lists the bit assignments of the Normal Interrupt Status Register. The interrupt status can be latched by setting the **nor_int_st_en_r** register to '1'.

Table 5-29. Normal Interrupt Status Register (NISR, Offset: 0x30)

Bit	Name	Type	Description	Default Value
15	err_interrupt_r	ROC	Error Interrupt This bit will be set to '1' if any bit in the err_int_status_r register is set to '1'.	0x0
[14:13]	-	Rsvd	Reserved	0x0
12	re_tuning_int_r	ROC	Re-Tuning Event This status will be set if the re-tuning request in the Present State Register changes from '0' to '1'. The host controller requests the host driver to perform re-tuning for the next data transfer. The current data transfer (Not the large block count) can be completed without re-tuning. 1: Re-tuning is required. 0: Re-tuning is not required.	0x0
11	Int_C_r	ROC	INT_C This status bit will be set if INT_C is enabled and the INT_C# pin is at the low level. Writing this bit to '1' will not clear this bit. This bit can be cleared by resetting the INT_C interrupt factor.	0x0
10	Int_B_r	ROC	INT_B This status bit will be set if INT_B is enabled and the INT_B# pin is at the low level. Writing this bit to '1' will not clear this bit. This bit can be cleared by resetting the INT_B interrupt factor.	0x0
9	Int_A_r	ROC	INT_A This status will be set if INT_A is enabled and INT_A# pin is at the low level. Writing this bit to '1' will not clear this bit. This bit can be cleared by resetting the	0x0

			INT_A interrupt factor.	
8	card_int_r	ROC	Card Interrupt In the 4-bit mode, the card interrupt is sampled during the interrupt cycle. When this bit is set to '1', the interrupt will need to be handled by the Host Driver. The host driver needs to set card_int_st_en to '0' and clear the card_int_r status to avoid driving the interrupt signal to the host system again. If the Host Driver has handled the card interrupt completely, the card_int_st_en should be set to '1' to re-start sampling. Note: When issuing abort command with card interrupt is enabled, there might be some dummy card interrupts. Users are recommended enabling card interrupt only when card interrupt function is needed.	0x0
7	card_remove_r	RW1C	Card Remove This status bit will be set if sys_card_inst in the Present State Register changes form '1' to '0'. When the host driver writes '1' to this bit, this status will be cleared.	0x0
6	card_insert_r	RW1C	Card Insert This status bit will be set if sys_card_inst in the Present State Register changes from '0' to '1'. When the host driver writes '1' to this bit, this status will be cleared.	0x0
5	buf_r_rdy_r	RW1C	Buffer Read Ready This status bit will be set if buf_ren_r in the Present State Register changes from '0' to '1'.	0x0
4	buf_w_rdy_r	RW1C	Buffer Write Ready This status bit will be set if buf_wen_r in the Present State Register changes from '0' to '1'.	0x0
3	dma_interrupt_r	RW1C	DMA Interrupt This status bit will be set if the host controller detects that the SDMA buffer boundary is reached during a transfer, except in the last block is completed. In case of ADMA, the int field will be set in the descriptor line and the host controller will generate the interrupt once the descriptor line is done.	0x0
2	blk_gap_evt_r	RW1C	Block Gap Event By setting sp_blk_gap_req to '1', this bit will be set when a read or write transaction is stopped at the	0x0

Vangotech

			block gap. (1) Read transaction This bit will be set when data_lin_act_r changes from '1' to '0'. The read wait should be supported for this function in the SDIO card. (2) Write transaction This bit will be set when wr_tran_act_r changes from '1' to '0'.	
1	tran_complete_r	RW1C	Transfer Complete A data transfer will be completed or a data transfer will be stopped at the block gap during a transmission to generate a transfer complete interrupt. (1) Read transaction This bit will be set when rd_tran_act_r changes from '1' to '0'. (2) Write transaction This bit will be set when data_lin_act_r changes from '1' to '0'. (3) Command with busy This bit will be set when the busy data line is released.	0x0
0	cmd_complete_r	RW1C	Command Complete This bit will be set when the command response is received and the CRC check is okay. Auto CMD12 and Auto CMD23 will not generate the command complete interrupt. The following table shows that the Command Timeout Error bit has higher priority than the Command Complete bit. When both bits are set to '1', it indicates that the response has not been correctly received. Command Complete / Command Timeout Error / Meaning of Status 0 / 0 / Interrupted by another factor Don't care / 1 / Response not received within 64 SCLK cycles 1 / 0 / Response received	0x0

5.2.6.17. Error Interrupt Status Register

Table 5-30 lists the status of the Error Interrupt Status Register. The interrupt status can be latched when the **err_int_st_en_r** registers are set to '1'.

Table 5-30. Error Interrupt Status Register (EISR, Offset: 0x32)

Bit	Name	Type	Description	Default Value
[15:11]	-	Rsvd	Reserved	0x0
10	Tuning_err_r	RW1C	Tuning Error This bit is set when an unrecoverable error is detected in a tuning circuit except during the tuning pRCedure (Occurrence of an error during the tuning pRCedure is indicated by the Sampling Select). By detecting the Tuning Error, the Host Driver needs to abort a command executing and perform tuning. To reset the tuning circuit, the Sampling Clock should be set to '0' before executing the tuning pRCedure. The Tuning Error has higher priority than the other error interrupts generated during the data transfer. By detecting the Tuning Error, the Host Driver should discard the data transferred by a current read/write command and retry data transfer after the Host Controller is retrieved from the tuning circuit error.	0x0
9	adma_err_r	RW1C	ADMA Error This bit will be set to '1' when an error is detected in the ADMA error status register during the ADMA transfer. In addition, the Host Controller will set this interrupt when it detects the invalid descriptor data (Valid = 0).	0x0
8	auto_cmd12_err_r	RW1C	Auto CMD12 Error This bit will be set to '1' when any error is detected in the auto CMD12 error status register.	0x0
7	cur_lim_err_r	RW1C	Current Limit Error By setting the SD Bus Power bit in the Power Control register, the Host Controller will be requested to supply power for the SD Bus. If the Host Controller supports the Current Limit function, it can be protected from an illegal card by stopping supplying power to the card. In this case, this bit will indicate a failure status. Reading 1 means that the Host Controller is not supplying power to the SD card due to some failures. Reading 0 means that the Host Controller is supplying power and no error has occurred. The Host Controller may require some sampling time to detect the current limit. This bit should be always set to '0' if the Host Controller does not support this function.	0x0
6	data_end_bit_err_r	RW1C	Data End Bit Error This status is set when the host controller detects 0 at the	0x0

Vangotech

Bit	Name	Type	Description	Default Value
			end bit of the read data, which uses the data line or at the end bit of the write CRC status.	
5	data_crc_err_r	RW1C	Data CRC Error This status is set when the host controller detects the CRC error during the read transfer or detects the write CRC status that is not of the value of "010".	0x0
4	data_timeout_err_r	RW1C	Data Timeout Error This bit will be set under the following conditions: (1) Wait to read data timeout (2) Wait write CRC status timeout (3) Busy timeout after write CRC status (4) Busy timeout for the R1b and R5b types	0x0
3	cmd_idx_err_r	RW1C	Command Index Error This bit will be set when the command index in the command response is different from the command.	0x0
2	cmd_end_bit_err_r	RW1C	Command End Bit Error This bit will be set when the end bit of the command response is 0.	0x0
1	cmd_crc_err_r	RW1C	Command CRC Error This bit will be set when the command CRC error is detected.	0x0
0	cmd_timeout_err_r	RW1C	Command Timeout Error This bit will be set when no response is sent from the card within 64 SD card clock.	0x0

5.2.6.18. Normal Interrupt Status Enable Register

Table 5-31 lists the bit assignments of the Normal Interrupt Status Enable Register. If the corresponding bit of the interrupt source in the normal interrupt status enable register is set to '1', the interrupt becomes active, which is latched and available for the host driver in the normal interrupt status register.

Table 5-31. Normal Interrupt Status Enable Register (NISER, Offset: 0x34)

Bit	Name	Type	Description	Default Value
15	Reserved	RO	Fixed to '0' Error interrupts are controlled by the error interrupt status enable register.	0x0
[14:13]	-	Rsvd	Reserved	0x0
12	re_tuning_st_en	RW	Re-Tuning Event Status Enable	0x0
11	Int_C_st_en	RO	Fixed to '0'	0x0
10	Int_B_st_en	RO	Fixed to '0'	0x0

Vangotech

Bit	Name	Type	Description	Default Value
9	Int_A_st_en	RO	Fixed to '0'	0x0
8	card_int_st_en	RW	Card Interrupt Status Enable If this bit is set to '1', card_int_r will be served by the Host Driver. The Host Driver should set this bit to '0' before serving card_int_r and should reset this bit after card_int_r completes requesting card. The detection of card_int_r will be stopped when this bit is set to '0' and will be restarted when this bit is set to '1'.	0x0
7	card_remove_st_en	RW	Card Remove Status Enable	0x0
6	card_insert_st_en	RW	Card Insert Status Enable	0x0
5	buf_r_rdy_st_en	RW	Buffer Read Ready Status Enable	0x0
4	buf_w_rdy_st_en	RW	Buffer Write Ready Status Enable	0x0
3	dma_interrupt_st_en	RW	DMA Interrupt Status Enable	0x0
2	blk_gap_evt_st_en	RW	Block Gap Event Status Enable	0x0
1	tran_complete_st_en	RW	Transfer Complete Status Enable	0x0
0	cmd_complete_st_en	RW	Command Complete Status Enable	0x0

5.2.6.19. Error Interrupt Status Enable Register

Table 5-32 lists the bit assignments of the Error Interrupt Status Enable Register. If the corresponding bit of the interrupt source in the Error Interrupt Status Enable Register is set to '1' and if the interrupt becomes active, the active state will be latched and will be available for the host driver in this register.

Table 5-32. Error Interrupt Status Enable Register (EISER, Offset: 0x36)

Bit	Name	Type	Description	Default Value
[15:11]	-	Rsvd	Reserved	0x0
10	Tuning_err_st_en	RW	Tuning Error Status Enable	0x0
9	adma_err_st_en	RW	ADMA Error Status Enable	0x0
8	auto_cmd12_err_st_en	RW	Auto CMD12 Error Status Enable	0x0
7	cur_lim_err_st_en	RW	Current Limit Error Status Enable	0x0
6	data_end_bit_err_st_en	RW	Data End Bit Error Status Enable	0x0
5	data_crc_err_st_en	RW	Data CRC Error Status Enable	0x0
4	data_timeout_err_st_en	RW	Data Timeout Error Status Enable	0x0
3	cmd_idx_err_st_em	RW	Command Index Error Status Enable	0x0
2	cmd_end_bit_err_st_en	RW	Command End Bit Error Status Enable	0x0
1	cmd_crc_err_st_en	RW	Command CRC Error Status Enable	0x0
0	cmd_timeout_err_st_en	RW	Command Timeout Error Status Enable	0x0

Vangotech

5.2.6.20. Normal Interrupt Signal Enable Register

Table 5-33 lists the bit assignments of the Normal Interrupt Signal Enable register. This register is used to select the interrupt status that is notified to the host system as an interrupt. These interrupt statuses share the same interrupt line.

Table 5-33. Normal Interrupt Signal Enable Register (NISEN, Offset: 0x38)

Bit	Name	Type	Description	Default Value
[15:13]	-	Rsvd	Reserved	0x0
12	re_tuning_sig_en	RW	Re-Tuning Event Signal Enable	0x0
11	Int_C_sig_en	RO	Fixed to '0'	0x0
10	Int_B_sig_en	RO	Fixed to '0'	0x0
9	Int_A_sig_en	RO	Fixed to '0'	0x0
8	card_int_sig_en	RW	Card Interrupt Signal Enable	0x0
7	card_remove_sig_en	RW	Card Remove Signal Enable	0x0
6	card_insert_sig_en	RW	Card Insert Signal Enable	0x0
5	buf_r_rdy_sig_en	RW	Buffer Read Ready Signal Enable	0x0
4	buf_w_rdy_sig_en	RW	Buffer Write Ready Signal Enable	0x0
3	dma_interrupt_sig_en	RW	DMA Interrupt Signal Enable	0x0
2	blk_gap_evt_sig_en	RW	Block Gap Event Signal Enable	0x0
1	tran_complete_sig_en	RW	Transfer Complete Signal Enable	0x0
0	cmd_complete_sig_en	RW	Command Complete Signal Enable	0x0

5.2.6.21. Error Interrupt Signal Enable Register

Table 5-34 lists the bit assignments of the Error Interrupt Signal Enable register. This register is used to select the interrupt status that is regarded by the host system as an interrupt. These interrupt statuses share the same interrupt line.

Table 5-34. Error Interrupt Signal Enable Register (EISEN, Offset: 0x3A)

Bit	Name	Type	Description	Default Value
[15:11]	-	Rsvd	Reserved	0x0
10	tuning_err_sign_en	RW	Tuning Error Signal Enable	0x0
9	adma_err_sig_en	RW	ADMA Error Signal Enable	0x0
8	auto_cmd12_err_sig_en	RW	Auto CMD12 Error Signal Enable	0x0
7	cur_lim_err_sign_en	RW	Current Limit Error Signal Enable	0x0
6	data_end_bit_err_sig_en	RW	Data End Bit Error Signal Enable	0x0
5	data_crc_err_sig_en	RW	Data CRC Error Signal Enable	0x0
4	data_timeout_err_sig_en	RW	Data Timeout Error Signal Enable	0x0
3	cmd_idx_err_sig_em	RW	Command Index Error Signal Enable	0x0
2	cmd_end_bit_err_sig_en	RW	Command End Bit Error Signal Enable	0x0

vangotech

Bit	Name	Type	Description	Default Value
1	cmd_crc_err_sig_en	RW	Command CRC Error Signal Enable	0x0
0	cmd_timeout_err_sig_en	RW	Command Timeout Error Signal Enable	0x0

5.2.6.22. Auto CMD12 Error Status Register

Table 5-35 lists the status of the Auto CMD12 Error Status register. When the auto_cmd12_en register is set to '1' and the Auto CMD12 error status register is set, the host driver will check this register to identify what kind of error happens during executing AUTO CMD12. This register will be valid only when **auto_cmd12_err_r** is set to '1'.

Table 5-35. Auto CMD12 Error Status Register (AC12ES, Offset: 0x3C)

Bit	Name	Type	Description	Default Value
[15:8]	-	Rsvd	Reserved	0x0
7	cmd_no_ex_by_cmd12_r	ROC	Command Not Executed by Auto CMD12 Error This bit indicates that the command followed Auto CMD12 is not executed due to an Auto CMD12 error (Bit 1 to bit 4) in this register. The bit will be set to '0' when Auto CMD Error is generated by Auto CMD23.	0x0
[6:5]	-	Rsvd	Reserved	0x0
4	auto_cmd_idx_err_r	ROC	Auto CMD Index Error	0x0
3	auto_cmd_end_bit_err_r	ROC	Auto CMD End Bit Error	0x0
2	auto_cmd_crc_err_r	ROC	Auto CMD CRC Error	0x0
1	auto_cmd_timeout_err_r	ROC	Auto CMD Timeout Error	0x0
0	auto_cmd12_no_ex_r	ROC	Auto CMD12 Not Executed If the memory multiple block data transfer is not started due to the command error, this bit will not be set because it is not necessary to issue Auto CMD12. Setting this bit to '1' means that the Host Controller cannot issue Auto CMD12 to stop the memory multiple block data transfer due to some errors. If this bit is set to '1', other error status bits (D0[4 :1] are meaningless. This bit is set to '0' when Auto	0x0

The relationship between Auto CMD CRC Error and Auto CMD Timeout Error is shown in Table 5-36.

Table 5-36. Relationship between CRC Error and Timeout Error for Auto CMD

Auto CMDCRC Error	Auto CMD Timeout Error	Error Type	Auto CMDCRC Error

0	0	No Error	0
0	1	Response Timeout Error	0
1	0	Response CRC Error	1
1	1	CMD line conflict	1

5.2.6.23. Host Control 2 Register

Table 5-37 lists the bit assignments of the Host Control 2 register.

Table 5-37. Host Control 2 Register (HC2, Offset: 0x3E)

Bit	Name	Type	Description	Default Value
15	preset_val_en	RW	Preset Value Enable Since the operating SDCLK frequency and I/O driving strength are dependent on the Host System implementation, it is difficult to determine these parameters in the Standard Host Driver. When this bit is set, the SDCLK frequency will be automatically generated and the driver strength will be selected without considering the specific system conditions. This bit enables the functions defined in the Preset Value registers. 1: Automatic selection by the preset value 0: SDCLK and driver strength are controlled by the Host driver.	0x0
14	asyn_int_en	RW	Asynchronous Interrupt Enable This bit can be set to '1' if a card supports the asynchronous interrupts and the Asynchronous Interrupt Support is set to '1' in the Capabilities register. If this bit is set to '1', the Host driver will stop SDCLK during the asynchronous interrupt period to save power. During this period, the Host controller will continue delivering the Card Interrupt to the host when it is asserted by the card.	0x0
[13:8]	-	Rsvd	Reserved	0x0
7	sample_clk_sel	RW	Sampling Clock Select This bit is used by the Host controller to select the sampling clock to receive CMD and DAT. This bit is set by the tuning procedure and is valid after the completion of tuning. Setting this bit to '1' means that the tuning is completed successfully.	0x0

Vangotech

			Setting this bit to '0' means that the tuning is failed. 1: Tuned clock is used to sample data. 0: Fixed clock is used to sample data.	
6	execute_tuning	RWAC	Execute Tuning This bit is set to '1' to start the tuning procedure and will be automatically cleared when the tuning procedure is completed. The result of tuning is indicated to Sampling Clock Select. The tuning procedure is aborted by writing '0'. 1: Execute tuning 0: Not tuned or tuning completed	0x0
[5:4]	driver_str_sel	RW	Driver Strength Select This bit selects the Host Controller output driver in the 1.8 V signaling. This field is not effective in the 3.3 V signaling. This field can be set according to the driver type A; types C and D support the bits in the Capabilities register. 0b11 Driver type D is selected. 0b10 Driver type C is selected. 0b01 Driver type A is selected. 0b00 Driver type B is selected.(Default)	0x0
3	1V8_sig_en	RW	1.8-V Signaling Enable This bit controls the voltage regulator for the I/O cell. 3.3 V is supplied to the card regardless of the signaling voltage. 1: 1.8-V signaling 0: 3.3-V signaling	0x0
[2:0]	UHS_mode_sel	RW	UHS Mode Select This field is used to select one of the UHS-I modes and is effective when the 1.8-V Signaling Enable is set to '1'. 0b111 Reserved 0b110 Reserved 0b101 Reserved 0b100 DDR50 0b011 SDR104 0b010 SDR50 0b001 SDR25 0b000 SDR12	0x0

Vangotech

5.2.6.24. Capabilities 0/1 Register

Table 5-38 lists the information of the Capabilities 0 and Capabilities 1 Register. The host controller may implement these values during initialization.

Table 5-38. Capabilities 0/1 Register (CAP, Offset: 0x40)

Bit	Name	Type	Description	Default Value
[63:56]	-	Rsvd	Reserved	0x0
[55:48]	clk_multiplier	RO	Clock Multiplier This field indicates the value of the clock multiplier of the programmable clock generator. Not support and is fixed to '0x0'.	0x0
[47:46]	re_tuning_mode	RO	Re-Tuning Modes This field selects the re-tuning method and limits the maximum data length. Re-Tuning Mode 1 The host controller does not have any internal logic to detect when the re-tuning needs to be performed. In this case, the Host driver should maintain all the re-tuning timings by using a Re-Tuning timer. To enable the insertion of the re-tuning procedure during data transfers, the data length per read/write command should be limited up to 4 MB. Only supports mode 1 and is fixed to '0b00'.	0x0
45	tuning_SDR50	RW	Use Tuning for SDR50 If this bit is set to '1', this Host controller will require the tuning to operate SDR50. If the version setting of host controller is not 3.0 (i.e. spec_ver_num in register 0xFE is not 0b10), then this field will be set to '0b0'.	0x1
44	-	Rsvd	Reserved	0x0
[43:40]	timer_cnt_for_re_tuning	RO	Timer Count for Re-Tuning This field indicates an initial value of the re-tuning timer for the Re-Tuning	0xF

Vangotech

Bit	Name	Type	Description	Default Value
			Mode 1 through Mode 3. If the version setting of host controller is not 3.0 (i.e. spec_ver_num in register 0xFE is not 0b10), then this field will be set to '0b0000'.	
39	-	Rsvd	Reserved	0x0
38	driver_D_support	RW	Driver Type D Support This bit indicates that the Driver Type D is supported for the 1.8 V signaling.	0x1
37	driver_C_support	RW	Driver Type C Support This bit indicates that the Driver Type C is supported for the 1.8 V signaling.	0x1
36	driver_A_support	RW	Driver Type A Support This bit indicates that the Driver Type A is supported for the 1.8 V signaling.	0x1
35	-	Rsvd	Reserved	0x0
34	DDR50_support	RW	DDR50 Support	0x1
33	SDR104_support	RW	SDR104 Support 1: SDR104 will be supported if SDC _MP_DLL_ON is defined. 0: SDR104 is not supported.	0x1
32	SDR50_support	RW	SDR50 Support	0x1
[31:30]	Slot_type	RW	Slot Type This field indicates the usage of a slot by a specific Host System (A host controller register set is defined as per slot). Embedded Slot for One Device (01b) indicates that only one non-removable device will be connected to a SD bus slot. Shared Bus Slot (10b) can be set if the Host Controller supports the Shared Bus Control register. The version supports "Removable Card slot" or "Embedded Slot for One Device". This bit can only be set to either '00' or '01'.	0x1
29	async_int_support	RO	Asynchronous Interrupt Support	0x1

Vangotech

Bit	Name	Type	Description	Default Value
			Please refer to the SDIO Specification, Version 3.00, on the asynchronous interrupt.	
28	bus_64_support	RO	64-bit System Bus Support Setting this bit to '1' indicates that the host controller supports the 64-bit address descriptor mode and is connected to the 64-bit address system bus. Not support and is fixed to '0'.	0x0
27	-	Rsvd	Reserved	0x0
26	voltage_1_8_support	RW	Voltage supports 1.8 V.	0x1
25	voltage_3_0_support	RW	Voltage supports 3.0 V.	0x0
24	voltage_3_3_support	RW	Voltage supports 3.3 V.	0x0
23	suspend_resume_support	RW	Suspend/Resume Support This bit suspends/resumes the SDIO function of the host controller.	0x1
22	sdma_support	RW	SDMA Support This bit indicates whether the host controller can use SDMA to transfer data.	HwInit
21	hi_speed_support	RW	High Speed Support This bit indicates whether the host controller can support the high-speed mode and can supply io_sd_clk from 25 MHz to 50 MHz.	0x1
20	adma1_support	RO	ADMA1 Support This bit indicates whether the host controller is capable of using legacy ADMA1. Not support and is fixed to '0'.	0x0
19	adma2_support	RW	ADMA2 Support This bit indicates whether the host controller is capable of using ADMA2.	HwInit
18	8bit_support	RW	8-bit Support for Embedded Device This bit indicates whether the host controller is capable of using the 8-bit bus width mode. The feature depends on the hardware	HwInit

Vangotech

Bit	Name	Type	Description	Default Value
			configuration, SDC _BUS_*n*. 1: The 8-bit bus width is supported if SDC _BUS_8 is defined. 0: The 8-bit bus width is not supported if SDC _BUS_4 is defined.	
[17:16]	max_blk_len	RW	Maximum Block Length This value indicates the maximum block size that can be read and written by the host driver to the buffer in the host controller. The buffer should transfer this block size without wait cycles. The feature depends on the hardware configuration, SDC _DFIFO_*n*. The following three sizes can be defined: 0b10: 2048 bytes if SDC _DFIFO_REG_8 or SDC _DFIFO_REG_16 or SDC _DFIFO_SRAM_4K is defined. 0b01: 1024 bytes if SDC _DFIFO_SRAM_2K is defined. 0b00: 512 bytes if SDC _DFIFO_SRAM_1K is defined.	HwInit
[15:8]	base_clk_for_SD_clk	RW	Base Clock Frequency For SD Clock This value indicates the base (Maximum) clock frequency for the SD Clock. 0xFF: 255 MHz …: … 0x02: 2 MHz 0x01: 1 MHz 0x00: Get information via another method If the real frequency is 16.5 MHz, the lager value should be set to 0x11 (17 MHz) because this value is used by the host driver to calculate the clock divider value. The value should not	HwInit

Vangotech

Bit	Name	Type	Description	Default Value
			exceed the upper limit of the SD clock frequency. If these bits are all '0', the host system has to acquire information via another method.	
7	timeout_clk_unit	RW	Timeout Clock Unit This bit shows the unit of the base clock frequency used to detect data_timeout_err_r.	0x1
6	-	Rsvd	Reserved	0x0
[5:0]	timeout_clk_freq	RW	Timeout Clock Frequency This bit shows the base clock for detecting data_timeout_err_r. The base clock is dominated by the system clock. Users should obtain the system clock frequency	0x0

5.2.6.25. Maximum Current Capabilities Register

Table **5-39** lists the maximum current capability for each voltage. The value is meaningful when the voltage support is set in the capabilities register. If this register is set to '0', the information will be supplied by the host system via another method. The current unit of this register is in a step of 4 mA. **错误!未找到引用源。** lists the current value of the register.

Table 5-39. Maximum Current Capabilities Register (MCC, Offset: 0x48)

Bit	Name	Type	Description	Default Value
[63:24]	-	Rsvd	Reserved	0x0
[23:16]	max_cur_for_vlg_1_8	RW	Maximum current for a voltage of 1.8 V	0x0
[15:8]	max_cur_for_vlg_3_0	RW	Maximum current for a voltage of 3.0 V	0x0
[7:0]	max_cur_for_vlg_3_3	RW	Maximum current for a voltage of 3.3 V	0x0

This register measures the current in a step of 4 mA. The current value of each voltage level is listed in Table 5-40.

Table 5-40. Definitions of Maximum Current Value

Register Value	Current Value
0	Get information via another method
1	4 mA
2	8 mA
3	12 mA

Vangotech

….	….
255	1020 mA

5.2.6.26. Force Event Register for Auto CMD Error Status

Table 5-41 lists the Force Event register for the Auto CMD error status. The Force Event register is not a physical register. It is an address to which the Auto CMD error status register can be written. The force event register is only for debugging.

Table 5-41. Force Event Register for Auto CMD Error Status (FACERR, Offset: 0x50)

Bit	Name	Type	Description	Default Value
[15:8]	-	Rsvd	Reserved	0x0
7	f_cmd_no_ex_by_cmd12	WO	Force event for the Command Not Executed by Auto CMD12 Error	0x0
[6:5]	-	Rsvd	Reserved	0x0
4	f_cmd_idx_err	WO	Force event for the Auto CMD Index Error	0x0
3	f_cmd_end_bit_err	WO	Force event for the Auto CMD End Bit Error	0x0
2	f_cmd_crc_err	WO	Force event for the Auto CMD CRC Error	0x0
1	f_cmd_timeout_err	WO	Force event for the Auto CMD Timeout Error	0x0
0	f_cmd12_no_ex	WO	Force event for the Auto CMD12 Not Executed	0x0

5.2.6.27. Force Event Register for Error Interrupt Status

Table 5-42 lists the Force Event Register for the error interrupt status. The Force Event register is not a physical register. It is an address to which the error interrupt status register can be written. This Force Event register is for debugging only. The effect of writing to this address will be reflected in the error interrupt status register if the corresponding bit of the error interrupt status enable register is set.

Table 5-42. Force Event Register for Error Interrupt Status (FERR, Offset: 0x52)

Bit	Name	Type	Description	Default Value
[15:13]	-	Rsvd	Reserved	0x0
12	f_ahb_resp_err	WO	Force Event for the AHB response Error	0x0
[11:10]	-	Rsvd	Reserved	0x0
9	f_adma_err	WO	Force Event for the ADMA Error	0x0
8	f_auto_cmd_err	WO	Force Event for the Auto CMD Error	0x0
7	f_cur_lim_err	WO	Force Event for the Current Limit Error	0x0
6	f_data_end_bit_err	WO	Force Event for the Data End Bit Error	0x0
5	f_data_crc_err	WO	Force Event for the Data CRC Error	0x0

Vangotech

Bit	Name	Type	Description	Default Value
4	f_data_timeout_err	WO	Force Event for the Data Timeout Error	0x0
3	f_cmd_idx_err	WO	Force Event for the Command Index Error	0x0
2	f_cmd_end_bit_err	WO	Force Event for the Command End Bit Error	0x0
1	f_cmd_crc_err	WO	Force Event for the Command CRC Error	0x0
0	f_cmd_timeout_err	WO	Force Event for the Command Timeout Error	0x0

5.2.6.28. ADMA Error Status Register

Table 5-43 lists the status of the ADMA Error Status register. When the ADMA error interrupt is occurred, adma_err_st_r holds the ADMA state and the ADMA system address register holds the address around the error descriptor. When an error occurs, the host driver should watch the ADMA state to identify the error descriptor address.

Table 5-43. ADMA Error Status Register (AESR, Offset: 0x54)

Bit	Name	Type	Description	Default Value
[7:3]	-	Rsvd	Reserved	0x0
2	adma_len_err	ROC	ADMA Length Mismatch Error This error occurs under the following two conditions: (1) When Block Count Enable is set, the total data length specified by the Descriptor table will be different from the one specified by the Block Count and Block Length. (2) The total data length cannot be divided by the block length.	0x0
[1:0]	adma_err_st_r	ROC	ADMA Error State This field indicates the state of ADMA when an error occurs during an ADMA transfer. <table><tr><th>D01~D00</th><th>ADMA Error State when error is occurred</th><th>Contents of SYS_SDR register</th></tr><tr><td>0b11</td><td>ST_TFR (Transfer Data)</td><td>Points next of the error descriptor</td></tr><tr><td>0b10</td><td>Never set this state</td><td>(Not used)</td></tr><tr><td>0b01</td><td>ST_FDS (Fetch Descriptor)</td><td>Points the error descriptor</td></tr></table>	0x0

Bit	Name	Type	Description			Default Value
			0b00	ST_STOP (Stop DMA)	Points next of the error descriptor	

5.2.6.29. ADMA System Address Register

Table **5-44** lists the address of the ADMA System Address register. The version of the Host controller only supports the 32-bit Address ADMA function.

Table 5-44. ADMA System Address Register (ALSAR, Offset: 0x58)

Bit	Name	Type	Description	Default Value
[31:0]	adma_lo_addr_r	RW	Lower 32-bit ADMA System Address The 32-bit address descriptor only uses the lower 32-bit of the ADMA system address register. ADMA ignores the lower 2-bit of this register and assumes it to be 00.	0x0

5.2.6.30. Preset Value Register

Table 5-45 lists a set of the preset values for each card or device.

Table 5-45. Preset Value Registers (PVn, Offset: 0x60 ~ 0x6F, n = 0 ~ 3)

Offset	Preset Value Register	Signal Voltage	Default Value
0x60	Preset value for initialization	3.3 V or 1.8 V	16'h0200
0x62	Preset value for the default speed	3.3 V	16'h0008
0x64	Preset value for the high speed	3.3 V	16'h0004
0x66	Preset value for SDR12	1.8 V	16'h0008
0x68	Preset value for SDR25	1.8 V	16'h0004
0x6A	Preset value for SDR50	1.8 V	16'h0002
0x6C	Preset value for SDR104	1.8 V	16'h0001
0x6E	Preset value for DDR50	1.8 V	16'h0004

When Preset Value Enable in the Host Control 2 register is set to '1', SDCLK Frequency Select in the Clock Generator Select of the Clock Control register and Driver Strength Select in the Host Control 2 register will be automatically set based on the selected bus speed mode. This means that the Host driver does not need to set these fields when preset is enabled. A Preset Value for Initialization (060h) will not be selected by the bus speed mode. Before starting the initialization sequence, the Host driver needs to set a clock preset value to SDCLK Frequency Select in the Clock Control register. Preset Value Enable can be set after the initialization procedure is complete.

Table **5-46** lists the speed mode selections used for the Preset Value register.

Table 5-46. Speed Mode Selection

Selected Bus Speed	1.8 V Signaling	High Speed Enable	UHS-I Mode Selection

Vangotech

Mode	Enable (Host Control 2)	(Host Control 1)	(Host Control 2)
Default speed	0	0	Don't care
High speed	0	1	Don't care
SDR12	1	Don't care	0b000
SDR25	1	Don't care	0b001
SDR50	1	Don't care	0b010
SDR104	1	Don't care	0b011
DDR50	1	Don't care	0b100
Reserved	1	Don't care	0b101 ~ 0b111

Table **5-47**defines the bit assignment of the Preset Value register.

Table 5-47. Preset Value Register Based on Speed Mode

Bit	Name	Type	Description	Default Value
[15:14]	driver_str_sel	RO	Driver Strength Select Value Driver Strength is supported by the 1.8 V signaling bus speed modes. This field is meaningless for the 3.3 V signaling. 0b11 — Driver type D is selected. 0b10 — Driver type C is selected. 0b01 — Driver type A is selected. 0b00 — Driver type B is selected.	HwCfg
[13:11]	-	Rsvd	Reserved	0x0
10	clk_gen_sel	RO	Clock Generator Select Value The version does not support the programmable clock generator and is fixed to '0'.	0x0
[9:0]	SDCLK_freq_sel	RO	SDCLK Frequency Select Value The 10-bit preset value for setting the SDCLK Frequency Select in the Clock Control The register is described by a host system.	0x0

5.2.6.31. Host Controller Version Register

Table 5-48 lists the bit assignment of the Host Controller Version register.

Table 5-48. Host Controller Version Register (HCVR, Offset: 0XFE)

Bit	Name	Type	Description	Default Value
[15:8]	vendor_ver_num	RO	Vendor Version Number This status is reserved for the vendor version number. The host driver should not use this status.	0
[7:0]	spec_ver_num	RW	Specification Version Number	2

			This field shows the specification version of the supportable host controller. 0b00: SD Host Specification Version 1.00 0b01: SD Host Specification Version 2.00 0b10: SD Host Specification Version 3.00 Others: Reserved	

5.2.6.32. Vendor-defined Register 0

Table 5-49 lists the assignment of the Vendor-defined Register 0.

Table 5-49. Vendor-defined Register 0 (VR0, Offset: 0X100)

Bit	Name	Type	Description	Default Value
[31:28]	-	Rsvd	Reserved	0x0
[27:24]	NCRC	RW	Write CRC Status Wait Cycle The host controller is used to set five SCLK clock cycles for the specifications and round-chip effect. Users can add the wait cycle if the SD card has no meet the specification.	0x0
[23:17]	-	Rsvd	Reserved	0x0
16	int_edge_sel	RW	Internal SCLK Edge Selection 1: The CMD and DAT line output at the rising edge of SCLK. 0: The CMD and DAT line output at the falling edge of SCLK. Please note that this value must be '0' in SDMA mode.	0x0
[15:14]	-	Rsvd	Reserved	0x0
[13:8]	p_lat_off	RW	Pulse Latch Offset When the Host Controller uses the pulse latch to sample the read data and response, users need to set the latch offset to correctly sample the value. The value set should be small than the SDCLK Frequency Select. 0x3F: Latch value at the 63rd sdclk1x rising edge after the SCLK edge …: … 0x01: Latch value at the 1st sdclk1x rising edge after the SCLK edge 0x00: Latch value at the SCLK edge	0x0
[7:1]	-	Rsvd	Reserved	0x0
0	p_lat_en	RW	Pulse Latch Enable This bit is used to select the read data and respond the	0x1

Vangotech

Bit	Name	Type	Description	Default Value
			sampling function. (1) Pulse Latching The host controller use the functions required for adjusting p_lat_off to correctly sample the values. Users should take this value into conditions with the round-chip effect to set p_lat_off. (2) Multiphase DLL Latching The host controller requires one extra analog IP to achieve the function. For the SDR104 speed, the function should be included by the hardware configuration. 1: Use the pulse latching function for the read data and response 0: Use the multiphase DLL latching for the read data and response	

5.2.6.33. Vendor-defined Register 1

Table 5-50 lists the bit assignments of the Vendor-defined Register 1.

Table 5-50. Vendor-defined Register 1 (VR1, Offset: 0X104)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0x0
24	cmd_conflict_en	RW	Command Conflict Checker Enable The host controller can check the CMD line conflict error. Users can set this bit to '1' to enable the checker.	0x0
[23:19]	-	Rsvd	Reserved	0x0
[18:16]	NSB	RW	NSB Timing of SD Specification The host controller is set to five SCLK clock cycles for the specification. Users can add the busy wait cycle if the SD card does not meet the specification.	0x0
[15:12]	-	Rsvd	Reserved	0x0
[11:8]	NCR	RW	NCR Timing of SD Specification The host controller is set to 64 SCLK clock cycles for the specification. Users can add the response wait cycle if the SD card does not meet the specification.	0x0
[7:4]	-	Rsvd	Reserved	0x0

Vangotech

Bit	Name	Type	Description	Default Value
3	Io_sd_rstn	RW	Output low-active reset signal In general, it is used to eMMC pin "RST_n".	0x0
2	mmc_boot_ack_en	RW	eMMC Booting Mode Acknowledge Enable The eMMC specification defines the Booting Acknowledge that can be disabled. Users can enable the function.	0x0
[1:0]	mmc_boot	RW	eMMC Booting Mode Selection The eMMC specification defines two Booting modes and Bus Test modes. Users need to set the function to achieve these modes. The read data in the boot mode can be read from the data port or transferred by SDMA. To exit from the boot mode, the "software reset for all" must be set after clearing this register to '0'. (When in the alternative boot mode, the CMD0 for terminating the boot mode should be issued after setting "software reset for all".) The eMMC version 4.3 does not support ultra-speed in boot modes, including SDR50, SDR104 and DDR50. 0b11 – eMMC Bus Test mode 0b10 – eMMC Alternative Boot mode 0b01 – eMMC Boot mode 0b00 – Normal mode	0x0

5.2.6.34. Vendor-defined Register 2

Table 5-51. Vendor-defined Register 2 (VR2, Offset: 0X108)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0x0
0	clk_ctrl_sw_rst	RWAC	Clock Control Software Reset Users can reset the clock control of the host controller. When this bit is set to '1', the clock control will reset to the default value.	0x0

5.2.6.35. Vendor-defined Register 3

Table 5-52. Vendor-defined Register 3 (VR3, Offset: 0X10C)

Bit	Name	Type	Description	Default Value
[31:29]	-	Rsvd	Reserved	0x0

Vangotech

[28:24]	sd_delay_sel_bound	RW	Repeating Times of Auto-tuning Operation The default value is 0x1F, which represents that a total of 32 times tuning pattern read will be executed. This value should be matched with the stage of the multiphase DLL.	0x1F
[23:21]	-	Rsvd	Reserved	0x0
[20:16]	sd_delay_val	RW	SD Delay Value Selection The output delay value of the host controller to the multiphase DLL for delay control. Users can set the value by itself or let the auto-tuning operation to find the best setting value.	0x0
[15:13]	-	Rsvd	Reserved	0x0
[12:8]	crc16_error_thres	RW	DATA CRC16 Error Threshold Value When the times of CRC16 error reaches this threshold value, the tuning_err_r status (Offset 0x32 bit 10) will be set.	0x8
[7:5]	-	Rsvd	Reserved	0x0
[4:0]	tuning_success_thres	RW	Auto-tuning Operation Success Times Threshold If the success time of the tuning pattern read operation is greater than this threshold, the auto-tuning operation will be successful. The value is suggested to be greater than 8 when the total times of the tuning pattern read operation is 32.	0x8

5.2.6.36. Vendor-defined Register 4

Table 5-53. Vendor-defined Register 4 (VR4, Offset: 0X110)

Bit	Name	Type	Description	Default Value
[31:0]	tuning_record	RO	Auto-tuning Operation Result The auto-tuning operation result can be checked in this register. Bit[x] 1: Tuning pattern read is correct when sd_delay_val is x. 0: Tuning pattern read is incorrect when sd_delay_val is x. (x = 0 ~ 31) sd_delay_val can be adjusted by checking the result in this register.	0x0

5.2.6.37. Vendor-defined Register 5

Table 5-54. Vendor-defined Register 5 (VR5, Offset: 0X114)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0x0
[3:0]	db_timeout	RW	Card Insertion De-bounce Cycle 0: 2^9 sys_hclk cycles 1: 2^{10} sys_hclk cycles 15: 2^{24} sys_hclk cycles	0x0

5.2.6.38. Vendor-defined Register 6

Table 5-55. Vendor-defined Register 6 (VR6, Offset: 0X118)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0x0
0	hburst_incr	RW	AHB Bus Burst Type 1: AHB master uses INCR as the AHB burst type. 0: AHB master uses SINGLE and INCR4 as the AHB burst type.	0x1

5.2.6.39. Vendor-defined Register 7

Table 5-56. Vendor-defined Register 7 (VR7, Offset: 0X11C)

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0x0
7	rsp_fifo_overrun	RW	Response FIFO overrun status	0x0
6	cmd_fifo_underrun	RW	Command FIFO underrun status	0x0
5	stop_sdclk_by_user	RW1C	Stop SD clock due to user set to stop status	0x0
4	stop_sdclk_insert_unstable	RW1C	Stop SD clock due to card detection is unstable status	0x0
3	stop_sdclk_wait_buf_ava_rd	RW1C	Stop SD clock due to data buffer is un-available status	0x0
2	stop_sdclk_wr_fifo_empty	RW1C	Stop SD clock due to write FIFO is empty status	0x0
1	stop_sdclk_rd_fifo_afull	RW1C	Stop SD clock due to read FIFO is almost full status	0x0

Vangotech

0	ahb_resp_err_sts	RW1C	AHB Master Response Error Status This bit is set when the AHB master receives an error type response. If ahb response error occurs, then it might be some unexpected interrupts like dma interrupt or adma error interrupt happened. User should clear all these error status and issue abort commands (if any data line is not in IDLE state).	0x0

5.2.6.40. Vendor-defined Register 8

Table 5-57. Vendor-defined Register 8 (VR8, Offset: 0X120)

Bit	Name	Type	Description	Default Value
31	sd_1V8_fail_en	RW	Input signal sd_1V8_en_fail Enable	0x0
[30:8]	-	Rsvd	Reserved	0x0
7	rsp_fifo_overrun_en	RW	Response FIFO overrun status Enable	0x0
6	cmd_fifo_underrun_en	RW	Command FIFO underrun status Enable	0x0
5	stop_sdclk_by_user_en	RW	Stop SD clock due to user set to stop status Enable	0x0
4	stop_sdclk_insert_unstable_en	RW	Stop SD clock due to card detection is unstable status Enable	0x0
3	stop_sdclk_wait_buf_ava_rd_en	RW	Stop SD clock due to data buffer is un-available status Enable	0x0
2	stop_sdclk_wr_fifo_empty_en	RW	Stop SD clock due to write FIFO is empty status Enable	0x0
1	stop_sdclk_rd_fifo_afull_en	RW	Stop SD clock due to read FIFO is almost full status Enable	0x0
0	ahb_resp_err_sts_en	RW	AHB Master Response Error Status Enable 1: Enable the AHB master response error status 0: Disable the AHB master response error status	0x0

5.2.6.41. Vendor-defined Register 9

Table 5-58. Vendor-defined Register 9 (VR9, Offset: 0X124)

Bit	Name	Type	Description	Default Value

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0x0
7	rsp_fifo_overrun_sig_en	RW	Response FIFO overrun status Signal Enable	0x0
6	cmd_fifo_underruen_sig_en	RW	Command FIFO underrun status Signal Enable	0x0
5	stop_sdclk_by_user_sig_en	RW	Stop SD clock due to user set to stop Signal Enable	0x0
4	stop_sdclk_insert_unstable_ sig_en	RW	Stop SD clock due to card detection is unstable Signal Enable	0x0
3	stop_sdclk_wait_buf_ava_rd_ sig_en	RW	Stop SD clock due to data buffer is un-available Signal Enable	0x0
2	stop_sdclk_wr_fifo_empty_ sig_en	RW	Stop SD clock due to write FIFO is empty Signal Enable	0x0
1	stop_sdclk_rd_fifo_afull_ sig_en	RW	Stop SD clock due to read FIFO is almost full Signal Enable	0x0
0	ahb_resp_err_sig_en	RW	AHB Master Response Error Signal Enable 1: Enable the interrupt generation when the AHB master response status is set. 0: Disable the interrupt generation when the AHB master response status is set.	0x0

5.2.6.42. DMA Handshake Enable Register

Table 5-59. DMA Handshake Enable Register (DHER, Offset: 0X128)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0x0
0	dma_hsk_en	RW	DMA Handshake Enable 1: Enable the DMA handshake protocol for the data port access 0: Disable the DMA handshake protocol for the data port access Note: When this register is set to '1', buf_r_rdy_r and buf_w_rdy_r will not be set and become ineffective. Data will be transferred by the external DMA and the flow control will be controlled by the DMA handshake protocol.	0x0

Vangotech

5.2.6.43. Hardware Attributes Register

Table 5-60. Hardware Attributes Register (HWA, Offset: 0X178)

Bit	Name	Type	Description	Default Value
[31:11]	-	Rsvd	Reserved	0x0
[10:0]	hw_config	RO	Bit 10 0: 1V8 fail enable reset value is 0. Bit 9 1: Read FIFO Depth is 8. Bit 8 1: ASYNC Bit 7 1: 8-bit SD data bus Bit 6 1: CPRM present Bit 5 1: DLL present Bit 4 1: DATA FIFO is 4K SRAM. Bit 3: Reserved Bit 2: Reserved Bit 1: Reserved Bit 0: Reserved	0x3F0

5.2.6.44. IP Revision Register

Table 5-61. IP Revision Register (IPRR, Offset: 0X17C)

Bit	Name	Type	Description	Default Value
[31:0]	Rev_num	RO	Revision number of SDC	0x010A00

5.2.6.45. Cipher Mode Control Register

This register is the configurable register for the CPRM function. When the CPRM function is used, this register will be used to select the mode of cipher function to encrypt or decrypt.

Table 5-62. Cipher Mode Control Register (CMCR, Offset: 0x180)

Bit	Name	Type	Description	Default Value
[31:11]	-	Rsvd	Reserved	0x0
10	swap_HL	RW	This bit is effective only in the auto_c2_ecbc_en mode and auto_c2_dcbc_en mode.	0x0

Vangotech

			When this bit is set in the auto_c2_ecbc_en mode: The high-word and low-word of the encrypted data will be swapped before being written to TX FIFO. When this bit is set in the auto_c2_dcbc_en mode: The high-word and low-word of the encrypted data will be swapped before decryption.	
8	sec_access_en	RW	Secret Constant Table Access Enable This bit must be enabled before writing or reading the secret constant table. Once this bit is enabled, the firmware will always access from the very beginning of the secret constant table.	0x0
7	auto_c2_dcbc_en	RW	Auto C2 Decryption with C-CBC Mode Enable In this mode, data will be automatically decrypted and sent to the 64-bit line buffer. FTSDC020 only supports the data lengths that are multiples of 8 bytes in this mode.	0x0
6	auto_c2_ecbc_en	RW	Auto C2 Encryption with C-CBC Mode Enable In this mode, data written to the 64-bit buffer will be automatically encrypted and sent to TX FIFO. FTSDC020 only supports data lengths which are the multiples of 8 bytes in this mode.	0x0
5	rngc2_g_en	RW	C2 Random Number Generator Enable	0x0
4	c2_dcbc_en	RW	C2 Decryption with C-CBC Mode Enable	0x0
3	c2_d_en	RW	C2 Decryption with ECB Mode Enable	0x0
2	c2_ecbc_en	RW	C2 Encryption with C-CBC Mode Enable	0x0
1	c2_e_en	RW	C2 Encryption with ECB Mode Enable	0x0
0	c2_g_en	RW	C2 One Way Function Enable	0x0

5.2.6.46. Cipher Mode Status Register

Table 5-63 lists the status of the Auto CMD12 Error Status register. When the auto_cmd12_en register is set to '1' and the Auto CMD12 error status register is set, the host driver will check this register to identify what kind of error happens during executing AUTO CMD12. This register will be valid only when **auto_cmd12_err_r** is set to '1'.

Table 5-63. Cipher Mode Status Register (CMSR, Offset: 0x184)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0x0
0	cp_rdy	RW1C	Cipher is ready. When this bit is set to '1', reading 0x19C and 0x1A0 will get cipher or plain text.	0x0

5.2.6.47. Cipher Mode Signal Enable Register

Table 5-64 lists the bit assignment of the Cipher Mode Signal Enable register. This register is used to select which interrupt status is notified to the host system as the interrupt. These interrupt statuses share the same interrupt line.

Table 5-64. Cipher Mode Signal Enable Register (CMSE, Offset: 0x188)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0x0
0	cp_rdy_sig_en	RW	Cipher Ready Signal Enable	0x0

5.2.6.48. Low Word of Input Data Register

Each encryption/decryption requires two 64-bit inputs, key and data, and generates one 64-bit output data. Ports 0x18C and 0x190 are for the data input. 0x194 and 0x198 are for the key input. 0x19C and 0x1A0 are for the data output.

Table 5-65. Low Word of Input Data Register (LWID, Offset: 0x18C)

Bit	Name	Type	Description	Default Value
[31:0]	R_r	RW	Input port for the input data bits[31:0]	0x0

5.2.6.49. High Word of Input Data Register

Table 5-66. High Word of Input Data Register (HWID, Offset: 0x190)

Bit	Name	Type	Description	Default Value
[31:0]	L_r	RW	Input port for the input data bits[63:32]	0x0

5.2.6.50. Low Word of Input Key Register

Table 5-67. Low Word of Input Key Register (LWIK, Offset: 0x194)

Bit	Name	Type	Description	Default Value
[31:0]	key_b_r	RW	Input port for the input key bits[31:0]	0x0

5.2.6.51. High Word of Input Key Register

Table 5-68. High Word of Input Key Register (HWIK, Offset: 0x198)

Bit	Name	Type	Description	Default Value
[23:0]	key_a_r	RW	Input port for the input key bits[55:32]	0x0

5.2.6.52. Low Word of Output Data Register

Table 5-69. Low Word of Output Data Register (LWOD, Offset: 0x19C)

Bit	Name	Type	Description	Default

Vangotech

				Value
[31:0]	cp_out_low_r	RO	Output port for the output data bits[31:0]	0x0

5.2.6.53. High Word of Output Data Register

Table 5-70. High Word of Output Data Register (HWOD, Offset: 0x1A0)

Bit	Name	Type	Description	Default Value
[31:0]	cp_out_hi_r	RO	Output port for the output data bits[63:32]	0x0

5.2.6.54. Secret Constant Table Data Port

CPRM requires the 256-byte secret constant released from 4C entity. To initialize the secret constant table, the firmware should set **sec_access_en** (0x180 bit 8) first and then send continual 256-byte data by writing 0x1A4 256 times. In the byte access, only 0x1A4 can be accessed. Accessing 0x1A5, 0x1A6, and 0x1A7 are impermissible.

Table 5-71. Secret Constant Table Data Port (SCTDP, Offset: 0x1A4)

Bit	Name	Type	Description	Default Value
[7:0]	sec_data_port	RW	Secret constant table data port	0x0

Vangotech

Chapter 6. Display

This chapter provides a detailed description of the display feature of V8130L processor from following aspects.

Vangotech

6.1. LCDC

6.1.1. Overview

- **TFT panel interface**
  - Pixel clock rate of up to 150 MHz
  - RGB parallel output: 18-bit or 24-bit bus interface
  - Swaps between RGB and BGR
  - Programmable resolution of up to 2048x2048
  - Programmable polarity/duration for output enable, vertical sync., horizontal sync., and pixel clock
  - Data/Synchronization on/off controls
- **Image input format**
  - RGB 12 (4:4:4)/16 (5:6:5)/15 (5:5:5)/24 (8:8:8) bits per pixel
  - Palette (8-bit, 4-bit, 2-bit, and 1-bit pixels)
  - YcbCr422 (16-bit per pixel)
  - YcbCr420 (Separate memory locations)
  - Little-endian, big-endian
- **Palette**
  - 256 tries 16-bit RGB color palette RAM
- **Color management**
  - Programmable contrast, brightness, sharpness, saturation, and hue controls
- **Three-channel gamma correction**
- **Dithering**
- **Picture-in-Picture (PiP)**
  - Displays a maximum of three PiP windows
  - Resolution of PiP window similar to main window
  - Supports 8-bit global blending levels
  - Supports 8-bit ARGB (8888) for each picture
  - Supports 1-bit ARGB (1555) for each picture
  - Supports color-key
- **Picture-out-Picture (PoP)**
  - Displays 4-in-1 PoP window
- **Interrupt control**
  - Master bus error
  - Frame status
  - FIFO under-run
  - Memory base update
- **Complex OSD (On-Screen Display)**
  - RAM-based font
    - User definable font
    - Font size: 12× 18
    - Supports a maximum of 256 Standard Fonts (SF)

Vangotech

Supports a maximum of 85 Multi-Color Fonts (MCF)

Border/Shadow effect

Zoom-in size: × 2, × 3, and × 4

Zoom-out size: × 0.5

OSD windows

Supports up to four windows

Flexible OSD window size: Up to 64 fonts on each vertical or horizontal within a range of 512 fonts

Transparency level: 25%, 50%, 75%, and 100%

Programmable window shadow color/size and border color/size

Font-position-based horizontal and vertical positions of a window

Shares 16-entry color palette for foreground, background, shadow, and border color

Supports special effects of all fonts in OSD window, such as transparency, zoom-in/zoom-out, and shadow/border, sharing same setting

Zoom-in/Zoom-out fonts to be configured in horizontal and vertical directions

- **Simple OSD**

  RAM-based programmable size: 12x16

  Font variety of up to 256

  Maximal window fonts: 512

  Supports flexible dimensions for both horizontal and vertical directions

  Programmable window position

  Supports up-scaling function with factors of 1, 2, 3, and 4

  Transparency level: 25%, 50%, 75%, and 100%

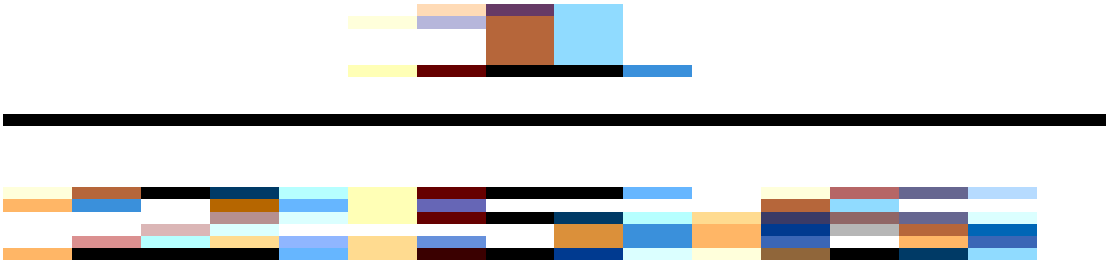
  4-entry and 8-bit color palette for foreground

  3-entry and 8-bit color palette for background (One entry reserved for transparency)

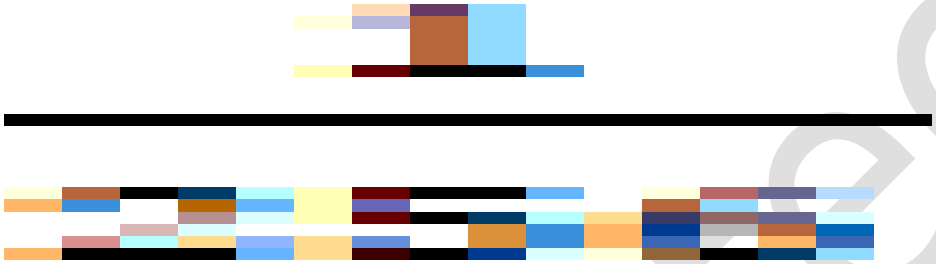
- **Scaler** (Does not support TV interlace input or interlace output)

Vangotech

Down-scaling ration ranging from



x



to 1x1

Arbitrary ration between 1x1 and 2x2 for up-scaling stage

Three interpolation modes for up-scaling: Nearly bilinear interpolation, threshold nearly bilinear interpolation, and most neighborhood interpolation

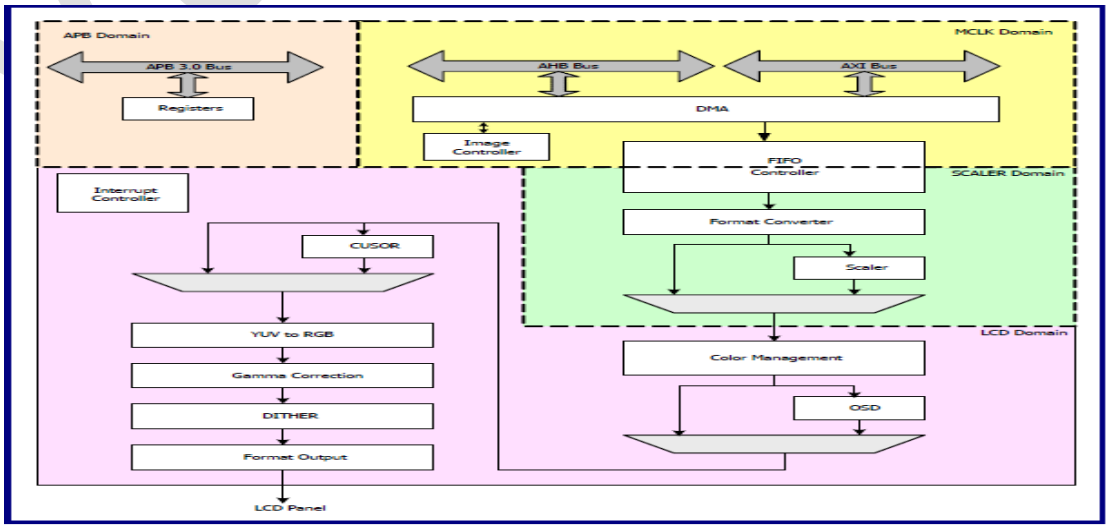
- **Hardware cursor**

  Supports 64x64 hardware cursor

- **Others**

  Programmable color bar generator

6.1.2. Block Diagram



Vangotech

Figure 6-1. Block Diagram

6.1.3. Operations and Functional Descriptions

6.1.3.1. Application Connection

Figure 6-2 illustrates an example of LCDC applications. The LCD controller first reads the image data from SDRAM via the SDRAM controller to the TFT display panel. The CPU controls the entire system, including the programming registers of each controller and updates the frame buffer content of the SDRAM.

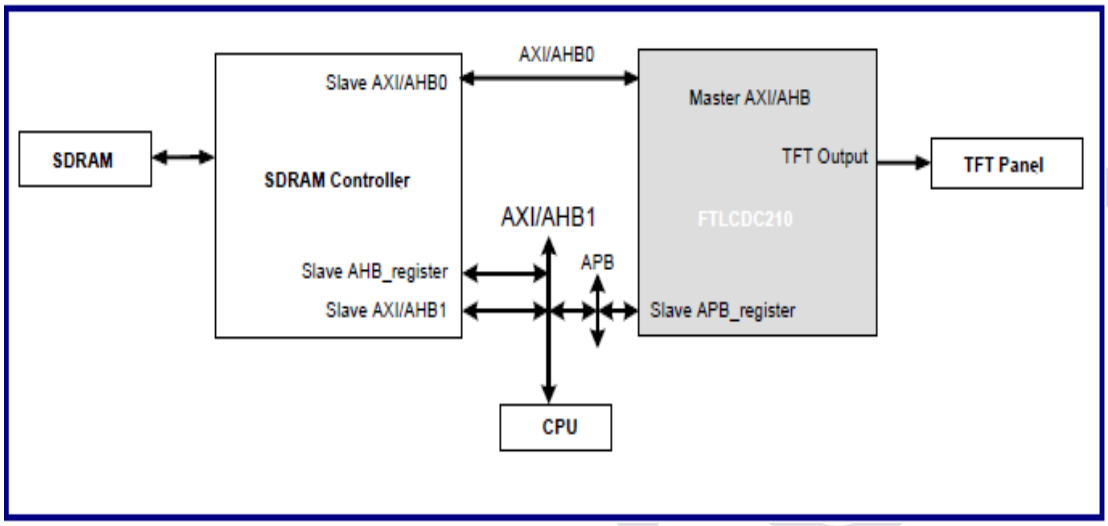


Figure 6-2. Example of LCDC Application

6.1.4. Register List

Module Name	Base Address
LCDC	0x5620_0000

Name	Address	Type	Width	Description	Reset Value
lcd_function_en	0x000	R/W	20	LCD function enable	0x00
lcd_panel_pixel	0x004	R/W	21	LCD panel pixel parameter	0x00
lcd_intr_mask	0x008	R/W	4	LCD interrupt enable mask	0x00
lcd_intr_clr	0x00C	W	4	LCD interrupt status clear	0x00
lcd_intr_status	0x010	R	4	LCD interrupt status	0x00
lcd_frame_buffer	0x014	R/W	16	LCD frame buffer parameter	0x00
lcd_im0frm0base	0x018	R/W	30	LCD panel image0 frame0 base address	0x00
lcd_im1frm0base	0x024	R/W	30	LCD panel image1 frame0 base address	0x00
lcd_im2frm0base	0x030	R/W	30	LCD panel image2 frame0 base address	0x00
lcd_im3frm0base	0x03C	R/W	30	LCD panel image3 frame0 base address	0x00
lcd_patbarcolor	0x048	R/W	8	LCD pattern generator	0x00
lcd_fifoth	0x04C	R/W	32	LCD FIFO threshold	0x04040404
lcd_bandwithCtrl	0x050	R/W	10	LCD bandwidth control	0x200

-	0x058	R	-	LCD revision	-
lcd_vbibase	0x05C	R/W	30	LCD panel image4 frame0 base address	0x00
lcd_fifo4th	0x060	R/W	8	LCD FIFO threshold	0x04
lcd_hortiming	0x100	R/W	32	LCD horizontal timing control	0x00
lcd_vertiming0	0x104	R/W	32	LCD vertical timing control0	0x00
lcd_vertiming1	0x108	R/W	8	LCD vertical timing control1	0x00
lcd_polarity	0x10C	R/W	23	LCD polarity control	0x00
lcd_pipblend	0x300	R/W	24	LCD PIP parameters	0x00
lcd_pip1pos	0x304	R/W	29	PiP sub-picture1 position	0x00
lcd_pip1dim	0x308	R/W	27	PiP sub-picture1 dimension	0x00
lcd_pip2pos	0x30C	R/W	27	PiP sub-picture2 position	0x00
lcd_pip2dim	0x310	R/W	27	PiP sub-picture2 dimension	0x00
lcd_pipriority	0x314	R/W	8	PiP image priority	0xE4
lcd_param1	0x318	R/W	32	PiP/PoP image format 1	0x00
lcd_param2	0x31C	R/W	6	PiIP/PoP image format 2	0x00
lcd_colorkey1	0x320	R/W	25	PiP color key1	0x00
lcd_colorkey2	0x324	R/W	25	PiP color key2	0x00
lcd_colorkey3	0x328	R/W	25	PiP color key3	0x00
lcd_pip3pos	0x32C	R/W	27	PiP sub-picture3 position	0x00
lcd_pip3dim	0x330	R/W	27	PiP sub-picture3 dimension	0x00
lcd_colorman0	0x400	R/W	14	LCD color management parameter 0	0x2000
lcd_colorman1	0x404	R/W	15	LCD color management parameter 1	0x2000
lcd_colorman2	0x408	R/W	24	LCD color management parameter 2	0x00
lcd_colorman3	0x40C	R/W	21	LCD color management parameter 3	0x40000
lcd_GaRdLUT	0x600 ~0x6FC	R/W	32	LCD gamma red look-up table	-
lcd_GaGnLUT	0x700 ~0x7FC	R/W	32	LCD gamma green look-up table	-
lcd_GaBuLUT	0x800 ~0x8FC	R/W	32	LCD gamma blue look-up table	-
PaletteWritePort	0xA00 ~0xBFC	W	32	LCD palette RAM write accessing port	-
scal_hor_no_in	0x1100	R/W	12	Scaler horizontal resolution of input	0x00
scal_ver_no_in	0x1104	R/W	12	Scaler vertical resolution of input	0x00
scal_hor_no_out	0x1108	R/W	14	Scaler horizontal resolution of	0x00

Vangotech

				output	
scal_ver_no_out	0x110C	R/W	14	Scaler vertical resolution of output	0x00
scal_misc	0x1110	R/W	9	Scaler miscellaneous control registers	0x00
scal_hor_high_th	0x1114	R/W	9	Horizontal up-scaling high threshold	0x00
scal_hor_low_th	0x1118	R/W	9	Horizontal up-scaling low threshold	0x00
scal_ver_high_th	0x111C	R/W	9	Vertical up-scaling high threshold	0x00
scal_ver_low_th	0x1120	R/W	9	Vertical up-scaling low threshold	0x00
scal_hor_ver_num	0x112C	R/W	16	Scaler resolution parameter	0x00
cursor_position	0x1200	R/W	30	Cursor position parameter	0x00
cur_color1 ~ cur_color15	0x1204 ~0x123C	R/W	24	Cursor color parameter 1 ~ 15	0x00
cur_font	0x1300 ~0x13FC	R/W	32	32 * 32 cursor font ram	-
cur_font	0x1600 ~0x1DFC	R/W	32	64 * 64 cursor font ram	-
vs_control	0x1500	R/W	21	Virtual screen control	0x00
vs1_control	0x1504	R/W	21	Virtual screen1 control	0x00
vs2_control	0x1508	R/W	21	Virtual screen2 control	0x00
vs3_control	0x150C	R/W	21	Virtual screen3 control	0x00
osd_windowOn	0x2000	R/W	4	OSD windows on/off	0x00
osd_fontctrl0	0x2004	R/W	24	OSD font control register 0	0x00
osd_fontctrl1	0x2008	R/W	13	OSD font control register 1	0x00
osd_window1ctrl0	0x200C	R/W	16	OSD window1 control 0	0x00

6.1.5. Register Descriptions

6.1.5.1. LCD Function Enable Parameter

Table 6-1. LCD Function Enable Parameter (lcd_function_en, Offset 0x0000)

Bit	Name	Type	Description	Default Value
19	Deflicker_En	R/W	Deflicker enable When this bit set to '1', LCDC will perform the deflicker function to replace the original sharpness function. 0:Disable the deflicker function. 1:Enable the deflicker function.	0x0
18	Reserved	-	-	
17	Double_En	R/W	Double word command enable	0x0

Vangotech

			When this bit set to '1', the DMA will use the double word command (Eight bytes) to catch the image data from frame buffer. 0:Disable the double word access 1:Enable the double word access	
16	AddrSyn_En	R/W	Address sync. enable When this bit sets to '1', the initial address of each DMA buffer can be updated to new address with the "AddrUpdate" bit set to '1'. When user wants to update the base address of all DMA buffers at same time, this bit must be set to enable the sync. address update. 0:Disable the address synchronous 1:Enable the address synchronous	0x0
15	VBI_En	R/W	VBI enable This bit informs LCDC to insert the VBI information from the DMA buffer4 to the vertical blanking lines. This bit is effective when TVEn is set to '1'. 0:Disable VBI 1:Enable VBI Note: If VBI is enabled, the OSD and cursor function will be disabled.	0x0
14	PatGen	R/W	Test pattern generator When the pattern generator is enabled, LCDC will automatically send out the patterns to the panel depending on the programmed value of 0x48. 0:Turn-off pattern generator 1:Turn-on pattern generator	0x0
13	TVEn	R/W	TV output function When this bit is set, LCDC supports the TV output. 0:Disable the TV output 1:Enable the TV output	0x0
12	CursorEn	R/W	Hardware cursor When this bit is set, LCDC supports the hardware cursor function. 0:Disable the hardware cursor 1:Enable the hardware cursor	0x0
[11:10]	PiPEn	R/W	Picture-in-Picture (PiP) mode When PiPEn is set, LCDC can support the PiP function. The PiP function can support windows from different frame buffers. 00:Disable PiP 01:Single PiP window	0x0

Vangotech

			10:Double PiP window 11:Triple PiP window	
[9:8]	BlendEn	R/W	Alpha Blending enable When this bit is set, the PiP function can support the Alpah Blending function. 00=Disable the Alpah Blending function 01=Enable the global Alpha Blending function. The blending levels are depending on the "PiPBlend" register. 10=Enable the pixel Alpha Blending function. The format of the input images must be ARGB8888 or ARGB1555. The blending levels are depending on the "A" component of the input data. 11=Reserved	0x0
7	PoPEn	R/W	Picture-out-Picture (PoP) mode When this bit is set, LCDC can support the PoP function. The PoP function can only be divided into four quarters in one screen. 0:Disable the PoP function 1:Enable the PoP function	0x0
6	DitherEn	R/W	Dither enable When this bit is set, LCDC can support the dithering function. The dithering type is depending on the value of DitherType. 0:Disable the dithering function 1:Enable the dithering function	0x0
5	ScalerEn	R/W	Scaler enable When this bit is set, LCDC can support the scaling function. The detailed scaling dimension is depending on the scaler control registers. 0:Disable the scaling function 1:Enable the scaling function	0x0
4	OSDEn	R/W	OSD enable When this bit is set, LCDC can support the On-Screen Display (OSD) function. The displaying type of OSD is depending on the programming OSD control register. 0:Disable the On-Screen Display (OSD) 1:Enable the On-Screen Display (OSD)	0x0
3	EnYCbCr	R/W	YcbCr input mode control When this bit is set, it informs LCDC that the data type in the frame buffer is the YcbCr format. 0:Enable the RGB format 1:Enable the YcbCr format	0x0

Vangotech

2	EnYCbCr420	R/W	YcbCr420 input mode control When this bit is set, it informs LCDC that the data type in the frame buffer is YcbCr420 format. 0:Enable the YcbCr422 input mode 1:Enable the YcbCr420 input mode EnYCbCr must set to '1' to apply these functions.	0x0
1	LCDon	R/W	LCD screen on/off control 0:Disable the LCD screen (All data output pins are forced to '0'.) 1:Enable the LCD screen (Normal operation)	0x0
0	LCDen	R/W	LCD controller enable control 0:Disable the LCD controller and force all LCD signals, including synchronization and the overall data, to zero. 1:Enable the LCD controller (Normal operation)	0x0

6.1.5.2. LCD Panel Pixel Parameter

Table 6-2. LCD Panel Pixel Parameter (lcd_panel_pixel, Offset 0x0004)

Bit	Name	Type	Description	Default Value
20	TVRST	R/W	TV_CLK domain reset The signal can reset all status in the TV clock domain. This register can only be cleared by users.	0x0
19	PRST	R/W	PCLK domain reset The signal can reset all status except the configuration parameters in the APB clock domain. This register can only be cleared by users.	0x0
18	LRST	R/W	LC_CLK domain reset The signal can reset all status in the LCD clock domain. This register can only be cleared by users.	0x0
17	MRST	R/W	Master domain reset (AXI or AHB) The signal can reset all status in the AXI/AHB domain. This register can only be cleared by users.	0x0
16	AddrUpdate	R/W	Address update This bit is valid when AddrSyn_En (0000H) is set. When AddrUpdate is set and AddrSyn_En = '1', the base address of each DMA buffer can be updated.	0x0
[15:14]	UpdateSrc	R/W	Update source selection This bit, IntNxtBase, is used to specify the image sources to choose from: 00: Image0 01: Image1 10: Image2	0x0

Vangotech

			11: Image3	
[13:12]	DitherType	R/W	Dithering type This value decides which dithering algorithm can be used. The value is valid when DitherEn is set. 00: Transfer 888 to 565 01: Transfer 888 to 555 10: Transfer 888 to 444 11: Reserved	0x0
11	PanelType	R/W	TFT panel color depth selection This value chooses the color depth of the output format. 0: 6 bits per channel with a 18-bit panel interface 1: 8 bits per channel with a 24-bit panel interface	0x0
[10:9]	Vcomp	R/W	Generate the vertical status interrupt: 00: Start of the vertical sync. 01: Start of the vertical back porch 10: Start of the vertical active image 11: Start of the vertical front porch	0x0
[8:7]	RGBTYPE	R/W	RGB input format This bit identifies the input RGB format when BppFifo = '100'. 00: RGB 565 input 01: RGB 555 input 10: RGB 444 input 11: Reserved	0x0
[6:5]	Endian	RW	Frame buffer data endianness control 00: Little-endian byte little-endian pixel 01: Big-endian byte big-endian pixel 10: Reserved 11: Reserved	0x0
4	BGRSW	R/W	RGB or BGR output format selection 0: RGB normal output 1: BGR red and blue swapped	0x0
3	PWROFF	R/W	This bit directly rives the output pin, LC_PWROFF.	0x0
[2:0]	BppFifo	R/W	Pixel format in FIFO The value informs the data format in each DMA buffer (Bits per pixel). 000: 1 bpp 001: 2 bpp 010: 4 bpp 011: 8 bpp 100: 16 bpp	0x0

Vangotech

			101: 24 bpp Others: Reserved	

6.1.5.3. LCD Interrupt Enable Mask Parameter

Table 6-3. LCD Interrupt Enable Mask Parameter (lcd_intr_mask, Offset 0x0008)

Bit	Name	Type	Description	Default Value
3	IntBusErrEn	R/W	AXI master error interrupt enable This bit is set to enable the bus error interrupt. 1: Enable 0: Disable	0x0
2	IntVstatusEn	R/W	Vertical duration comparison interrupt enable This bit is set to enable the vertical status interrupt. 1: Enable 0: Disable	0x0
1	IntNxtBaseEn	R/W	Next frame base address updated interrupt enable This bit is set to enable the successfully base address updated interrupt. 1: Enable 0: Disable	0x0
0	IntFIFOUdnEn	R/W	FIFO under-run interrupt enable This bit is set to enable the FIFO under-run interrupt. 1: Enable 0: Disable	0x0

6.1.5.4. LCD Interrupt Status Clear Parameter

Table 6-4. LCD Interrupt Status Clear Parameter(lcd_intr_clr, Offset 0x000C)

Bit	Name	Type	Description	Default Value
3	ClrBusErr	W	Setting this bit to '1' clears the AXI master error interrupt status. Setting this bit to '0' has no effect.	0x0
2	ClrVstatus	W	Setting this bit to '1' clears the interrupt status of vertical duration comparison. Setting this bit to '0' has no effect.	0x0
1	ClrNxtBase	W	Setting this bit to '1' clears the frame buffer base address update interrupt status. Setting this bit to '0' has no effect.	0x0
0	ClrFIFOUdn	W	Setting this bit to '1' clears the FIFO under-run interrupt status. Setting this bit to '0' has no effect.	0x0

Vangotech

6.1.5.5. LCD Interrupt Status Parameter

Table 6-5. LCD Interrupt Status Parameter(lcd_intr_status, Offset 0x0010)

Bit	Name	Type	Description	Default Value
3	IntBusErr	R	AXI master error status This bit is set when the AMBA AXI master encounters a bus error response from a slave.	0x0
2	IntVstatus	R	Vertical comparison When this bit is set, one of the four vertical durations is reached. This duration can be selected via the LCDControl registers.	0x0
1	IntNxtBase	R	Frame buffer base address update When this bit is set, the current base address registers are successfully updated by the next address registers.	0x0
0	IntFIFOUdn	R	FIFO under-run When this bit is set, the DMA FIFO is a read access for an under-run condition caused by empty.	0x0

6.1.5.6. Frame Buffer Parameter

Please note that the scaling down function only supports 16 bpp or above.

Table 6-6. Frame Buffer Parameter (lcd_frame_buffer, Offset 0x0014)

Bit	Name	Type	Description	Default Value
[15:14]	Im3ScalDown	R/W	Scaling down for image3 The register is valid when PoP enable without PiP. The image from "LCDImage3FrameBase" can be scaled down depending on the value. 00: Disable 01: Image3 will be scaling down to 1/2 x 1/2. 10: Image3 will be scaling down to 1/2 x 1. 11: Reserved Please note that this register field can only be set to '00' under the following conditions: ● VirtualScreenEn or LCM_En is set.	0x0
[13:12]	Im2ScalDown	R/W	Scaling down for image2 The register is valid when PoP enable without PiP. The image from "LCDImage2FrameBase" can be scaled down depending on the value. 00: Disable 01: Image2 will be scaling down to 1/2 x 1/2.	0x0

Vangotech

Bit	Name	Type	Description	Default Value
			10: Image2 will be scaling down to 1/2 x 1. 11: Reserved Please note that this register field can only be set to '00' under the following conditions: ● VirtualScreenEn or LCM_En is set.	
[11:10]	Im1ScalDown	R/W	Scaling down for image1 The register is valid when PoP enable without PiP. The image from "LCDImage1FrameBase" can be scaled down depending on the value. 00: Disable 01: Image1 will be scaling down to 1/2 x 1/2. 10: Image1 will be scaling down to 1/2 x 1. 11: Reserved Please note that this register field can only be set to '00' under the following conditions: ● VirtualScreenEn or LCM_En is set.	0x0
[9:8]	Im0ScalDown	R/W	Scaling down for image0 The register is valid when PoP enable without PiP. The image from "LCDImage0FrameBase" can be scaled down depending on the value. 00: Disable 01: Image0 will be scaling down to 1/2 x 1/2. 10: Image0 will be scaling down to 1/2 x 1. 11: Reserved Please note that this register field can only be set to '00' under the following conditions: ● VirtualScreenEn or LCM_En is set.	0x0
[7:0]	Reserved	-	-	

6.1.5.7. LCD Panel Image0 Frame0 Base Address

Table 6-7. Image0 Frame0 Base Address (lcd_im0frm0base, Offset 0x0018)

Bit	Name	Type	Description	Default Value
[31:2]	LCDImage0FrameBase	R/W	LCD frame0 base address of image0 This is the start address of the frame data in memory. The two LSB bits are defaulted to 0. The total occupied memory address range is: Starting address: {LCDImageFrame0Base, 0, 0}	0x0000

Vangotech

			If the system is AXI bus, the base address must be set as 4-kbyte alignment.	
[1:0]	-	-	-	-

6.1.5.8. LCD Panel Image1 Frame0 Base Address

Table 6-8. Image1 Frame0 Base Address(lcd_im1frm0base, Offset 0x0024)

Bit	Name	Type	Description	Default Value
[31:2]	LCDImage1FrameBase	R/W	LCD frame0 base address of image1 This is the start address of the frame data in memory. The two LSB bits are defaulted to 0. The total occupied memory address range is: Starting address: {LCDImage1Frame0Base, 0, 0} If the system is AXI bus, the base address must be set as 4-kbyte alignment.	0x0000
[1:0]	-	-	-	-

6.1.5.9. LCD Panel Image2 Frame0 Base Address

Table 6-9. Image2 Frame0 Base Address(lcd_im2frm0base, Offset 0x0030)

Bit	Name	Type	Description	Default value
[31:2]	LCDImage2FrameBase	R/W	LCD frame0 base address of image2 This is the start address of the frame data in memory. The two LSB bits are default 0. The total occupied memory address range is: Starting address: {LCDImage2Frame0Base, 0, 0} If the system is AXI bus, the base address must be set as 4-kbyte alignment.	0x0000

6.1.5.10. LCD Panel Image3 Frame0 Base Address

Table 6-10. Image3 Frame0 Base Address(lcd_im3frm0base, Offset 0x003C)

Bit	Name	Type	Description	Default Value
[31:2]	LCDImage3FrameBase	R/W	LCD frame0 base address of image3 This is the start address of the frame data in memory. The defaults of two LSB bits are 0s. The total occupied memory address range is: Starting address: {LCDImage3Frame0Base, 0, 0} If the system is AXI bus, the base address must be	0x0000

Vangotech

			set as 4-kbyte alignment.	

6.1.5.1. PatGen Pattern Bar Distance Parameter

Table 6-11. PatGen Pattern Bar Distance Parameter (lcd_patbarcolor Offset 0x0048)

Bit	Name	Type	Description	Default Value
[31:8]	-	-	Reserved	0x0
[7:6]	Img3PatGen	R/W	Pattern generator of image 3 LCDC will generate the different test patterns of image3, depending on the value. 00: Vertical color bar 01: Horizontal color bar 10: Single color 11: Reserved	0x0
[5:4]	Img2PatGen	R/W	Pattern generator of image2 LCDC will generate the different test patterns of image2, depending on the value. 00: Vertical color bar 01: Horizontal color bar 10: Single color 11: Reserved	0x0
[3:2]	Img1PatGen	R/W	Pattern generator of image1 LCDC will generate the different test patterns of image1, depending on the value. 00: Vertical color bar 01: Horizontal color bar 10: Single color 11: Reserved	0x0
[1:0]	Img0PatGen	R/W	Pattern Generator of image0 LCDC will generate the different test patterns of image0, depending on the value. 00: Vertical color bar 01: Horizontal color bar 10: Single color 11: Reserved	0x0

6.1.5.2. FIFO Threshold Control Parameter

Table 6-12. FIFO Threshold Control Parameter (lcd_fifoth, Offset 0x004C)

Bit	Name	Type	Description	Default value
[31:24]	Buf3Threshold	R/W	DMA buffer 3 threshold	0x04

Vangotech

			The buffer 3 will assert request for master to fetch data in when the empty count of buffer 3 is larger than this value. This threshold must be larger or equal to 4.	
[23:16]	Buf2Threshold	R/W	DMA buffer 2 threshold The buffer 2 will assert request for master to fetch data in when the empty count of buffer 2 is larger than this value. This threshold must be larger or equal to 4.	0x04
[15:8]	Buf1Threshold	R/W	DMA buffer 1 threshold The buffer 1 will assert request for master to fetch data in when the empty count of buffer 1 is larger than this value. This threshold must be larger or equal to 4.	0x04
[7:0]	Buf0Threshold	R/W	DMA buffer 0 threshold The buffer 0 will assert request for master to fetch data in when the empty count of buffer 0 is larger than this value. This threshold must be larger or equal to 4.	0x04

6.1.5.3. Bus Bandwidth Control Parameter

If users turn on the VBI Enable, VBI FIFO will receive the highest priority to use the bus.

Table 6-13. Bus Bandwidth Control Parameter (lcd_bandwithCtrl, Offset 0x0050)

Bit	Name	Type	Description	Default Value
9	Lock_En	R/W	Enable the LOCK command When this bit is set to '1', LCDC will use LOCK command to request transfer once the image data in LCDC is going empty. The default setting is '1'. 0: Disable the LOCK command 1: Enable the LOCK command	0x1
8	Ratio_En	R/W	Enable the bus bandwidth ratio When this bit is set to '1', LCDC will use the round robin with the bandwidth ratio to arbitrate between different frame buffers. When this bit is set to '0', LCDC only uses the round robin to perform the arbitration between image frame buffers. 0: Disable the bus bandwidth ratio 1: Enable the bus bandwidth ratio	0x0
[7:6]	BwctrlF3	R/W	Bus bandwidth control ratio for the Image3 Frame buffer When Ratio_En is set, DMA of LCDC will use this ratio to catch data for image0. For example, BwctrlF0 = Ratio4, BwctrlF1 = Ratio 2, BwctrlF2 = Ratio1, and BwctrlF3 = Ratio 1. Therefore, the bandwidth ratio for four frame buffers is 4:2:1:1. The sequence of bus accessing is F0 F1 F2 F3 F0 F1 F0 F0. 00: Ratio 1 01: Ratio 2	0x0

			10: Ratio 4	
[5:4]	BwctrlF2	R/W	Bus bandwidth control ratio for Image2 Frame buffer When Ratio_En is set, DMA of LCDC will use this ratio to catch data for image1. 00: Ratio 1 01: Ratio 2 10: Ratio 4	0x0
[3:2]	BwctrlF1	R/W	Bus bandwidth control ratio for Image1 Frame buffer When Ratio_En is set, DMA of LCDC will use this ratio to catch data for image2. 00: Ratio 1 01: Ratio 2 10: Ratio 4	0x0
[1:0]	BwctrlF0	R/W	Bus bandwidth control ratio for Image0 Frame buffer When Ratio_En is set, DMA of LCDC will use this ratio to catch data for image3. 00: Ratio 1 01: Ratio 2 10: Ratio 4	0x0

6.1.5.4. Revision Register

Table 6-14. Revision Register(lcd_revision, Offset: 0x58)

Bit	Name	Type	Description	Default Value
[31:24]	-	-	Reserved	
[23:16]	MAJOR_REV	RO	Major revision number	0x01
[15:8]	MINOR_REV	RO	Minor revision number	0x09
[7:0]	REL_REV	RO	Release number	0x00

6.1.5.1. LCD Panel VBI Base Address

Table 6-15. VBI Base Address (lcd_vbibase, Offset 0x005C)

Bit	Name	Type	Description	Default Value
[31:2]	LCDVBIBase	R/W	LCD VBI base address This is the start address of the VBI data in memory. The defaults of two LSB bits are 0s. The total occupied memory address range is: Starting address: {LCDVBIBase, 0, 0} If the system is AXI bus, base address must be set as 4-kbyte alignment. Note: This base address is used for VBI when the VBI is enabled.	0x0000

Vangotech

6.1.5.2. FIFO Threshold Control Parameter

Table 6-16. FIFO Threshold Control Parameter (lcd_fifo4th, Offset 0x0060)

Bit	Name	Type	Description	Default Value
[7:0]	Buf4Threshold	R/W	DMA buffer 4 threshold The buffer 4 will assert the request for master to fetch data in when the empty count of buffer4 (VBI FIFO) is larger than this value. This threshold must be larger than or equal to 4.	0x04

6.1.5.3. LCD Horizontal Timing Control Parameter

Table 6-17. LCD Horizontal Timing Control Parameter (lcd_hortiming, Offset 0x0100)

Bit	Name	Type	Description	Default value
[31:24]	HBP	R/W	Horizontal Back Porch Actual horizontal back porch = HBP + 1 HBP is the number of LC_PCLK periods between the LC_HS falling edge and the start of active data. The 8-bit HBP field is used to specify the number of the pixel clock periods inserted at the beginning of each line or row of pixels. After the line clock for the previous line has been de-asserted, the value in HBP is used to count the number of pixel clocks to wait before starting the next display line. HBP can generate a delay of 1 to 256 pixel clock cycles.	0x00
[23:16]	HFP	R/W	Horizontal Front Porch Actual horizontal front porch = HFP + 1 HFP is the number of LC_PCLK periods between the end of active data and the LC_HS rising edge. The 8-bit HFP field sets the number of the pixel clock intervals at the end of each line or row of pixels, before the LCD line clock is pulsed. Once a complete line of pixels is transmitted to the LCD driver, the value in HFP is used to count the number of pixel clocks to wait before asserting the line clock. HFP can generate a period of 1 to 256 pixel clock cycles.	0x00
[15:8]	HW	R/W	Horizontal Synchronization Pulses Width Actual horizontal synchronization pulses width = HW + 1 HW is the width of the LC_HS signal in LC_PCLK periods. The 8-bit HW field specifies the pulse width of the line clock in the passive mode, or the horizontal synchronization pulse in the active mode.	0x00
[7:0]	PL	R/W	Pixels-per-Line Actual pixels-per-line = 16 * (PL + 1)	0x00

Vangotech

			The PL bit field specifies the number of pixels in each line of row of the screen. PL is a 6-bit value that represents between 16 and 2048 PL. PL is used to count the number of pixel clock that occur before the HFP is applied (Program the value required divided by 16, minus 1). User must ensure that each line start is double-word alignment and 16 * (PL + 1) * bpp is multiple of 128-bit if 64-bit AXI bus is used.	

6.1.5.4. LCD Vertical Timing Control Parameter

Table 6-18. LCD Vertical Timing Control Parameter (lcd_vertiming0, Offset 0x0104)

Bit	Name	Type	Description	Default Value
[31:24]	VFP	R/W	Vertical Front Porch Actual vertical front porch = VFP VFP is the number of the inactive lines at the end of frame before vertical synchronization period. The 8-bit VFP field is used to specify the number of the line clocks to insert at the end of each frame. Once a complete frame of pixels is transmitted to the LCD display, the value in VFP is used to count the number of line clock periods to wait. After the count has elapsed the vertical synchronization (LC_VS) signal is asserted. VFP generates from 0 to 255-line clock cycles.	0x00
[23:22]	-	-	Reserved	0x0
[21:16]	VW	R/W	Vertical Synchronization Pulse Width Actual vertical synchronization pulse width = VW + 1 VW is the number of horizontal synchronization lines. Program to the number of lines required minus one. The 6-bit VW field is used to specify the pulse width of the vertical synchronization pulse. The register is programmed with the number of line clocks in vertical synchronization period minus one. Number of horizontal synchronization lines.	0x00
[15:12]	-	-	Reserved	0x0
[11:0]	LF	R/W	Lines-per-Frame Actual line-per-frame = LF + 1 LF is the number of the active lines per frame. The LF field specifies the total number of lines or rows on the LCD panel being controlled. LF is an 11-bit value allowing between 1 and 2048 lines.	0x000

Vangotech

6.1.5.5. LCD Vertical Back Porch Parameter

Table 6-19. LCD Vertical Back Porch Parameter (lcd_vertiming1,Offset 0x0108)

Bit	Name	Type	Description	Default Value
[7:0]	VBP	R/W	Vertical Back Porch Actual vertical back porch = VBP, when LCDC_SHARP is turned off. Actual vertical back porch = VBP + 1, when LCDC_SHARP is turned on. VBP is the number of the inactive lines at the start of a frame after vertical synchronization period. The 8-bit VBP field is used to specify the number of line clocks inserted at the beginning of each frame. The VBP count starts just after the vertical synchronization signal for the previous frame has been negated for the active mode. After this has occurred, the count value in VBP sets the number of the line clock periods inserted before the next frame. VBP generates from 1 to 256 extra line clock cycles.	0x00

6.1.5.6. LCD Polarity Control Parameter

Table 6-20. LCD Polarity Control Parameter(lcd_polarity, Offset 0x010C)

Bit	Name	Type	Description	Default Value
[22:16]	TV_DivNo	R/W	TV panel clock divisor control (TV_PCLK) TV_PCLK is divided from TV_CLK. The actual divisor value is equal to (TV_DivNo + 1).	0x00
[14:8]	DivNo	R/W	LCD panel clock divisor control (LC_PCLK) LC_PCLK is divided from LC_CLK. The actual divisor value is equal to (DivNo + 1).	0x00
[7:5]	-	-	Reserved	
4	IPWR	R/W	0: LC_PWROFF output is active high. 1: LC_PWROFF output is active low. This bit is used to select the active polarity of the output pin LC_PWROFF.	0x0
3	IDE	R/W	0: LC_DE/LC_AC output pin is active high. 1: LC_DE/LC_AC output pin is active low. The invert output enable bit is used to select the active polarity of the output enable signal.	0x0
2	ICK	R/W	0: Data is driven on the LCD data lines on the rising edge of LC_PCLK. 1: Data is driven on the LCD data lines on the falling edge of LC_PCLK.	0x0

Vangotech

			The ICK bit is used to select the edge of the panel clock on which the pixel data is driven out onto the LCD data lines.	
1	IHS	R/W	0: LC_HS/LC_LP pin is active high and inactive low. 1: LC_HS/LC_LP pin is active low and inactive high. The invert horizontal sync. bit is used to select the active polarity of the horizontal sync. signal.	0x0
0	IVS	R/W	0: LC_VS/LC_FLM pin is active high and inactive low. 1: LC_VS/LC_FLM pin is active low and inactive high. The invert vertical sync. bit is used to select the active polarity of the vertical sync. signal.	0x0

6.1.5.7. PiP Blending Parameter

When PiP is enabled, the following restriction applies:

PiPBlend_h < 257

PiPBlend_l < 257

PiPBlend_d < 257

The outputted pixel value can be divided into the following three cases:

Case 0: When the pixel covers four images:

Output pixel value = (High_Image x PiPBlend_h x 256 x 256 + Mid_Image x (256 – PiPBlend_h) x PiPBlend_l x 256 + Low_Image x (256-PiPBlend_h) x (256-PiPBlend_l) x PiPBlend_d + Down_Image x (256-PiPBlend_h) x (256-PiPBlend_l) x (256-PiPBlend_d)/(256 x 256 x 256)

Case 1: When the pixel covers three images:

Output pixel value = (High_Image x PiPBlend_h x 256 + Mid_Image x (256 – PiPBlend_h) x PiPBlend_l + Low_Image x (256-PiPBlend_h) x (256-PiPBlend_l))/(256 x 256)

Case 2: When the pixel covers two images:

Output pixel value = (High_Image x PiPBlend_h + Low_Image x (256 – PiPBlend_h))/256

Note: The priority of image is High_Image > Mid_Image > Low_Image > Down_Imgae.

Table 6-21. PiP Blending Parameter(lcd_pipblend, Offset 0x0300)

Bit	Name	Type	Description	Default Value
[23:16]	PiPBlend_d	R/W	Alpha blending level of an image with the down priority The valid value is 0 ~ 254, and 255 represents 256.	0x0
[15:8]	PiPBlend_h	R/W	Alpha blending level of an image with a higher priority The valid value is 0 ~ 254, and 255 represents	0x0

			256.	
[7:0]	PiPBlend_l	R/W	Alpha blending level of an Image with the lower priority The valid value is 0 ~ 254, and 255 represents 256.	0x0

6.1.5.8. PiP Sub-Picture1 Position Parameter

When PiP is enabled, the following restrictions apply:

1. PiP1Hpos + PiP1Hdim ≤ The horizontal resolution of a panel
2. PiP1Vpos + PiP1Vdim ≤ The vertical resolution of a panel
3. PiP2Hpos + PiP2Hdim ≤ The horizontal resolution of a panel
4. PiP2Vpos + PiP2Vdim ≤ The vertical resolution of a panel
5. PiP3Hpos + PiP3Hdim ≤ The horizontal resolution of a panel
6. PiP3Vpos + PiP3Vdim ≤ The vertical resolution of a panel
7. If TV interlace output is set, the format of input image must be progressive.

Table 6-22. PiP Sub-Picture1 Position Parameter(lcd_pip1pos, Offset 0x0304)

Bit	Name	Type	Description	Default Value
28	PiP_Update	W	When this bit is set, FTLCDC210 will start to update all PiP dimensions and position registers. This bit will be automatically cleared when registers are updated.	0x0
[26:16]	PiP1Hpos	R/W	Specify the horizontal position of the sub-picture1 of a PiP window The reference original position is the left boundary of the LCD screen. The valid value is ranging from 1 to 2047. Note: When the register bit, “En1YCbCr422”, is enabled, the value must be an odd number.	0x0
[15:11]	-	-	Reserved	-
[10:0]	PiP1Vpos	R/W	Specify the vertical position of the sub-picture1 of a PiP window The reference original position is the up boundary of the LCD screen. The valid value is ranging from 1 to 2047. When output format set as interlace, this field must set as even number.	0x0

6.1.5.9. PiP Sub-Picture1 Dimension Parameter

Table 6-23. PiP Sub-Picture1 Dimension Parameter (lcd_pip1dim, Offset 0x0308)

Bit	Name	Type	Description	Default

				Value
[31:27]	-	-	Reserved	0x0
[26:16]	PiP1Hdim	R/W	Specify the horizontal dimension of the sub-picture1 of a PiP window Note: ● When the register bit, "En1YCbCr422", is enabled, the value must be an even number. ● The horizontal dimension *bpp should be the multiple of 32-bit or 64-bit when the AXI/AHB bus is 32 bit or 64 bit, respectively.	0x0
[15:11]	-	-	Reserved	-
[10:0]	PiP1Vdim	R/W	Specify the vertical dimension of the sub-picture1 of a PiP window	0x0

6.1.5.10. PiP Sub-Picture2 Position Parameter

Table 6-24. PiP Sub-Picture2 Position Parameter(lcd_pip2pos, Offset 0x030C)

Bit	Name	Type	Description	Default Value
[26:16]	PiP2Hpos	R/W	Specify the horizontal position of the sub-picture2 of a PiP window The reference original position is the left boundary of the LCD screen. The valid value is ranging from 1 to 2047. Note: When the register bit, "En2YCbCr422", is enabled, the value must be an odd number.	0x0
[15:11]	-	-	Reserved	0x0
[10:0]	PiP2Vpos	R/W	Specify the vertical position of the sub-picture2 of a PiP window The reference original position is the up boundary of the LCD screen. The valid value is ranging from 1 to 2047. When output format set as interlace, this field must set as even number.	0x0

6.1.5.11. PiP Sub-Picture2 Dimension Parameter

Table 6-25. PiP Sub-Picture2 Dimension Parameter(lcd_pip2dim, Offset 0x0310)

Bit	Name	Type	Description	Default Value
[26:16]	PiP2Hdim	R/W	Specify the horizontal dimension of the sub-picture2 of a PiP window Notes: ● When the register bit, "En2YCbCr422", is enabled, the value must be an even	0x0

			number. ● The horizontal dimension *bpp should be the multiple of 32 bit or 64 bit when the AXI/AHB bus is 32 bit or 64 bit, respectively	
[15:11]	-	-	Reserved	0x0
[10:0]	PiP2Vdim	R/W	Specify the vertical dimension of the sub-picture2 of a PiP window	0x0

6.1.5.12. PiP Image Priority Parameter

The image priority register controls the priority of each image when images are overlapped. The priority setting must be different for each image.

Table 6-26. PiP Image Priority Parameter(lcd_pipriority, Offset 0x0314)

Bit	Name	Type	Description	Default Value
[7:6]	Img3_priority	R/W	Image3 priority 00: Low priority 01: Middle priority 10: High priority 11: Down priority	0x3
[5:4]	Img2_priority	R/W	Image2 priority 00: Low priority 01: Middle priority 10: High priority 11: Down priority	0x2
[3:2]	Img1_priority	R/W	Image1 priority 00: Low priority 01: Middle priority 10: High priority 11: Down priority	0x1
[1:0]	Img0_priority	R/W	Image0 priority 00: Low priority 01: Middle priority 10: High priority 11: Down priority	0x0

6.1.5.13. PiP/PoP Image Format 1 Parameter

The image format register controls the format of each image when two, three, or four images are appeared at the same frame.

It is suggested to refer to the following flows when changing the PiP/PoP image format:

1. Disable the PIP
2. Wait for VSYNC (By waiting for the IntVstatus)

Vangotech

3. Modify the register
4. Enable the PIP

Table 6-27. PiP/PoP Image Format 1 Parameter(lcd_param1,Offset 0x0318)

Bit	Name	Type	Description	Default Value
[31:30]	RGBTYPE3	R/W	Image3 RGB format This field shows the image3 input format when BppFifo3 – 100. 00: RGB 565 01: RGB 555 10: RGB 444 11: Reserved	0x0
[29:28]	RGBTYPE2	R/W	Image2 RGB Format This field shows the image2 input format when BppFifo2 – 100. 00: RGB 565 01: RGB 555 10: RGB 444 11: Reserved	0x0
[27:26]	RGBTYPE1	R/W	Image1 RGB format This field shows the image1 input format when BppFifo1 – 100. 00: RGB 565 01: RGB 555 10: RGB 444 11: Reserved	0x0
[25:24]	RGBTYPE0	R/W	Image0 RGB format This field shows the image0 input format when BppFifo0 – 100. 00: RGB 565 01: RGB 555 10: RGB 444 11: Reserved	0x0
[23:22]	Endian3	R/W	Image3 endian control 00: Little-endian byte and little-endian pixel 01: Big-endian byte and big-endian pixel 10: Little-endian byte and big-endian pixel (WinCE) 11: Reserved	0x0
[21:20]	Endian2	R/W	Image2 endian control 00: Little-endian byte and little-endian pixel 01: Big-endian byte and big-endian pixel	0x0

Vangotech

			10: Little-endian byte and big-endian pixel (WinCE) 11: Reserved	
[19:18]	Endian1	R/W	Image1 endian control 00: Little-endian byte and little-endian pixel 01: Big-endian byte and big-endian pixel 10: Little-endian byte and big-endian pixel (WinCE) 11: Reserved	0x0
[17:16]	Endian0	R/W	Image0 endian control 00: Little-endian byte and little-endian pixel 01: Big-endian byte and big-endian pixel 10: Little-endian byte and big-endian pixel (WinCE) 11: Reserved	0x0
15	En3YcbCr422	R/W	Image3 YcbCr422 mode 0: Image0 is the RGB format. 1: Image0 is the YcbCr422 format.	0x0
[14:12]	BppFifo3	R/W	Image3 pixel format The value indicates the data format of image3 (Bit per pixel). 000: 1 bpp 001: 2 bpp 010: 4 bpp 011: 8 bpp 100: 16 bpp 101: 24 bpp 110: ARGB8888 111: ARGB1555	0x0
11	En2YcbCr422	R/W	Image2 YcbCr422 mode 0: Image0 is the RGB format. 1: Image0 is the YcbCr422 format.	0x0
[10:8]	BppFifo2	R/W	Image2 pixel format The value indicates the data format of image2 (Bit per pixel). 000: 1 bpp 001: 2 bpp 010: 4 bpp 011: 8 bpp 100: 16 bpp 101: 24 bpp 110: ARGB8888	0x0

Vangotech

			111: ARGB1555	
7	En1YcbCr422	R/W	Image1 YcbCr422 mode 0: Image0 is the RGB format. 1: Image0 is the YcbCr422 format.	0x0
[6:4]	BppFifo1	R/W	Image1 pixel format The value indicates the data format of image1 (Bit per pixel). 000: 1 bpp 001: 2 bpp 010: 4 bpp 011: 8 bpp 100: 16 bpp 101: 24 bpp 110: ARGB8888 111: ARGB1555	0x0
3	En0YcbCr422	R/W	Image0 YcbCr422 mode 0: Image0 is the RGB format. 1: Image0 is the YcbCr422 format.	0x0
[2:0]	BppFifo0	R/W	Image0 Pixel format The value indicates the data format of image0 (Bit per pixel). 000: 1 bpp 001: 2 bpp 010: 4 bpp 011: 8 bpp 100: 16 bpp 101: 24 bpp 110: ARGB8888 111: ARGB1555	0x0

6.1.5.14. PiP/PoP Image Format 2 ParameParameter

The Images Format register decides the format of each image when two, three, or four images appear at the same time. This register is effective only when FTLCDC210_PIOP is configured.

Table 6-28. PiP/PoP Image Format 2 Parameter(lcd_param2, Offset 0x031C)

Bit	Name	Type	Description	Default Value
5	FgImgYcbCr420	R/W	Image format YCbCr420 for PiP The value indicates that the foreground image is the YCbCr420 format. 0 = Foreground image is not the YCbCr format. 1 = Foreground image is the YCbCr format.	0x0

Vangotech

			Note: FTLCDC210 only supports one image, YCbCr420. When the YCbCr420 format and PiP are tuned-on, PiP only supports of up to two images, including the background image. The base address of the YCbCr420 image format should be set as follows: The base address of Y is at LCDImage0FrameBase (18H). The base address of Cb is at LCDImage1FrameBase (24H). The base address of Cr is at LCDImage2FrameBase (30H).	
4	BgImgYcbCr420	R/W	Image format YCbCr420 for PiP The value indicates that the background image is the YCbCr420 format. 0 = Background image is not the YCbCr format. 1 = Background image is the YCbCr format.	0x0
3	Img3Format	R/W	Image3 format The value decides that the image3 in the frame buffer is the format of "Progress" or "Interlace." This bit is effective when TVEn is enabled. 0 = Image3 format is progress. 1 = Image3 format is interlace.	0x0
2	Img2Format	R/W	Image2 format The value decides that the image2 in the frame buffer is the format of "Progress" or "Interlace." This bit is effective when TVEn is enabled. 0 = Image2 format is progress. 1 = Image2 format is interlace.	0x0
1	Img1Format	R/W	Image1 format The value decides that the image1 in the frame buffer is the format of "Progress" or "Interlace." This bit is effective when TVEn is enabled. 0 = Image1 format is progress. 1 = Image1 format is interlace.	0x0
0	Img0Format	R/W	Image0 format The value decides that the image0 in the frame buffer is the format of "Progress" or "Interlace." This bit is effective when TVEn is enabled. 0 = Image0 format is progress. 1 = Image0 format is interlace.	0x0

Vangotech

6.1.5.15. PiP Color Key1

The pixel of the PiP sub-picture1 will be transparent when its pixel value equals to the value of ColorKey.

Table 6-29. PiP Color Key1(lcd_colorkey1, Offset 0x0320)

Bit	Name	Type	Description	Default Value
24	ColorKey1En	R/W	Color key for the PiP image1 function control 0: Disable 1: Enable	0x0
[23:0]	ColorKey1	R/W	Color key for the PiP image1 value	0x0

6.1.5.16. PiP Color Key2

The pixel of the PiP sub-picture2 will be transparent when its pixel value equals to the value of ColorKey.

Table 6-30. PiP Color Key2(lcd_colorkey2, Offset 0x0324)

Bit	Name	Type	Description	Default Value
24	ColorKey2En	R/W	Color key for the PiP image2 function control 0: Disable 1: Enable	0x0
[23:0]	ColorKey2	R/W	Color key for the PiP image2 value	0x0

6.1.5.17. PiP Color Key3

The pixel of the PiP sub-picture3 will be transparent when its pixel value equals to the value of ColorKey.

Table 6-31. PiP Color Key3 (lcd_colorkey3, Offset 0x0328)

Bit	Name	Type	Description	Default Value
24	ColorKey3En	R/W	Color key for the PiP image3 function control 0: Disable 1: Enable	0x0
[23:0]	ColorKey3	R/W	Color key for the PiP image3 value	0x0

6.1.5.18. PiP Sub-Picture3 Position

Table 6-32. PiP Sub-Picture3 Position(lcd_pip3pos, Offset 0x032C)

Bit	Name	Type	Description	Default value
[26:16]	PiP3Hpos	R/W	Specify the horizontal position of the sub-picture3 of a PiP window The reference original position is the left boundary of the LCD screen. The valid value is ranging from 1 to 2047. Note: When the register bit, "En3YCbCr422", is enabled, the value must be an odd number.	0x0

Vangotech

[15:11]	-	-	Reserved	-
[10:0]	PiP3Vpos	R/W	Specify the vertical position of the sub-picture3 of a PiP window The reference original position is the up boundary of the LCD screen. The valid value is ranging from 1 to 2047. When output format set as interlace, this field must set as even number.	0x0

6.1.5.19. PiP Sub-Picture3 Dimension Parameter

Table 6-33. PiP Sub-Picture3 Dimension Parameter (lcd_pip3dim, Offset 0x0330)

Bit	Name	Type	Description	Default Value
[26:16]	PiP3Hdim	R/W	Specify the horizontal dimension of the sub-picture3 of a PiP window Notes: ● When the register bit, "En3YCbCr422", is enabled, the value must be an even number. ● The horizontal dimension *bpp should be the multiple of 32 bit or 64 bit when the AXI/AHB bus is 32 bit or 64 bit, respectively.	0x0
[15:11]	-	-	Reserved	-
[10:0]	PiP3Vdim	R/W	Specify the vertical dimension of the sub-picture3 of a PiP window	0x0

6.1.5.20. LCD Palette RAM Write Accessing Port

Example of address 0x0A00 (For index values 0 and 1)

Table 6-34. Palette RAM(PaletteWritePort, Offset 0x0A00 ~ 0x0BFC)

Bit	Name	Type	Description
[31:16]	PaletteEty1	W	Not available for the YcbCr mode due to the fact that the data in the frame buffer represent raw YcbCr components and cannot be remapped. This 16-bits value is used to remap color when the index value equals to 1. The format is described below: D_{31-27}, form the MSB of red component and the remaining LSBs will be filled with '0'; D_{26-21}, form the MSB of green component and the remaining LSBs will be filled with '0'; D_{20-16}, form the MSB of blue component and the remaining LSBs will be filled with '0'.
[15:0]	PaletteEty0	W	This 16-bit value is used to remap color when the index value equals to 0. The operation is similar to previous one. D_{15-11}, form the MSB of red component and the remaining LSBs

			will be filled with '0'; D_{10-5}, form the MSB of green component and the remaining LSBs will be filled with '0'; D_{4-0}, form the MSB of blue component and the remaining LSBs will be filled with '0'.

6.1.5.21. LCD Color Management Parament0 Parameter

Table 6-35. LCD Color Management Parament0 Parameter(lcd_colorman0, Offset 0x0400)

Bit	Name	Type	Description	Default Value
[13:8]	SatValue	R/W	Saturation Value Cb (sat) = Cb (org) * (SatValue/32) Cr (sat) = Cr (org) * (SatValue/32)	0x20
7	BrightSign	R/W	Sign bit of brightness value 0: The value of brightness is positive. 1: The value of brightness is negative.	0x0
[6:0]	Brightness	R/W	Brightness level The range of the brightness level is from 0 to 127.	0x0

6.1.5.22. LCD Color Management Parament1 Parameter

This register defines the coefficient of the hue operation.

Cb (hue) = Cb (org) * cosθ – Cr (org) * sinθ

Cr (hue) = Cb (org) * sinθ + Cr (org) * cosθ

Table 6-36. LCD Color Management Parament1 Parameter(lcd_colorman1, Offset 0x0404)

Bit	Name	Type	Description	Default Value
14	SigHuCos	R/W	Sign bit of HuCosValue 0: The value of HuCosValue is positive. 1: The value of HuCosValue is negative.	0x0
[13:8]	HuCosValue	R/W	Hue value of coefficient Cosθ HuCosValue is the value of Cosθ. The coefficient of cosθ is (HuCosValue/32) θ is the rotating degree from -180° to 180°	0x20
7	-	-	Reserved	0x0
6	SigHuSin	R/W	Sign bit of HuSinValue 0: The value of HuSinValue is positive 1: The value of HuSinValue is negative	0x0
[5:0]	HuSinValue	R/W	Hue Value of coefficient Sinθ HuSinValue is the value of Sinθ. The coefficient of sin*θ* is (HuSinValue/32) θ is the rotating degree from -180° to 180°	0x0

Vangotech

6.1.5.23. LCD Color Management Parament2 Parameter

This register defines the coefficient of the sharpness operation.

Table 6-37. LCD Color Management Parament2 Parameter(lcd_colorman2, Offset 0x0408)

Bit	Name	Type	Description	Default Value
[23:20]	K1	R/W	Sharpness weight value1 The value determines the second weight of sharpness. The actual value is (K1/2).	0x0
[19:16]	K0	R/W	Sharpness weight value0 The value determines the first weight of sharpness. The actual value is (K0/2).	0x0
[15:8]	ShTh1	R/W	Sharpness threshold value1 The value determines the second threshold value of sharpness.	0x0
[7:0]	ShTh0	R/W	Sharpness threshold value0 The value determines the first threshold value of sharpness.	0x0

6.1.5.24. LCD Color Management Parament3 Parameter

This register defines the coefficient of the contrast operation.

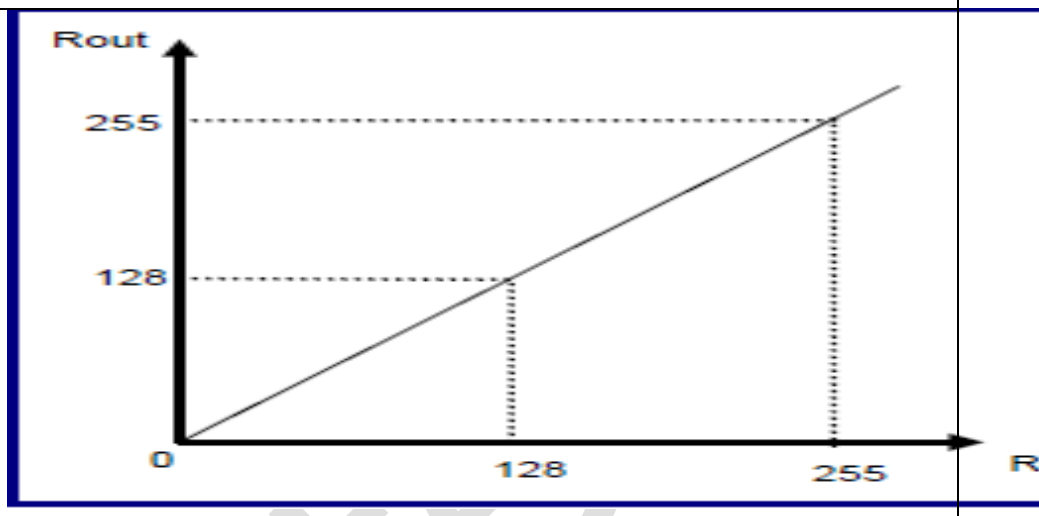
Table 6-38. LCD Color Management Parament3 Parameter(lcd_colorman3, Offset 0x040C)

Bit	Name	Type	Description	Default Value
[31:21]	-	-	Reserved	0x0
[20:16]	Contr_slope	R/W	Contrast cure slope The value determines the slope of contrast cure. The actual slope is the value divided by 4. Note: This value cannot be programmed to 0.	
[15:13]	-	-	Reserved	-
12	Contr_offset_sign	R/W	Contrast offset sign 1: (Contr_slope x 128) > 512 0: (Contr_slope x 128) < 512	0x0
[11:0]	Contr_offset	R/W	Contrast offset value The value is defined as absolute of "Contr_slope x 128 – 512"	0x0

6.1.5.25. Lookup Table of LCD Gamma Red

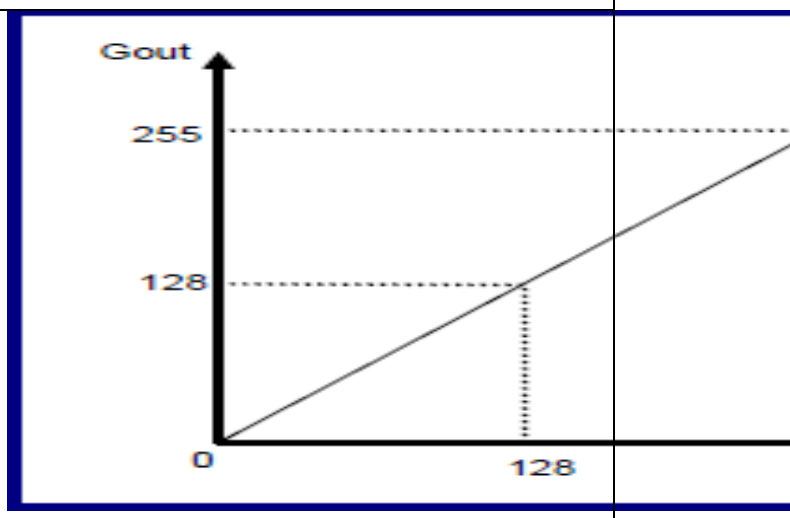
Table 6-39. Lookup Table of LCD Gamma Red(lcd_GaRdLUT, Offset 0x0600 ~ 0x06FC)

Vantotech

Bit	Name	Type	Offset	Description
[31:0]	GaRdLUT0	R/W	0x00	The corresponding output values of the Red component (0, 1, 2, 3)
[31:0]	GaRdLUT1	R/W	0x04	The corresponding output values of the Red component (4, 5, 6, 7)
[31:0]	GaRdLUT2	R/W	0x08	
[31:0]	GaRdLUT3	R/W	0x0C	
[31:0]	GaRdLUT4	R/W	0x10	
[31:0]	GaRdLUT5	R/W	0x14	
…	…	…	…	
[31:0]	GaRdLUT63	R/W	0xFC	The corresponding output values of the Red component (252, 253, 254, 255)

6.1.5.26. Lookup Table of LCD Gamma Green

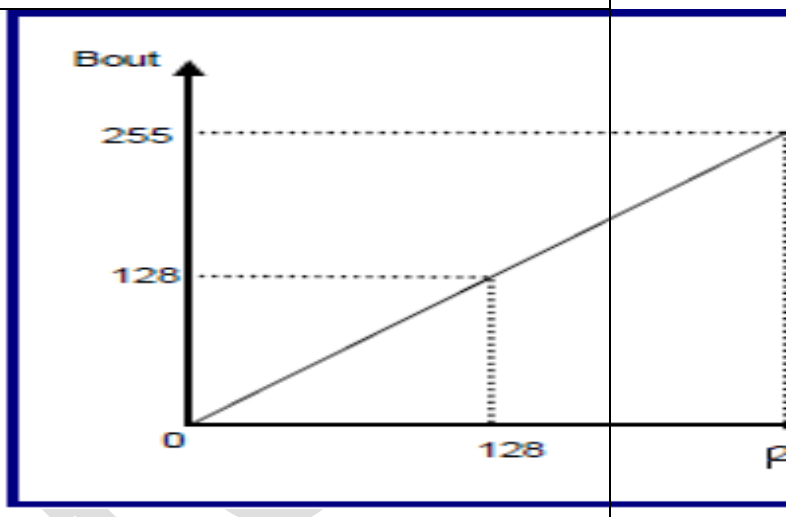
Table 6-40. Lookup Table of LCD Gamma Green(lcd_GaGnLUT, Offset 0x0700 ~ 0x07FC)

Bit	Name	Type	Offset	Description
[31:0]	GaGnLUT0	R/W	0x00	The corresponding output values of the Green component (0, 1, 2, 3)
[31:0]	GaGnLUT1	R/W	0x04	The corresponding output values of the Green component (4, 5, 6, 7)
[31:0]	GaGnLUT2	R/W	0x08	
[31:0]	GaGnLUT3	R/W	0x0C	
[31:0]	GaGnLUT4	R/W	0x10	
[31:0]	GaGnLUT5	R/W	0x14	
…	…	…	…	
[31:0]	GaGnLUT63	R/W	0xFC	The corresponding output values of the Green component (252, 253, 254, 255)

Vangote

6.1.5.27. Lookup Table of LCD Gamma Blue

Table 6-41. Lookup Table of LCD Gamma Blue (lcd_GaBuLUT, Offset 0x0800 ~ 0x08FC)

Bit	Name	Type	Offset	Description
[31:0]	GaBuLUT0	R/W	0x00	The corresponding output values of the Blue component (0, 1, 2, 3)
[31:0]	GaBuLUT1	R/W	0x04	The corresponding output values of the Blue component (4, 5, 6, 7)
[31:0]	GaBuLUT2	R/W	0x08	
[31:0]	GaBuLUT3	R/W	0x0C	
[31:0]	GaBuLUT4	R/W	0x10	
[31:0]	GaBuLUT5	R/W	0x14	
…	…	…	…	
[31:0]	GaBuLUT63	R/W	0xFC	The corresponding output values of the Blue component (252, 253, 254, 255)

6.1.5.28. Scaler Control Registers

The scaler cannot work when TV is enabled and the input and output format is not progressive. If users turn on TV and the input or output format is interlace, users should use the 0x14 register to perform simple scaling.

6.1.5.29. Horizontal Resolution Register of Scaler Input

Table 6-42. Horizontal Resolution Register of Scaler Input(scal_hor_no_in, Offset: 0x1100)

Bit	Name	Type	Description
[31:12]	-	-	Reserved
[11:0]	hor_no_in	R/W	Horizontal Resolution of Input Image Actual horizontal resolution of input image = hor_no_in + 1 This field identifies the input image horizontal resolution and must be programmed a nonzero value before scaler is active. The value is between 1 and 2047. Users must ensure that each line start is four-word alignment.

6.1.5.30. Vertical Resolution Register of Scaler Input

Table 6-43. Vertical Resolution Register of Scaler Input(scal_ver_no_in, Offset 0x1104)

Bit	Name	Type	Description	Default Value
[31:12]	-	-	Reserved	-

[11:0]	hor_no_in	R/W	Horizontal Resolution of Input Image Actual horizontal resolution of input image = hor_no_in + 1 This field identifies the input image horizontal resolution and must be programmed a nonzero value before scaler is active. The value is between 1 and 2047. Users must ensure that each line start is four-word alignment.	0x0

6.1.5.31. Horizontal Resolution Register of Scaler Output

Table 6-44. Horizontal Resolution Register of Scaler Output(scal_hor_no_out, Offset 0x 1108)

Bit	Name	Type	Description	Default Value
[31:14]	-	-	Reserved	-
[13:0]	Hor_no_out	R/W	Horizontal Resolution of Output Image Actual horizontal resolution of output image = Hor_no_out This field identifies the output image horizontal resolution and must be programmed as a nonzero value before the scaler is active. The value is between 1 and 4096. Note: The scaler output resolution should be equal to the panel active region resolution.	0x0

6.1.5.32. Vertical Resolution Register of Scaler Output

Table 6-45. Vertical Resolution Register of Scaler Output(scal_ver_no_out, Offset 0x110C)

Bit	Name	Type	Description	Default Value
[31:14]	-	-	Reserved	-
[13:0]	Ver_no_out	R/W	Vertical Resolution of Output Image Actual vertical resolution of output image = Ver_no_out This field identifies the output image vertical resolution and must be programmed as a nonzero value before scaler is active. The value is between 1 and 4096. Note: The scaler output resolution should be equal to the panel active region resolution.	0x0

6.1.5.33. Scaler Control Register

Table 6-46. Scaler Control Register (scal_misc, Offset 0x1110)

Bit	Name	Type	Description	Default Value
[31:14]	-	-	Reserved	
[8:6]	fir_sel	R/W	The scaling ratio selection of the 1st stage	0x0

Vangotech

			scaler 000: Bypass the 1st stage scaler 001: Scaling down to 1/2x1/2 010: Scaling down to 1/4x1/4 011: Scaling down to 1/8x1/8 100: Scaling down to 1/16x1/16 101: Scaling down to 1/32x1/32 110: Scaling down to 1/64x1/64 111: Scaling down to 1/128x1/128	
5	-	-	Reserved	
[4:3]	hor_inter_mode	R/W	The horizontal interpolation mode 00: Nearly bilinear mode Modes below are only available on the up-scaling. 01: Threshold nearly bilinear mode 10: Most neighborhood mode 11: Reserved	0x0
[2:1]	ver_inter_mode	R/W	The vertical interpolation mode 00: Nearly bilinear mode Modes below are only available on the up-scaling. 01: Threshold nearly bilinear mode 10: Most neighborhood mode 11: Reserved	0x0
0	bypass_mode	R/W	This bit identifies the 2nd stage scaler bypass mode and default the scaler is bypassed, this bit must be set in the initial state. Once it is set, it should not be changed. Users must clear this bit to make the scaler active.	0x0

6.1.5.34. Horizontal High Threshold Register

Table 6-47. Horizontal High Threshold Register(scal_hor_high_th, Offset 0x1114)

Bit	Name	Type	Description	Default Value
[31:9]	-	-	Reserved	
[8:0]	hor_high_th	R/W	Horizontal up-scaling high threshold register. The up-scaling coefficient is highly limited by this threshold value in the horizontal direction. The real value equals to Hor_high_th/256. This value is ranging from 0 ~ 256.	0x0

Vangotech

6.1.5.35. Horizontal Low Threshold Register

Table 6-48. Horizontal Low Threshold Register (scal_hor_low_th, Offset 0x1118)

Bit	Name	Type	Description	Default Value
[31:9]	-	-	Reserved	
[8:0]	hor_low_th	R/W	Horizontal up-scaling low threshold register. The up-scaling coefficient is low limited by this threshold value in the horizontal direction. The real value equals to Hor_low_th/256. This value is ranging from 0 ~ 256.	0x0

6.1.5.36. Vertical High Threshold Register

Table 6-49. Vertical High Threshold Register (scal_ver_high_th, Offset 0x111C)

Bit	Name	Type	Description	Default Value
[31:9]	-	-	Reserved	
[8:0]	ver_high_th	R/W	Vertical up-scaling high threshold register. The up-scaling coefficient is high limited by this threshold value in the vertical direction. The real value equals to ver_high_th/256. This value is ranging from 0 ~ 256.	0x0

6.1.5.37. Vertical Low Threshold Register

Table 6-50. Vertical Low Threshold Register(scal_ver_low_th, Offset 0x1120)

Bit	Name	Type	Description	Default Value
[31:9]	-	-	Reserved	-
[8:0]	ver_low_th	R/W	Vertical up-scaling low threshold register. The up-scaling coefficient is low limited by this threshold value in the vertical direction. The real value equals to ver_low_th/256. This value is ranging from 0~256.	0x0

6.1.5.38. Scaler Resolution Parameter

The parameter scal_hor_num and scal_ver_num are calculated as shown below, where ver_no_in/hor_no_in is either the input image resolution register value if the first scaler is bypassed or the output image resolution of the first scaler minus one if the first scaler is involved.

Scaling up:

Scal_ver_num = [(ver_no_in+1)/ver_no_out] x 256

Scal_hor_num = [(hor_no_in+1)/hor_no_out] x 256

Scaling down :

Scal_ver_num = [mod ((ver_no_in+1)/ver_no_out)] x 256/ver_no_out

Scal_hor_num = [mod ((hor_no_in+1)/hor_no_out)] x256/hor_no_out

Table 6-51. Scaler Resolution Parameter (scal_hor_ver_num, Offset 0x112C)

Bit	Name	Type	Description	Default Value
[31:16]	-	-	Reserved	-
[15:8]	scal_hor_num	R/W	The second-stage initial numerator of the scaler coefficient in the horizontal direction.	0x0
[7:0]	scal_ver_num	R/W	The second-stage initial numerator of the scaler coefficient in the vertical direction.	0x0

6.1.5.39. Cursor Position Control Register

Table 6-52. Cursor Position Control Register (cursor_position, Offset 0x1200)

Bit	Name	Type	Description	Default Value
[31:30]	-	-	Reserved	-
29	CurNew	R/W	New cursor 1: The cursor font database is 32 * 32 pixel. 0: The cursor font database is 12 * 16 pixel.	0x0
28	CurUpdate	R/W	Updated the cursor The values of "CurHPos" and "CurVPos" are actually updated to registers, when this bit is set to '1'. This bit can be automatically cleared.	0x0
[27:16]	CurHPos	R/W	Specify the horizontal position of the Cursor window The reference original position is the left boundary of the LCD screen. The valid value is ranging from 0 to 2048.	0x0
[15:12]	-	-	Reserved	-
[11:0]	CurVPos	R/W	Specify the vertical position of the Cursor window The reference original position is the upper boundary of the LCD screen. The valid value is ranging from 0 to 2048.	0x0

6.1.5.40. Cursor Color Control Register

Table 6-53. Cursor Color Control Register (cur_color1 ~ cur_color15, Offset 0x1204 ~0x123C)

Address	Bit	Name	Type	Description	Default

Offset					Value
0x1204	[23:0]	CurColor1	R/W	Cursor color palette entry 1 The format is YcbCr.	0x0
0x1208	[23:0]	CurColor2	R/W	Cursor color palette entry 2 The format is YcbCr.	0x0
0x120C	[23:0]	CurColor3	R/W	Cursor color palette entry 3 The format is YcbCr.	0x0
0x1210	[23:0]	CurColor4	R/W	Cursor color palette entry 4 The format is YcbCr.	0x0
0x1214	[23:0]	CurColor5	R/W	Cursor color palette entry 5 The format is YcbCr.	0x0
0x1218	[23:0]	CurColor6	R/W	Cursor color palette entry 6 The format is YcbCr.	0x0
0x121C	[23:0]	CurColor7	R/W	Cursor color palette entry 7 The format is YcbCr.	0x0
0x1220	[23:0]	CurColor8	R/W	Cursor color palette entry 8 The format is YcbCr.	0x0
0x1224	[23:0]	CurColor9	R/W	Cursor color palette entry 9 The format is YcbCr.	0x0
0x1228	[23:0]	CurColor10	R/W	Cursor color palette entry 10 The format is YcbCr.	0x0
0x122C	[23:0]	CurColor11	R/W	Cursor color palette entry 11 The format is YcbCr.	0x0
0x1230	[23:0]	CurColor12	R/W	Cursor color palette entry 12 The format is YcbCr.	0x0
0x1234	[23:0]	CurColor13	R/W	Cursor color palette entry 13 The format is YcbCr.	0x0
0x1238	[23:0]	CurColor14	R/W	Cursor color palette entry 14 The format is YcbCr.	0x0
0x123C	[23:0]	CurColor15	R/W	Cursor color palette entry 15 The format is YcbCr.	0x0

6.1.5.41. 32 * 32 Cursor Font Database Write Accessing Port

The database of each cursor font is stored as a 32 x 32 matrix in the font-RAM. This RAM is 32-bit wide, which requires 64 entries (2048-bit) to record the font content. If the hardware cursor is configured as the 64 x 64 pixel array, the cursor font database (0x1300) ~ (0x13FC) will be removed.

Table 6-54. Cursor Font (cur_font, Offset 0x1300 ~ 0x13FC)

Address Offset	Type	Description															
		P_{15}	P_{14}	P_{13}	P_{12}	P_{11}	P_{10}	P_9	P_8	P_7	P_6	P_5	P_4	P_3	P_2	P_1	P_0
0x1300(0)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0x1304(1)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1308(2)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x130C(3)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1310(4)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1314(5)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1318(6)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0
0x131C(7)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0x1320(8)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0
0x1324(9)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0x1328(10)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0
0x132C(11)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1330(12)	R/W	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1334(13)	R/W	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1
0x1338(14)	R/W	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1	1
0x133C(15)	R/W	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	1
0x1380(16)	R/W	0	0	0	1	1	0	0	0	0	0	0	0	1	1	1	1
0x1384(17)	R/W	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1
0x1388(18)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0x138C(19)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1390(20)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1394(21)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1398(22)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x139C(23)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13A0(24)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13A4(25)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13A8(26)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13AC(27)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13B0(28)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13B4(29)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13B8(30)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x13BC(31)	R/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

vangotech

6.1.5.42. 64 * 64 Cursor Font Database Write Accessing Port

The database of each cursor font is stored as a 64 x 64 matrix in the font-RAM. This RAM is 32-bit wide, which requires 512 entries (16384-bit) to record the font content. If the hardware cursor is configured as the 32 x 32 pixel array, the cursor font database (0x1600) ~ (0x1DFC) will be removed.

The cursor font 0x1600 is mapping to the screen location from (0, 0) to (7, 0), and so on.

Table 6-55. Cursor Font(cur_font, Offset 0x1600 ~ 0x1DFC)

Pixel Num / Addr offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1600 -160C (0)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1620 -162C (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	0	0
1640 -164C (2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1660 -166C (3)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1680 -168C (4)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16A0 -16AC (5)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16C0 -16CC (6)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16E0 -16EC (7)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1700 -170C (8)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1720 -172C (9)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1740 -174C (10)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1760 -176C (11)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1780 -178C (12)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17A0 -17AC (13)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17C0 -17CC (14)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17E0 -17EC (15)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1800 -180C (16)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1820 -182C (17)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1840 -184C (18)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1860 -186C (19)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1880 -188C (20)	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18A0 -18AC (21)	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0
18C0 -18CC (22)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0
18E0 -18EC (23)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0

Pixel Num / Addr offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1900 -190C (24)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1920 -192C (25)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1940 -194C (26)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1960 -196C (27)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1980 -198C (28)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
19A0 -19AC (29)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
19C0 -19CC (30)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
19E0 -19EC (31)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1A00 -1A0C (32)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1A20 -1A2C (33)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1A40 -1A4C (34)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1A60 -1A6C (35)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1A80 -1A8C (36)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4
1AA0 -1AAC (37)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4
1AC0 -1ACC (38)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	0	0	4	4	4	4
1AE0 -1AEC (39)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	0	0	0	4	4	4
1B00 -1B0C (40)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	0	0	0	0	0	4	4
1B20 -1B2C (41)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	4
1B40 -1B4C (42)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	0	0	0
1B60 -1B6C (43)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B80 -1B8C (44)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BA0 -1BAC (45)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BC0 -1BCC (46)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BE0 -1BEC (47)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Pixel Num / Addr offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1C00 -1C0C (48)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C20 -1C2C (49)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C40 -1C4C (50)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C60 -1C6C (51)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C80 -1C8C (52)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CA0 -1CAC (53)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CC0 -1CCC (54)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CE0 -1CEC (55)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D00 -1D0C (56)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D20 -1D2C (57)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D40 -1D4C (58)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D60 -1D6C (59)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D80 -1D8C (60)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DA0 -1DAC (61)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DC0 -1DCC (62)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DE0 -1DEC (63)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Pixel Num / Addr offset	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
1610 -161C (0)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1630 -163C (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	0	0
1650 -165C (2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1670 -167C (3)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1690 -169C (4)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16B0 -16BC (5)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16D0 -16DC (6)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16F0 -16FC (7)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1710 -171C (8)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1730 -173C (9)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1750 -175C (10)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1770 -177C (11)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1790 -179C (12)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17B0 -17BC (13)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17D0 -17DC (14)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17F0 -17FC (15)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1810 -181C (16)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1830 -183C (17)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1850 -185C (18)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1870 -187C (19)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1890 -189C (20)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18B0 -18BC (21)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18D0 -18DC (22)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18F0 -18FC (23)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Pixel Num / Addr offset	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
1910 -191C (24)	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1930 -193C (25)	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1950 -195C (26)	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1970 -197C (27)	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1990 -199C (28)	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19B0 -19BC (29)	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19D0 -19DC (30)	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19F0 -19FC (31)	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1A10 -1A1C (32)	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1A30 -1A3C (33)	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1A50 -1A5C (34)	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1A70 -1A7C (35)	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1A90 -1A9C (36)	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1AB0 -1ABC (37)	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1AD0 -1ADC (38)	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1AF0 -1AFC (39)	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B10 -1B1C (40)	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B30 -1B3C (41)	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B50 -1B5C (42)	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B70 -1B7C (43)	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1B90 -1B9C (44)	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BB0 -1BBC (45)	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BD0 -1BDC (46)	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1BF0 -1BFC (47)	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0

Pixel Num / Addr offset	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
1C10 -1C1C (48)	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0
1C30 -1C3C (49)	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0
1C50 -1C5C (50)	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0
1C70 -1C7C (51)	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C90 -1C9C (52)	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CB0 -1CBC (53)	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CD0 -1CDC (54)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1CF0 -1CFC (55)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D10 -1D1C (56)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D30 -1D3C (57)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D50 -1D5C (58)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D70 -1D7C (59)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1D90 -1D9C (60)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DB0 -1DBC (61)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DD0 -1DDC (62)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1DF0 -1DFC (63)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.1.5.43. LCM Function Registers

FTLCDC210 supports the LCM (Liquid Crystal Display Module) panel type. The LCM controller provides the 6800 and 8080 interfaces to communicate with the display panels, such as the 8-bit, 16-bit, and 18-bit mono-STN, CSTN, and TFT display panels. This controller provides the flexible programming timing registers to fit different panels.

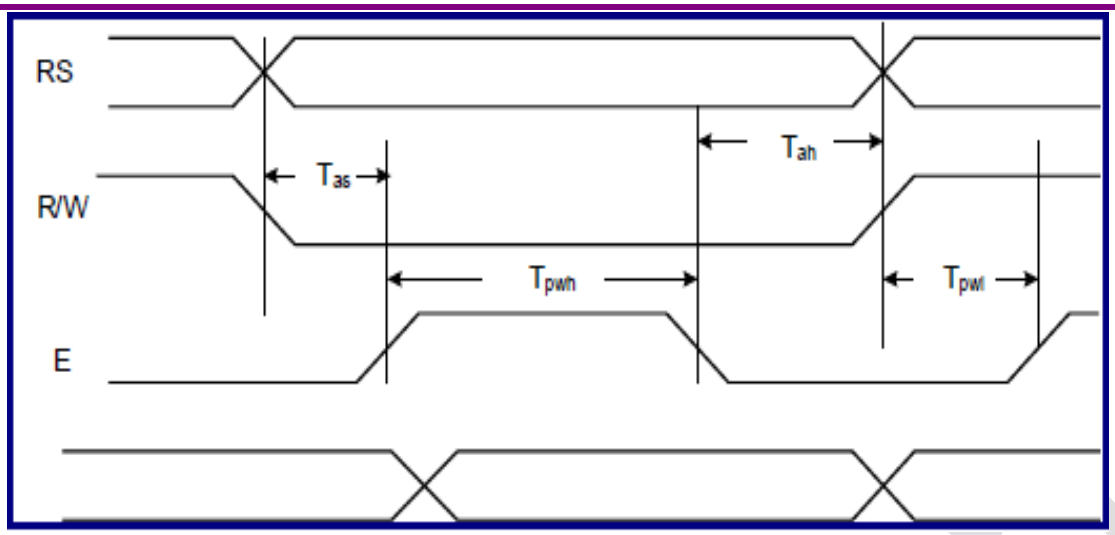


Figure 6-3. Write Operation in 6800 Mode

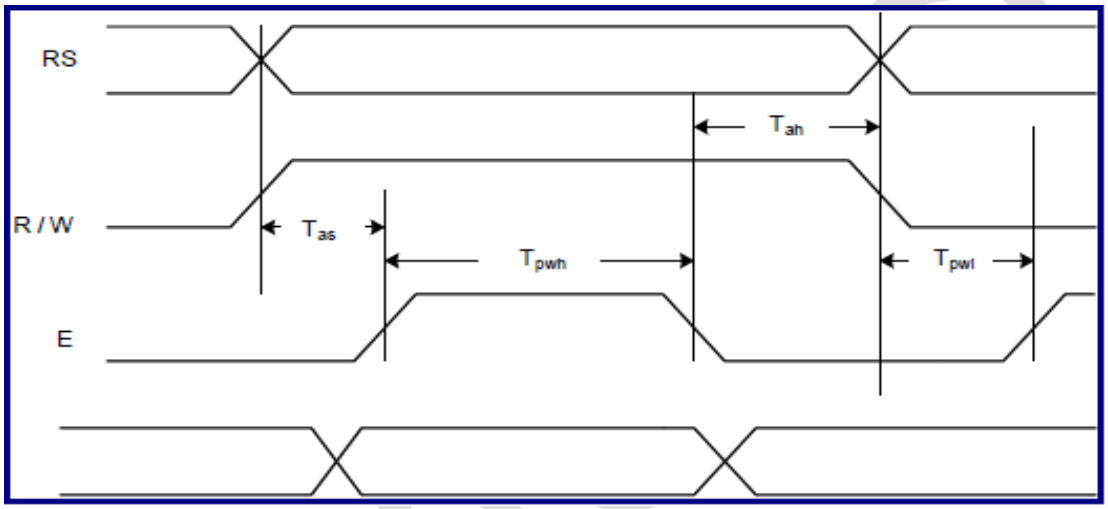


Figure 6-4. Read Operation in 6800 Mode

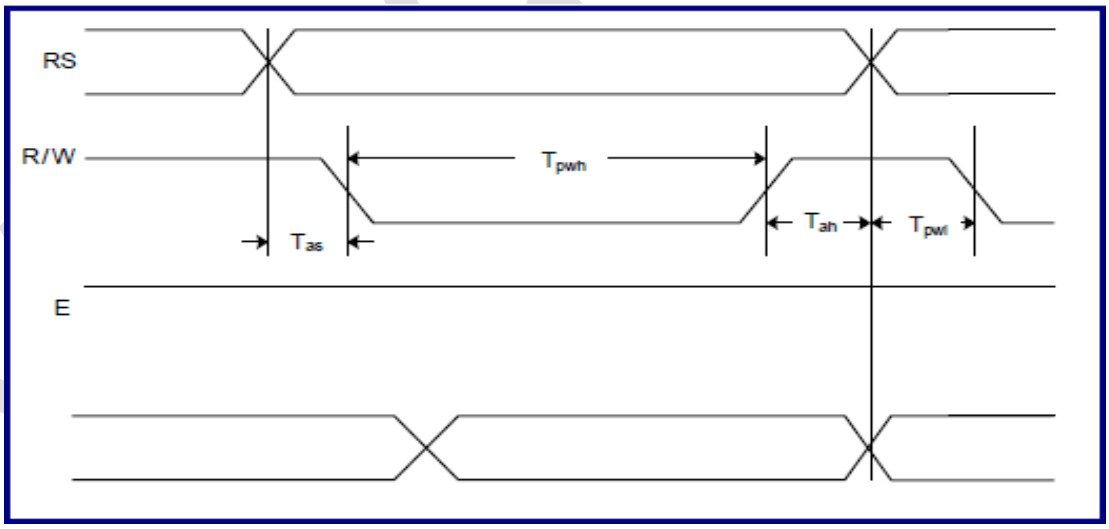


Figure 6-5. Write Operation in 8080 Mode

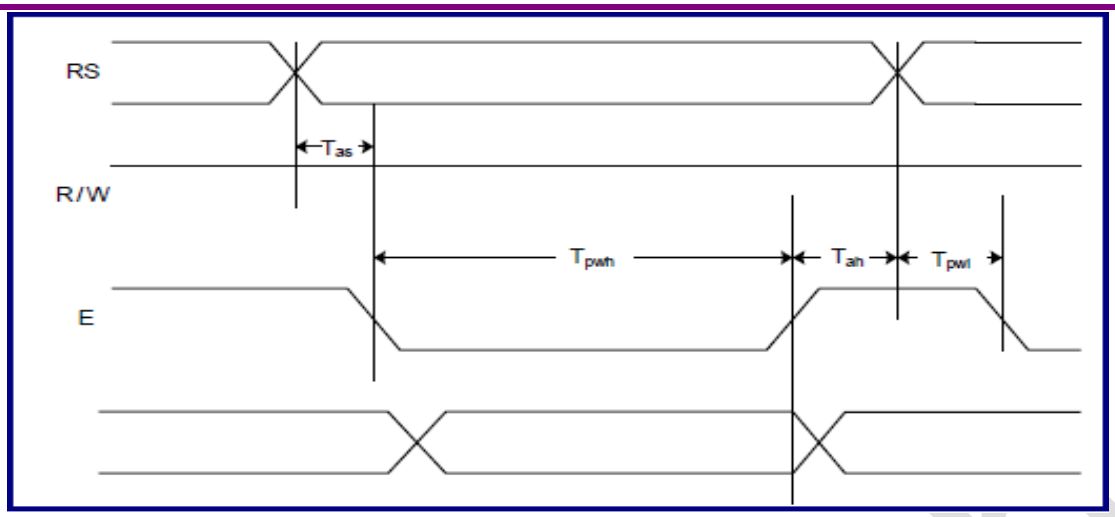


Figure 6-6. Read Operation in 8080 Mode

In the CPU mode, the register setting sequence of the read/write command/data is shown below.

Step	Action	
1	If Rdy_C&D = 1, go to step 2	
Read		**Write**
2	Write address to the LCM command register (Offset 0x1410)	Write address to the LCM command register (Offset 0x1410)
3	If Rdy_C&D = 1, go to step 4	If Rdy_C&D = 1, go to step 4
4	Write DMYRD_RS = 1	Write data to the LCD data window (Offset 0x140C)
5	If Rdy_C&D = 1, go to step 6	
6	Read data is ready in the 0x1410 register; users can obtain data from the 0x1410 register.	

6.1.5.44. Virtual Screen Control Regi ster

Table 6-56. Virtual Screen Control Register (vs_control, Offset 0x1500)

Bit	Name	Type	Description	Default Value
[31:16]	-	-	Reserved	-
20	VirtualScreenEn	R/W	Virtual screen enable 0: Disable 1: Enable	0x0
[19:17]	-	-	Reserved	-
[16:12]	PixelOffset	R/W	In 32-bit AMBA data bus, the starting address of a screen equals to "LCDImage0FrameBase" + "PixelOffset * bppfifo0". The unit of PixelOffset is pixel. Note: This field should be set as 0.	0x0
[11:0]	Image_H_Dim	R/W	The horizontal dimension of an image stored in the horizontal frame buffer (Pixels-per-line).	0x0

			Users must ensure that each line start is four-word alignment.	

6.1.5.45. Virtual Screen1 Control Register

Table 6-57. Virtual Screen1 Control Register(vs1_control, Offset 0x1504)

Bit	Name	Type	Description	Default Value
[31:16]	-	-	Reserved	-
20	VirtualScreen1En	R/W	Virtual screen enable 0: Disable 1: Enable	0x0
[19:17]	-	-	Reserved	-
[16:12]	PixelOffset1	R/W	In 32-bit AMBA data bus, the starting address of a screen equals to "LCDImage1FrameBase" + "PixelOffset1 * bppfifo1". The unit of PixelOffset1 is pixel.	0x0
[11:0]	Image1_H_Dim	R/W	The horizontal dimension of an image stored in the horizontal frame buffer (Pixels per line)	0x0

6.1.5.46. Virtual Screen2 Control Register

Table 6-58. Virtual Screen2 Control Register(vs2_control, Offset 0x1508)

Bit	Name	Type	Description	Default Value
[31:16]	-	-	Reserved	-
20	VirtualScreen2En	R/W	Virtual screen enable 0: Disable 1: Enable	0x0
[19:17]	-	-	Reserved	-
[16:12]	PixelOffset2	R/W	In 32-bit AMBA data bus, the starting address of a screen equals to "LCDImage2FrameBase" + "PixelOffset2 * bppfifo2". The unit of PixelOffset2 is pixel.	0x0
[11:0]	Image2_H_Dim	R/W	The horizontal dimension of an image stored in the horizontal frame buffer (Pixels per line)	0x0

6.1.5.47. Virtual Screen3 Control Register

Table 6-59. Virtual Screen3 Control Register (vs3_control, Offset 0x150C)

Bit	Name	Type	Description	Default Value
[31:16]	-	-	Reserved	-
20	VirtualScreen3En	R/W	Virtual screen enable	0x0

Vangotech

			0: Disable 1: Enable	
[19:17]	-	-	Reserved	-
[16:12]	PixelOffset3	R/W	In 32-bit AMBA data bus, the starting address of a screen equals to "LCDImage3FrameBase" + "PixelOffset3 * bppfifo3". The unit of PixelOffset3 is pixel.	0x0
[11:0]	Image3_H_Dim	R/W	The horizontal dimension of an image stored in the horizontal frame buffer (Pixels per line)	0x0

6.1.5.48. OSD Control Registers with Complex Configuration

In FTLCDC210, two different OSD functions can be provided by the hardware configuration. One is more complex than the other. The functions and gate counts are to be a trade-off issue. If users use FTLCDC210 with a defined FTLCDC210_OSD_SIMPLE macro, this sector can be ignored.

Note: Complex OSD is not supported if the TV is turned on.

6.1.5.49. OSD Windows On/Off Switch Register

Table 6-60. OSD Windows On/Off Switch Register (osd_windowOn, Offset 0x2000)

Bit	Name	Type	Description	Default Value
[31:4]	-	-	Reserved	-
3	OSDW4En	R/W	Control OSD window 4 0: Disable 1: Enable	0x0
2	OSDW3En	R/W	Control OSD window 3 0: Disable 1: Enable	0x0
1	OSDW2En	R/W	Control OSD window 2 0: Disable 1: Enable	0x0
0	OSDW1En	R/W	Control OSD window 1 0: Disable 1: Enable	0x0

6.1.5.50. OSD Font Control Register 0

Table 6-61. OSD Font Control Register 0 (osd_fontctrl0, Offset 0x2004)

Bit	Name	Type	Description	Default Value
[31:24]	-	-	Reserved	-
[23:20]	ShColorSel	R/W	Shadow color selection	0x0
[19:16]	BdColorSel	R/W	Border color selection	0x0
15	-	-	Reserved	-
[14:13]	HighLightType	R/W	Highlight type selection	0x0

			00: Normal 01: Border 10: Shadow 11: Reserved	
[12:10]	TranType	R/W	Transparency type selection 000: 0% 001: 25% 010: 50% 011: 75% 100: 100% Others: Reserved	0x0
9	HzoomOutMod	R/W	Select the zoom-out mode in the horizontal direction 0: Normal 1: 0.5x	0x0
[8:7]	HzoomInMod	R/W	Select the zoom-in mode in the horizontal direction 00: Normal 01: 2x 10: 3x 11: 4x	0x0
6	HzoomMod	R/W	Control the zoom-in and zoom-out modes in the horizontal direction 0: Zoom-out 1: Zoom-in	0x0
5	HzoomEn	R/W	Enable the zoom-in and zoom-out modes in the horizontal direction 0: Off 1: On	0x0
4	VzoomOutMod	R/W	Select the zoom-out mode in the vertical direction 0: Normal 1: 0.5x	0x0
[3:2]	VzoomInMod	R/W	Select the zoom-in mode in the vertical direction 00: Normal 01: 2x 10: 3x 11: 4x	0x0
1	VzoomMod	R/W	Select zoom-in or zoom-out mode in the vertical direction	0x0

Vangotech

			0: Zoom-out 1: Zoom-in	
0	VzoomEn	R/W	Control the zoom-in or zoom-out mode in the vertical direction 0: Off 1: On	0x0

6.1.5.51. OSD Font Control Register 1

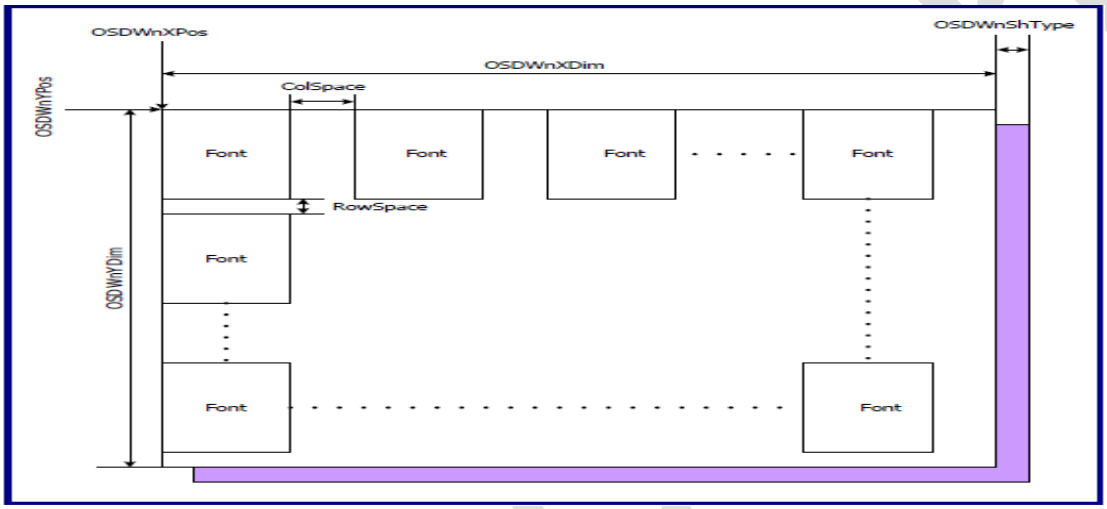


Table 6-62. OSD Font Control Register 1 (osd_fontctrl1, Offset 0x2008)

Bit	Name	Type	Description	Default Value
[31:11]	-	-	Reserved	-
[12:9]	ColSpace	R/W	Column space, 0 ~ 15 Note: When using the horizontal zoom-out mode (x0.5), the column space must be enabled and set from 7 to 15 pixels.	0x0
8	ColSpaceEn	R/W	Column space enable 0: Off 1: On	0x0
[7:5]	-	-	Reserved	0x0
[4:1]	RowSpace	R/W	Row space, 0 ~ 15	0x0
0	RowSpaceEn	R/W	Row space enable 0: Off 1: On	0x0

6.1.5.52. OSD Windows 1 to 4 Control Register 0 (Offset 0x200C, 0x2018, 0x2024, 0x2030)

Table 6-63 lists and describes the contents of the OSD window 1. OSD windows 2 to 4 are not listed below for having the same configurations.

Table 6-63. OSD Windows 1 to 4 Control Register 0

Bit	Name	Type	Description
[31:16]	-	-	Reserved
[15:8]	OSDW1Ypos	R/W	Specify the vertical position of the OSD window 1 The unit is font-based. The actual position is OSDW1Ypos +1.
[7:0]	OSDW1Xpos	R/W	Specify the horizontal position of the OSD window 1 The unit is font-based. The actual position is OSDW1Xpos +1.

6.1.5.53. OSD Windows 1 to 4 Control Register 1 (Offset 0x2010, 0x201C, 0x2028, 0x2034)

Table 6-64 lists and describes the contents for the OSD window 1. OSD windows 2 to 4 are not shown here, because they have the same configurations.

Table 6-64. OSD Windows 1 to 4 Control Register 1

Bit	Name	Type	Description
[31:30]	-	-	Reserved
[29:26]	OSDW1ShColor	R/W	Select the color of the OSD window 1 shadow effect from palette
[25:24]	OSDW1ShType	R/W	Select the width of the OSD window 1 shadow effect (Unit: Pixel) 00: 0 01: 1 10: 2 11: 3
[23:20]	OSDW1BdColor	R/W	Select the color of the OSD window 1 border effect from palette
[19:18]	OSDW1BdType	R/W	Select the width of the OSD window 1 border effect (Unit: Pixel) 00: 0 01: 1 10: 2 11: 3
17	OSDW1HLSel	R/W	Select the window highlight effect for the OSD window 1 0: Border 1: Shadow
16	OSDW1HLEn	R/W	Enable the window highlight effect for the OSD window 1

Vangotech

			0: Off 1: On
15	-	-	Reserved
[14:8]	OSDW1Ydim	R/W	Specify the vertical dimension of the OSD window 1. The unit is font-based, and the valid value is ranging from 1 to 64 of the normal fonts.
7	-	-	Reserved
[6:0]	OSDW1Xdim	R/W	Specify the horizontal dimension of the OSD window 1. The unit is font-based, and the valid value is ranging from 1 to 64 of the normal fonts.

6.1.5.54. OSD Window 1 to 4 Control Register 2 (Offset 0x2014, 0x2020, 0x202C, 0x2038)

Table 6-65 lists the contents for the OSD window 1. OSD windows 2 to 4 are not listed below for having the same configurations.

Table 6-65. OSD Windows 1 to 4 Control Register 2

Bit	Name	Type	Description
[31:9]	-	-	Reserved
[8:0]	OSDW1FontBase	R/W	This field shows the start address of display RAM for the OSD window 1

6.1.5.55. OSD Multi Multi-Color Font Base Address (Offset 0x203C)

Table 6-66. OSD Multi-Color Font Base Address

Bit	Name	Type	Description
[31:14]	-	-	Reserved
[13:0]	OSDMCFBaseNo	R/W	This field shows the starting font number of a multi-color font in the font-RAM. If the address is set to 0x200, the MCF font is not required for programming. When this number is multiplied by 18, it equals to the MCF physical base address.

6.1.5.56. OSD Palette Color Registers 0 ~ 15 (Offset 0x2040 ~ 0x207C)

The OSD design contains 16 palette colors that can be programmed. The tables below list and describe the contents of entry 0 in the palette table. Please note that the content of the palette must be programmed to Y, Cb, and Cr components if FTLCDC210_SHARP or FTLCDC210_COLOR is defined.

Table 6-67. OSD Palette Color Registers 0 ~ 15

Bit	Name	Type	Description

Vangotech

-	-	-	Reserved
[23:16]	PaletteColorR0	R/W	Palette color 0, Y component
[15:8]	PaletteColorG0	R/W	Palette color 0, Cb component
[7:0]	PaletteColorB0	R/W	Palette color 0, Cr component

Offset	Register	Offset	Register
0x2040	Palette Color 0	0x2060	Palette Color 8
0x2044	Palette Color 1	0x2064	Palette Color 9
0x0248	Palette Color 2	0x2068	Palette Color 10
0x204C	Palette Color 3	0x206C	Palette Color 11
0x2050	Palette Color 4	0x2070	Palette Color 12
0x2054	Palette Color 5	0x2074	Palette Color 13
0x2058	Palette Color 6	0x2078	Palette Color 14
0x205C	Palette Color 7	0x207C	Palette Color 15

6.1.5.57. OSD Font RAM Data (Offset 0x10000 ~ 0x147FC)

The database for each font is stored as a 12x18 matrix in font RAM. The data bus width for font RAM is 12-bits. One font utilizes 18 entries of font RAM. FTLCDC210 supports up to 256 single-color fonts during the RAM range. If the content data bit equals to one, it refers to the foreground, and if the content data bit equals to zero, it refers to the background. A simple example for a font, '1,' is shown in Table 3-90.

Table 6-68. 2×18 Font Matrix for Font '1'

	11	10	9	8	7	6	5	4	3	2	1	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
2	0	0	0	0	0	1	1	0	0	0	0	0
3	0	0	0	0	1	1	1	0	0	0	0	0
4	0	0	0	1	1	1	1	0	0	0	0	0
5	0	0	0	0	0	1	1	0	0	0	0	0
6	0	0	0	0	0	1	1	0	0	0	0	0
7	0	0	0	0	0	1	1	0	0	0	0	0
8	0	0	0	0	0	1	1	0	0	0	0	0
9	0	0	0	0	0	1	1	0	0	0	0	0
10	0	0	0	0	0	1	1	0	0	0	0	0
11	0	0	0	0	0	1	1	0	0	0	0	0
12	0	0	0	0	0	1	1	0	0	0	0	0
13	0	0	0	0	0	1	1	0	0	0	0	0
14	0	0	0	1	1	1	1	1	1	0	0	0
15	0	0	0	1	1	1	1	1	1	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0

Vangotech

The 12x18 font matrix can be partitioned into 18 12x1 blocks, labeled 0 ~ 17, that the real storing data order follows the block order in the font-RAM.

Table 6-69. Partial Content of Font-RAM Format

Address Offset	Bit	Type	Description											
			D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0x10000(0)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10004(1)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10008(2)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1000C(3)	[11:0]	R/W	0	0	0	0	1	1	1	0	0	0	0	0
0x10010(4)	[11:0]	R/W	0	0	0	1	1	1	1	0	0	0	0	0
0x10014(5)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10018(6)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1001C(7)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10020(8)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10024(9)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10028(10)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1002C(11)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10030(12)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10034(13)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10038(14)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x1003C(15)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x10040(16)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10044(17)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0

For Multi-Color Font (MCF) issue

Table 6-70. 12×18 Multi-color Font Matrix for Font '1'

	11	10	9	8	7	6	5	4	3	2	1	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
2	0	0	0	0	0	1	1	0	0	0	0	0
3	0	0	0	0	1	1	1	0	0	0	0	0
4	0	0	0	1	1	1	1	0	0	0	0	0
5	0	0	0	0	0	1	1	0	0	0	0	0
6	0	0	0	0	0	1	1	0	0	0	0	0
7	0	0	0	0	0	1	1	0	0	0	0	0
8	0	0	0	0	0	1	1	0	0	0	0	0
9	0	0	0	0	0	1	1	0	0	0	0	0
10	0	0	0	0	0	1	1	0	0	0	0	0
11	0	0	0	0	0	1	1	0	0	0	0	0
12	0	0	0	0	0	1	1	0	0	0	0	0
13	0	0	0	0	0	1	1	0	0	0	0	0

Vangotech

14	0	0	0	1	1	1	1	1	1	0	0	0
15	0	0	0	1	1	1	1	1	1	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0

The 12x18 multi-color font matrix can be placed into three standard font memories.

Table 6-71. Partial Content of Font-RAM Format

Address Offset	Bit	Type	Description											
			D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0x10000 (0)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10004 (1)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10008 (2)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1000C (3)	[11:0]	R/W	0	0	0	0	1	1	1	0	0	0	0	0
0x10010 (4)	[11:0]	R/W	0	0	0	1	1	1	1	0	0	0	0	0
0x10014 (5)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10018 (6)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1001C (7)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10020 (8)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10024(9)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10028(10)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1002C(11)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10030 (12)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10034 (13)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10038 (14)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1003C (15)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10040 (16)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10044 (17)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10048 (0)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1004C(2)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10050 (2)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10054 (3)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10058 (4)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1005C (5)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10060 (6)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10064 (7)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0

0x10068 (8)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1006C (9)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10070(10)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10074 (11)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10078 (12)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x1007C (13)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x10080 (14)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10084 (15)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10088 (16)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1008C (17)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10090 (0)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10094(1)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x10098 (2)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x1009C (3)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100A0 (4)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100A4 (5)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100A8 (6)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100AC (7)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100B0 (8)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100B4 (9)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100B8 (10)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100BC (11)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100C0 (12)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100C4 (13)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x100C8 (14)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x100CC (15)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x100D0 (16)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0

0x100D4 (17)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0

6.1.5.58. OSD Display Display-RAM Data (Offset 0x20000 ~ 0x207FC)

Fonts are used to build an OSD window. The font dimension can be defined by programming the associated registers. Each font in the window owns a unique attribute to describe its behavior. The format is an 18-bit wide entry and saved in a display RAM. The offset address of the first entry is 0x20000 and the last entry is 0x207FC. It is possible to display an OSD window with up to 512 fonts regardless of the window aspect.

Table 6-72. OSD Display-RAM Data

Address	Bit	TYPE	Description																	
			D17	**D16**	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
0x20000	[17:0]	R/W	Font index of 1st fon										Foreground palette index of the 1st font				Background palette index of the 1st font			
……..	….	….	……..										…..				…..			
0x207FC	[17:0]	R/W	Font index of last font										Foreground palette index of the last font				Background palette index of the last font			

6.1.5.59. OSD Dimension Control (Offset 0x2000)

Table 6-73. OSD Dimension Control

Bit	Name	Type	Description
[31:21]	-	-	Reserved
20	OSDUpdate	R/W	When this bit is set, FTLCDC210 will start updating all OSD dimensions and position registers. This bit will be auto cleared when registers are updated.
[19:18]	Reserved	-	-
[17:12]	OSDHdim	R/W	Specify the horizontal dimension of the OSD window; that is, the total font number of each row
[11:9]	-	-	Reserved
[8:4]	OSDVdim	R/W	Specify the vertical dimension of the OSD window; that is, the total vertical font number
[3:0]	Reserved	-	-

Vangotech

6.1.5.60. OSD Position Control (Offset 0x2004)

Table 6-74. OSD Position Control

Bit	Name	Type	Description
[31:24]	-	-	Reserved
[23:12]	OSDHPos	R/W	Specify the horizontal position of the OSD window. The reference original position is the left boundary of the LCD screen. The valid value is ranging from 1 to 1280. However, a fixed offset value of '3' between OSDHPos and the actual window position is recommended.
11	-	-	Reserved
[10:0]	OSDVPos	R/W	Specify the vertical position of the OSD window The reference original position is the upper boundary of the LCD screen. The valid value is ranging from 1 to 1280.

6.1.5.61. OSD Foreground Color Control (Offset 0x2008)

Please note that the content of the palette must be programmed to Y, Cb, and Cr components if FTLCDC210_SHARP or FTLCDC210_COLOR is defined.

Table 6-75. OSD Foreground Color Control

Bit	Name	Type	Description
[31:24]	OSDFrPal3	R/W	OSD foreground color palette entry 3 The format can be set below: {D_{31-30}, 6{& D_{31-30}}} forms MSB of the Y component. {D_{29-27}, 5{& D_{29-27}}} forms MSB of the Cb component. {D_{26-24}, 5{& D_{26-24}}} forms MSB of the Cr component.
[23:16]	OSDFrPal2	R/W	OSD foreground color palette entry 2 The operation is similar to entry 3.
[15:8]	OSDFrPal1	R/W	OSD foreground color palette entry 1 The operation is similar to entry 3.
[7:0]	OSDFrPal0	R/W	OSD foreground color palette entry 0 The operation is similar to entry 3.

6.1.5.62. OSD Background Color C ontrol (Offset 0x200C)

Please note that the content of the palette must be programmed to Y, Cb, and Cr components if

Vangotech

FTLCDC210_SHARP or FTLCDC210_COLOR is defined.

Table 6-76. OSD Background Color Control

Bit	Name	Type	Description
[31:24]	OSDBgPal3	R/W	OSD background color palette entry 3. The format can be set below: {D_{31-30}, 6{& D_{31-30}}} forms MSB of the Y component. {D_{29-27}, 5{& D_{29-27}}} forms MSB of the Cb component. {D_{26-24}, 5{& D_{26-24}}} forms MSB of the Cr component.
[23:16]	OSDBgPal2	R/W	OSD background color palette entry 2 The operation is similar to entry 3.
[15:8]	OSDBgPal1	R/W	OSD background color palette entry 1 The operation is similar to entry 3.
[7:2]	-	-	Reserved
[1:0]	OSDTrans	R/W	OSD background transparency control 00: 25% transparency 01: 50% transparency 10: 75% transparency 11: 100% transparency

6.1.5.63. OSD Font Database Write Accessing Port (Offset 0x8000 ~ 0xBFFC)

Each font database is stored as a 12x16 matrix in font RAM. The RAM is organized as 12-bit in width. Therefore, it requires 16 entries (192 bits) to record the font content. A bit that equals 1 refers to the foreground, and a bit that equals 0 refers to background. The total size of a font memory is 4Kx12, so the total number of fonts that can be loaded is 256. There is an example for a font, '1' at the location 0x8000 ~ 0x803C

Table 6-77. OSD Font Database Write Accessing Port

Address Offset	Bit	Type	Description											
			D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0x8000 (0)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0
0x8004 (1)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8008 (2)	[11:0]	R/W	0	0	0	0	1	1	1	0	0	0	0	0
0x800C (3)	[11:0]	R/W	0	0	0	1	1	1	1	0	0	0	0	0
0x8010 (4)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8014 (5)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8018 (6)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x801C (7)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8020 (8)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8024 (9)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0

0x8028 (10)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x802C (11)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8030 (12)	[11:0]	R/W	0	0	0	0	0	1	1	0	0	0	0	0
0x8034 (13)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x8038 (14)	[11:0]	R/W	0	0	0	1	1	1	1	1	1	0	0	0
0x803C (15)	[11:0]	R/W	0	0	0	0	0	0	0	0	0	0	0	0

6.1.5.64. OSD Window Attribute Write Accessing Port (Offset 0xC000H ~ 0xC7FC)

Fonts are used to build an OSD window. The font dimensions can be defined by programming the associated registers. Each font in the window owns a unique attribute to describe its behavior. The format is a 12-bit wide entry, as mentioned earlier. These attributes are saved in an attribute RAM and are sequentially placed, starting from the up-left corner, so that it can move from left to right and row by row. The offset address of the first attribute entry is 0xC000, and the last is 0xC7FC. Therefore, it is possible to display an OSD window with up to 512 fonts, regardless of the window aspect. Since the total font variety is 256, some fonts will be reused to build such a big window.

Table 6-78. OSD Window Attribute Write Accessing Port

Address Offset	**Bit**	**Type**	**Description**											
			D_{11}	**D_{10}**	**D_9**	**D_8**	**D_7**	**D_6**	**D_5**	**D_4**	**D_3**	**D_2**	**D_1**	**D_0**
0xC000	[11:0]	W	Font index of the 1st font								Foreground palette index of the 1st font		Background palette index of the 1st font	
……	….	….	……								……		……	
0xC7FC	[11:0]	W	Font index of the last font								Foreground palette index of the last font		Background palette index of the last font	

Vangotech

Chapter 7. Audio

This section details the interfaces that provided in V8130L, mainly includes:

- I2S

Vangotech

7.1. I2S

The detailed information of the I2S control registers, is same to SPI, provided in Chapter 8.

Vangotech

Chapter 8. Interface

This section details the interfaces that provided in V8130L, mainly includes:

-

Vangotech

USB2.0

- SSP
- SPIF
- SPI2AHB
- I2C
- UART
- MAC
- GPIO
- CAN
- KBC

Vangotech

8.1. USB2.0

8.1.1. Overview

FOTG210 is an USB 2.0 OTG controller, which can be a host controller or a peripheral controller. When acting as the host controller, it contains a USB host controller that supports transactions at all speeds. Without software intervention, the host controller can deal with the transaction-based data structure to offload CPU and automatically transmit and receive data on the USB bus. When acting as the peripheral controller, each endpoint, except endpoint 0, accepts the programmable HS/FS transfer types to provide a flexibility to suit all applications. In addition, complying with the OTG standards means that both Session Request Protocol (SRP) and Host Negotiation Protocol (HNP) are supported. The system bus can be an AXI 32-bit or 64-bit bus interface or AHB 32-bit bus interface. The transceiver interface is UTMI+ Level 3, which supports the high-speed, full-speed, and low-speed transfers with a high-speed or full-speed hub.

8.1.2. Features

- Compliant with USB Specification, Revision 2.0
- Compliant with On-The-Go supplement to USB 2.0 Specification, Revision 1.3
- Supports UTMI+ Level 3 compliant transceiver
- Compatible with EHCI 1.0
- AMBA AXI 64-bit data bus width interface
- Can be host or device.

  OTG for Dual role: The application will fix the device to Host or Device.Their USB role is fixed at each run time (decide by ID pin only, ID pin set 0 as host , set 1 as device)
- Supports point-to-point communications with one high-speed, full-speed, or low-speed device
- Supports high-speed or full-speed hub
- Supports configurable endpoints as HS/FS device
- Supports isochronous/interrupt/control/bulk transfers for both host and device
- Compatible with EHCI data structures (FSTN and SITD back points are not supported.)
- Supports embedded DMA access to FIFO
- Supports SRAM interface for FIFO
- Supports suspend, remote wake-up, and resume functions
- USB OTG software subsystem
- Configures FIFO to 4 kB

8.1.3. Block Diagram

FOTG210 contains the following functional blocks:

- DMA controller
- Host controller
- Host register file
- FIFO controller
- Device register file
- OTG register file

Vangotech

- Device controller
- Parallel interface engine
- OTG bus monitor

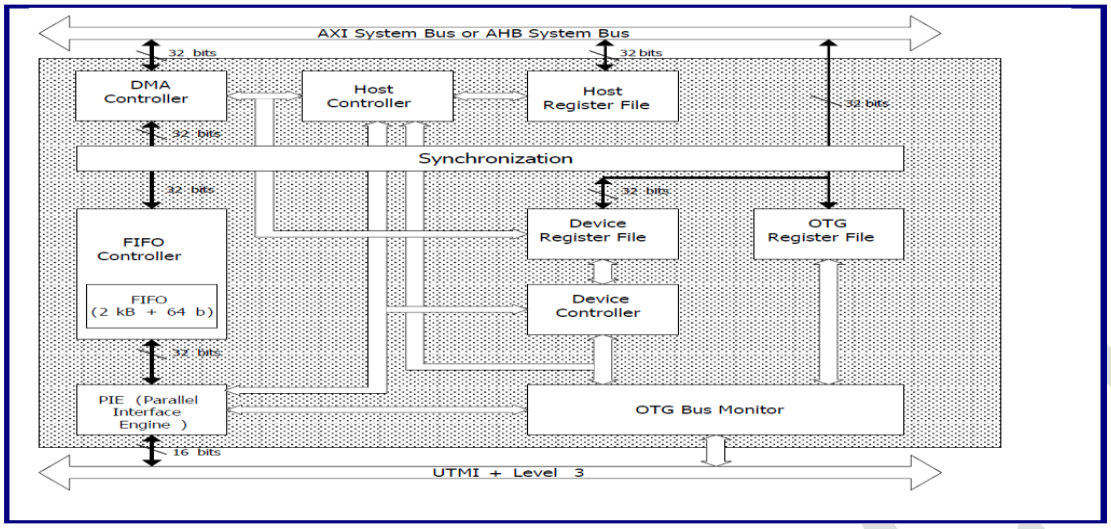


Figure 8-1. Functional Block Diagram

8.1.4. Timing Diagram

Please refer USB2.0 Specification and its On-The-Go Supplement to the USB 2.0 Specification.

8.1.5. Register List

Module Name	Base Address
OTG_0	0x1020 0000
OTG_1	0x1030 0000
OTG_2	0x1040 0000

Register Name	Offset	Description
HCCAP	000	HC Capability Register
HCSPARAMS	004	HC Structural Parameters
HCCPARAMS	008	HC Capability Parameters
USBCMD	010	HC USB Command Register
USBSTS	014	HC USB Status Register
USBINTR	018	HC USB Interrupt Enable Register
FRINDEX	01C	HC Frame Index Register
PERIODICLISTBASE	024	HC Periodic Frame List Base Address Register
ASYNCLISTADDR	028	HC Current Asynchronous List Address Register
PORTSC	030	HC Port Status and Control Register
HCMISC	040	HC Miscellaneous Register
FS_EOF	044	HC Full-Speed (FS) EOF1 Timing Point Register
HS_EOF	048	HC High-Speed (HS) EOF1 Timing Point Register
OTG_CSR	080	OTG Control Status Register
OTG_ISR	084	OTG Interrupt Status Register

OTG_IER	088	OTG Interrupt Enable Register
GLB_ISR	0C0	Global HC/DEV Interrupt Status Register
GLB_INT	0C4	Global Mask of HC/DEV Interrupt Register
REVISION	0E0	Revision Information Register
FEATURE	0E4	Configured Information Register
DEV_CTL	100	Device Main Control Register
DEV_ADR	104	Device Address Register
DEV_TST	108	Device Test Register
DEV_SFN	10C	Device SOF Frame Number Register
DEV_SMT	110	Device SOF Mask Timer Register
PHY_TST	114	PHY Test Mode Selector Register
DEV_VCTL	118	Device Vendor-Specific I/O Control Register
DEV_CXCFG	11C	Device CX Configuration Status Register
DEV_CXCFE	120	Device CX Configuration and FIFO Empty Status Register
DEV_ICR	124	Device Idle Counter Register
DEV_MIGR	130	Device Mask of Interrupt Group Register
DEV_MISG0	134	Device Mask of Interrupt Source Group 0 Register
DEV_MISG1	138	Device Mask of Interrupt Source Group 1 Register
DEV_MISG2	13C	Device Mask of Interrupt Source Group 2 Register
DEV_IGR	140	Device Interrupt Group Register
DEV_ISG0	144	Device Interrupt Source Group 0 Register
DEV_ISG1	148	Device Interrupt Source Group 1 Register
DEV_ISG2	14C	Device Interrupt Source Group 2 Register
DEV_RXZ	150	Device Receive Zero-Length Data Packet Register
DEV_TXZ	154	Device Transfer Zero-length Data Packet Register
DEV_ISE	158	Device Isochronous Sequential Error/Abort Register
DEV_INMPS	160+ (n-1)*004	Device IN Endpoint n MaxPacketSize Register (n = 1 ~ 8)
DEV_OUTMPS	180+ (n-1)*004	Device OUT Endpoint n MaxPacketSize Register (n = 1 ~ 8)
DEV_EPMAP0	1A0	Device Endpoint 1 ~ 4 Map Register
DEV_EPMAP1	1A4	Device Endpoint 5 ~ 8 Map Register
DEV_FMAP	1A8	Device FIFO Map Register
DEV_FCFG	1AC	Device FIFO Configuration Register
DEV_FIBC	1B0+ n*004	Device FIFO n Instruction and Byte Count Register (n = 0 ~ 3)
DMA_TFN	1C0	Device DMA Target FIFO Number Register
DMA_CPS0	1C4	Device DMA Controller Parameter Setting 0 Register
DMA_CPS1	1C8	Device DMA Controller Parameter Setting 1 Register
DMA_CPS2	1CC	Device DMA Controller Parameter Setting 2 Register
DMA_CPS3	1D0	Device DMA Controller Parameter Setting 3 Register

Vangotech

VDMA_CXFPS1	300	Device Virtual DMA CXF Parameter Setting 1 Register
VDMA_CXFPS2	304	Device Virtual DMA CXF Parameter Setting 2 Register
VDMA_F0PS1	308	Device Virtual DMA FIFO0 Parameter Setting 1 Register
VDMA_F0PS2	30C	Device Virtual DMA FIFO0 Parameter Setting 2 Register
VDMA_F1PS1	310	Device Virtual DMA FIFO1 Parameter Setting 1 Register
VDMA_F1PS2	314	Device Virtual DMA FIFO1 Parameter Setting 2 Register
VDMA_F2PS1	318	Device Virtual DMA FIFO2 Parameter Setting 1 Register
VDMA_F2PS2	31C	Device Virtual DMA FIFO2 Parameter Setting 2 Register
VDMA_F3PS1	320	Device Virtual DMA FIFO3 Parameter Setting 1 Register
VDMA_F3PS2	324	Device Virtual DMA FIFO3 Parameter Setting 2 Register
DEV_ISG3	328	Device Interrupt Source Group 3 Register
DEV_MISG3	32C	Device Mask of Interrupt Source Group 3 Register
VDMA_CTRL	330	Device Virtual DMA Control Register
LPM_CAP	334	Device LPM Capability Register
DEV_ISG4	338	Device Interrupt Source Group 4 Register
DEV_MISG4	33C	Device Mask of Interrupt Source Group 4 Register

Note: When disabled UCLK registers in the UCLK clock domain are inaccessible, both read and write operations are not allowed at this moment.

The notations listed below are used to indicate whether theregister isvalid in the host ONLY or device ONLY mode.

(HOV)	**Host Only Valid** This register will be valid even if FOTG210 only operates as a host. In the host ONLY mode, the HC registers and global registers are fully accessible. But the device and OTG register spaces are partial valid. Only the registers with the HOV notation can be accessed.
(POV)	**Peripheral Only Valid** This register will be valid even if FOTG210 only operates as a peripheral. In the device ONLY mode, the device registers and global registers are fully accessible. But the HC and OTG register spaces are partial valid. Only the registers with POV notation can be accessed.

The non-HOV device registers and non-POV host registers are undefined if FOTG210 is configured to host ONLY and device ONLY, respectively. Any access to the undefined address might cause unpredictable consequences.

8.1.6. Register Description

8.1.6.1. HC Capability Register

Table 8-1. HC Capability Register (HCCAP, Address = 0x000)

Bit	Name	Type	Description	Default Value
[31:16]	HCIVERSION	RO	**Host Controller Interface Version Number**	0x0100

Bit	Name	Type	Description	Default Value
			This is a 2-byte register containing a BCD encoding of the EHCI revision number supported by the host controller.	
[15:8]	-	Rsvd	Reserved	0
[7:0]	CAPLENGTH	RO	**Capability Register Length** This register is used as an offset to add to the register base to find out the beginning of the operational register space.	0x10

8.1.6.2. HC Structural Parameters

Table 8-2. HC Structural Parameters (HCSPARAMS, Address = 0x004)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0
[3:0]	N_PORTS	RO	**Number of Ports** This field specifies the number of the physical downstream ports implemented on the host controller.	0x1

8.1.6.3. HC Capability Parameters

Table 8-3. HC Capability Parameters (HCCPARAMS, Address = 0x008)

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
2	ASYN_SCH_PARK_CAP	RO	**Asynchronous Schedule Park Capability** The host controller supports the park feature for the high-speed queue heads in the asynchronous schedule. This feature can be disabled or enabled and can be set to a specific level by using the Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count fields in the USBCMD register.	0
1	PROG_FR_LIST_FLAG	RO	**Programmable Frame List Flag** When this bit is set to 1, the system software can specify and use a smaller frame list to configure the host controller via the Frame List Size field of the USBCMD register. This requirement ensures that the frame list is always physically contiguous.	1

Vangotech

Bit	Name	Type	Description	Default Value
0	-	Rsvd	Reserved	0

8.1.6.4. HC USB Command Register

Table 8-4. HC USB Command Register (USBCMD, Address = 0x010)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:16]	INT_THRC	RW	**Interrupt Threshold Control** This field is used by the system software to select the maximum rate at which the host controller will issue an interrupt. The valid values are defined as follows: Value / Maximum high-speed interrupt interval 0x00 / Reserved 0x01 / 1 micro frame 0x02 / 2 micro frames 0x04 / 4 micro frames 0x08 / 8 micro frames (Default, equals to 1 ms) 0x10 / 16 micro frames (2 ms) 0x20 / 32 micro frames (4 ms) 0x40 / 64 micro frames (8 ms) Note: In the full-speed mode, these registers are all reserved.	0x08
[15:12]	-	Rsvd	Reserved	0
11	ASYN_PK_EN	RW	Asynchronous Schedule Park Mode Enable The software uses this bit to enable or disable the Park mode. When this bit is set to '1', the Park mode will be enabled.	1
10	-	Rsvd	Reserved	0
[9:8]	ASYN_PK_CNT	RW	Asynchronous Schedule Park Mode Count This field contains the number of successive transactions allowed by the host controller to	0b11

Vangotech

			execute from a high-speed queue head on the asynchronous schedule.	
7	-	Rsvd	Reserved	0
6	INT_OAAD	RW	Interrupt on Asynchronous Advance Doorbell This bit is used as a doorbell by software to inform the host controller to issue an interrupt next time when an asynchronous schedule is advanced.	0
5	ASCH_EN	RW	Asynchronous Schedule Enable This bit controls the host controller to skip processing the asynchronous schedule. 0: Do not process the asynchronous schedule. 1: Use the ASYNCLISTADDR register to access the asynchronous schedule.	0
4	PSCH_EN	RW	Periodic Schedule Enable This bit controls the host controller to skip processing the periodic schedule. 0: Do not process the periodic schedule. 1: Use the PERIODICKISTBASE register to access the periodic schedule.	0
[3:2]	FRL_SIZE	RW	Frame List Size This field specifies the size of the frame list. 0b00: 1024 elements (4096 bytes, default value) 0b01: 512 elements (2048 bytes) 0b10: 256 elements (1024 bytes) 0b11: Reserved	00
1	HC_RESET	RW	Host Controller Reset This bit is used by software to reset the host controller.	0
0	RS	RW	Run/Stop When this bit is set to '1', the host controller will proceed with the execution of schedule. 0: Stop 1: Run	0

8.1.6.5. HC USB Status Register

Table 8-5. HC USB Status Register (USBSTS, Address = 0x014)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0

Vangotech

15	ASCH_STS	RO	Asynchronous Schedule Status This bit reports the actual status of asynchronous schedule.	0
14	PSCH_STS	RO	Periodic Schedule Status This bit reports the actual status of periodic schedule.	0
13	Reclamation	RO	Reclamation This is a read-only status bit, which is used to detect an empty asynchronous schedule.	0
12	HCHalted	RO	Host Controller Halted This bit will be zero whenever the Run/Stop bit is set to '1'. The host controller will set this bit to 1 after the execution is stopped. As a result of setting the Run/Stop bit to 0b.	1
[11:6]	-	Rsvd	Reserved	0
5	INT_OAA	RW1C	Interrupt on Async. Advance This status bit indicates the assertion of an interrupt on Async. Advance Doorbell.	0
4	H_SYSERR	RW1C	Host System Error The host controller will set this bit to '1' when a serious error occurs during a host system access involving the host controller module.	0
3	FRL_ROL	RW1C	Frame List Rollover The host controller will set this bit to '1' when the Frame List Index changes from its maximum value to zero.	0
2	PO_CHG_DET	RW1C	Port Change Detect The host controller will set this bit to '1' when any port has a change bit transiting from '0' to '1'. This bit is loaded with OR of all the PORTSC change bits.	0
1	USBERR_INT	RW1C	USB Error Interrupt The host controller will set this bit to '1' upon the completion of a USB transaction, resulting in an error condition.	0
0	USB_INT	RW1C	USB Interrupt The host controller will set this bit to '1' upon the completion of a USB transaction.	0

8.1.6.6. HC USB Interrupt Enable Register

Table 8-6. HC USB Interrupt Enable Register (USBINTR, Address = 0x018)

Vangotech

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0
5	INT_OAA_EN	RW	Interrupt on Async. Advance Enable When this bit is '1' and the Interrupt on Async. Advance bit in USBSTS register is '1', the host controller will issue an interrupt at the next interrupt threshold.	0
4	H_SYSERR_EN	RW	Host System Error Enable When this bit is '1' and the Host System Error Status bit in USBSTS register is '1', the host controller will issue an interrupt.	0
3	FRL_ROL_EN	RW	Frame List Rollover Enable When this bit is '1' and the Frame List Rollover bit in USBSTS register is '1', the host controller will issue an interrupt.	0
2	PO_CHG_INT_EN	RW	Port Change Interrupt Enable When this bit is '1' and the Port Change Detect bit in USBSTS register is '1', the host controller will issue an interrupt.	0
1	USBERR_INT_EN	RW	USB Error Interrupt Enable When this bit is'1' and the USBERRINT bit in USBSTS register is '1', the host controller will issue an interrupt at the next interrupt threshold.	0
0	USB_INT_EN	RW	USB Interrupt Enable When this bit is '1' and the USBINT bit in USBSTS register is '1', the host controller will issue an interrupt at the next interrupt threshold.	0

8.1.6.7. HC Frame Index Register

Table 8-7. HC Frame Index Register (FRINDEX, Address = 0x01C)

Bit	Name	Type	Description	Default Value
[31:14]	-	Rsvd	Reserved	0
[13:0]	FRINDEX	RW	Frame Index This register is used by the host controller to index the frame into the periodic frame list. It updates every 125 microseconds. The HC Frame Index Register cannot be written unless	0

Vangotech

			the host controller is at the Halted state.	

8.1.6.8. HC Periodic Frame List Base Address Register

Table 8-8. HC Periodic Frame List Base Address Register (PERIODICLISTBASE, Address = 0x024)

Bit	Name	Type	Description	Default Value
[31:12]	PERI_BASADR	RW	Periodic Frame List Base Address This 32-bit register contains the beginning address of the periodic frame list in the system memory. These bits correspond to the memory address signals [31:12].	Undefined
[11:0]	-	Rsvd	Reserved	0

8.1.6.9. HC Current Asynchronous List Address Register

Table 8-9. HC Current Asynchronous List Address Register (ASYNCLISTADDR, Address = 0x028)

Bit	Name	Type	Description	Default Value
[31:5]	Async_ladr	RW	Current Asynchronous List Address This 32-bit register contains the address of the next asynchronous queue head to be executed. These bits correspond to the memory address signals [31:5].	Undefined
[4:0]	-	Rsvd	Reserved	0

8.1.6.10. HC Port Status and Control Register

Table 8-10. HC Port Status and Control Register (PORTSC, Address = 0x030)

Bit	Name	Type	Description	Default Value
[31:21]	-	Rsvd	Reserved	0
20	HC_TST_PKDONE	RW	Data Transfer is Done for Test Packet in Host Port Test Control The firmware has completely sent the whole test patterns to FIFO for a PHY test packet mode by writing '1' to this bit.	0
[19:16]	PORT_TEST	RW	Port Test Control When this field is zero, the port will not operate in the test mode. A non-zero value indicates that the port is operating in the test mode and the specific test mode is indicated by a specific value. The encoding of the test	0

<table>
<tr><td></td><td></td><td></td><td>mode bits are as follows (0b0110 ~ 0b1111 are reserved.):
<table>
<tr><th>Bit</th><th>Test Mode</th></tr>
<tr><td>0b0000</td><td>Test mode is not enabled</td></tr>
<tr><td>0b0001</td><td>Test J_STATE</td></tr>
<tr><td>0b0010</td><td>Test K-STATE</td></tr>
<tr><td>0b0011</td><td>Test SE0_NAK</td></tr>
<tr><td>0b0100</td><td>Test Packet</td></tr>
<tr><td>0b0101</td><td>Test FORCE_ENABLE</td></tr>
</table>
Note: When this signal is set to “0b0100” (Test packet), the test packet must be filled into FIFO by DMA first, and then set HC_TST_PKDONE to ‘1’.</td><td></td></tr>
<tr><td>[15:12]</td><td>-</td><td>Rsvd</td><td>Reserved</td><td>0</td></tr>
<tr><td>[11:10]</td><td>LINE_STS</td><td>RO</td><td>Line State
These bits reflect the current logical levels of the D+ and D- signal lines.
<table>
<tr><th>Bit[11:10]</th><th>USB State</th></tr>
<tr><td>0b00</td><td>SE0</td></tr>
<tr><td>0b10</td><td>J state</td></tr>
<tr><td>0b01</td><td>K state</td></tr>
<tr><td>0b11</td><td>Undefined</td></tr>
</table></td><td>HwInit</td></tr>
<tr><td>9</td><td>-</td><td>Rsvd</td><td>Reserved</td><td>0</td></tr>
<tr><td>8</td><td>PO_RESET</td><td>RW</td><td>Port Reset
1 = Port is reset.
0 = Port is not reset.
When software writes ‘1’ to this bit, the bus reset sequence defined in the USB specification will start. When software writes ‘0’ to this bit, the bus reset sequence will be terminated. Software must keep this bit at ‘1’ long enough to ensure the reset sequence.
Note: Before setting this bit, the RUN/STOP bit should be set to ‘0’.</td><td>0</td></tr>
<tr><td>7</td><td>PO_SUSP</td><td>RW</td><td>Port Suspend
1 = Port is at the suspend state.
0 = Port is not at the suspend state.
The Port Enable bit and the Suspend bit of this register define the port states as follows:
<table>
<tr><th>Bits[Port Enable,</th><th>Port State</th></tr>
</table></td><td>0</td></tr>
</table>

Vangotech

			Suspend] 0b0X — Disable 0b10 — Enable 0b11 — Suspend At the suspend state, the downstream data propagation will be blocked on this port, except the port reset. At the suspend state, the port will be sensitive to resume the detection. The host controller will ignore writing a zero to this bit. The host controller will unconditionally set this bit to '0' under the following conditions: ● Software sets the Force Port Resume bit to '0' (From '1'). ● Software sets the Port Reset bit to '1' (From '0'). Note: Before setting this bit, the RUN/STOP bit should be set to '0'.	
6	F_PO_RESM	RW	Force Port Resume 1 = Resume detected/driven on port 0 = No resume detected/driven on port Software will set this bit to '1' to resume signaling. The host controller will set this bit to '1' if a J-to-K transition is detected when the port is at the suspend state. When this bit transits to '1' because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register will also be set to '1'.	0
[5:4]	-	Rsvd	Reserved	0
3	PO_EN_CHG	RW1C	Port Enable/Disable Change 0 = No change 1 = Port enable/disable status is changed.	0
2	PO_EN	RW	Port Enable/Disable 0 = Disable 1 = Enable Ports can only be enabled by the host controller as part of the reset and enable. Software cannot enable a port by writing '1' to this field.	0
1	CONN_CHG	RW1C	Connect Status Change	0

Vangotech

			0 = Do not change the current status 1 = Change the current connect status This bit indicates that the current connect status is changed on the port.	
0	CONN_STS	RO	Current Connect Status 0 = No device is presented. 1 = Device is presented on the port. This bit reflects the current state of the port and may not directly correspond to the event that the connect status change bit is set. When TST_FORCEEN is set to '1', this signal will be the output of u_hdiscon.	0

8.1.6.11. HC Miscellaneous Register

Table 8-11. HC Miscellaneous Register (HCMISC, Address = 0x040)

Bit	Name	Type	Description	Default Value
[31:12]	-	Rsvd	Reserved	0
11	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
10	-	Rsvd	-	0
9	HC_CONN_DET_EN	RW	Host Connect Detect Enable This bit is used to enable the connective detection as SCLK is disabled. The sideband output, usbh_conn_det_n, goes low once a device is connected. 0 = Disable the connective detection 1 = Enable the connective detection	0
8	HC_WKP_DET_EN	RW	Host Wakeup Detect Enable This bit is used to enable the remote wakeup detection as SCLK is disabled. The sideband output, usbh_wkp_det_n, goes low once the attached device issues remote wakeup. 0 = Disable the remote wakeup detection 1 = Enable the remote wakeup detection	0
7	-	Rsvd	Reserved	0
6	HostPhy_Suspend	RW	Host Transceiver Suspend Mode Active high This bit places the transceiver in the	0

Vangotech

<table>
<tr><td></td><td></td><td></td><td>suspend mode that draws the minimal power from the power supplies.
This bit is only used in the host mode and is in the system clock domain instead of the UCLK clock domain.</td><td></td></tr>
<tr><td>[5:4]</td><td>EOF2_Time</td><td>RW</td><td>EOF 2 Timing Points
These bits control the EOF2 timing point before the next SOF.
<table>
<tr><th colspan="2">High-Speed EOF2 time</th></tr>
<tr><td>0b00</td><td>2 clocks (30 MHz) = 66 ns</td></tr>
<tr><td>0b01</td><td>4 clocks (30 MHz) = 133 ns</td></tr>
<tr><td>0b10</td><td>8 clocks (30 MHz) = 266 ns</td></tr>
<tr><td>0b11</td><td>16 clocks (30 MHz) = 533 ns</td></tr>
<tr><th colspan="2">Full-Speed EOF2 time</th></tr>
<tr><td>0b00</td><td>20 clocks (30 MHz) = 666 ns</td></tr>
<tr><td>0b01</td><td>40 clocks (30 MHz) = 1.33 μs</td></tr>
<tr><td>0b10</td><td>80 clocks (30 MHz) = 2.66 μs</td></tr>
<tr><td>0b11</td><td>160 clocks (30 MHz) = 5.33 μs</td></tr>
<tr><th colspan="2">Low-Speed EOF2 time</th></tr>
<tr><td>0b00</td><td>40 clocks (30 MHz) = 1.33 μs</td></tr>
<tr><td>0b01</td><td>80 clocks (30 MHz) = 2.66 μs</td></tr>
<tr><td>0b10</td><td>160 clocks (30 MHz) = 5.33 μs</td></tr>
<tr><td>0b11</td><td>320 clocks (30 MHz) = 10.66 μs</td></tr>
</table></td><td>0</td></tr>
<tr><td>[3:2]</td><td>EOF1_Time</td><td>RW</td><td>EOF 1 Timing Points
These bits control the EOF1 timing point before the next SOF. These bits should be adjusted according to the maximum packet size.
<table>
<tr><th colspan="2">High-Speed EOF1 time</th></tr>
<tr><td>0b00</td><td>540 clocks (30 MHz) = 18 μs</td></tr>
<tr><td>0b01</td><td>360 clocks (30 MHz) = 12 μs</td></tr>
<tr><td>0b10</td><td>180 clocks (30 MHz) = 6.0 μs</td></tr>
<tr><td>0b11</td><td>720 clocks (30 MHz) = 24 μs</td></tr>
<tr><th colspan="2">Full-Speed EOF1 time</th></tr>
<tr><td>0b00</td><td>1600 clocks (30 MHz) = 53.3 μs</td></tr>
<tr><td>0b01</td><td>1400 clocks (30 MHz) = 46.6 μs</td></tr>
<tr><td>0b10</td><td>1200 clocks (30 MHz) = 40 μs</td></tr>
<tr><td>0b11</td><td>21000 clocks (30 MHz) = 700 μs</td></tr>
<tr><th colspan="2">Low-Speed EOF1 time</th></tr>
<tr><td>0b00</td><td>3750 clocks (30 MHz) = 125 μs</td></tr>
<tr><td>0b01</td><td>3500 clocks (30 MHz) = 116 μs</td></tr>
</table></td><td>0</td></tr>
</table>

Vangotech

			0b10 3250 clocks (30 MHz) = 108 μs 0b11 4000 clocks (30 MHz) = 133 μs	
[1:0]	ASYN_SCH_SLPT	RW	Asynchronous Schedule Sleep Timer These bits control the sleep timer of the asynchronous schedule. 0b00 5.0 μs 0b01 10 μs 0b10 15 μs 0b11 20 μs	0b01

8.1.6.12. HC Full-Speed (FS) EOF1 Timing Point Register

This register is used to configure the Full-Speed (FS) EOF1 timing point before the next SOF instead of using the options of register 0x040 bits[3:2]. When these bits are non-zero, the EOF1 timing point will be set according to this register. Otherwise, this register will be disabled and the EOF1 timing point will be set according to the register 0x040 bits[3:2].

The formula of the Full-Speed (FS) EOF1 timing is showed as below:

FS EOF1 timing = (1 ms - (125 μs * FS_EOF[14:12] + (Clock period of UCLK) * FS_EOF[11:0]))

These bits should be adjusted according to the maximum packet size. Any improper configuration may cause the controller to hang.

Table 8-12. HC Full-Speed (FS) EOF1 Timing Point Register (FS_EOF, Address = 0x044)

Bit	Name	Type	Description	Default Value
[31:29]	QHTravSM	RO	QH Traversal State Machine 0b000: IDLE 0b001: Fetch QH 0b010: Advance QH 0b011: EXE 0b100: Write back qTD 0b101: Turnaround	0
[28:26]	QHtrSM	RO	QH Transfer Traversal State Machine 0b000: IDLE 0b001: Turnaround 0b010: Begin 0b011: Process 0b100: DMA	0
[25:22]	DMAdmSM	RO	DMA Data Movement State Machine 0h0: IDLE 0h1: Turnaround 0h4: Start1	0

			0h5: Start2 0h6: Start3 0h8: ALL 0h9: ONE 0ha: TWO 0hc: Write back	
[21:20]	iTDTravSM	RO	iTD Traversal State Machine 0b00: IDLE 0b01: Fetch iTD 0b10: EXE 0b11: Turnaround	0
[19:16]	iTDdmaSM	RO	iTD DMA Data Movement State Machine 0h0: IDLE 0h1: Turnaround 0h4: Start1 0h5: Start2 0h6: Start3 0h8: ALL 0h9: ONE 0ha: TWO 0hc: Write back	0
[15]	-	Rsvd	Reserved	0
[14:12]	FS_EOF1_Time_125us	RW	Full-Speed EOF 1 Timing Points (Unit: 125 μs) The unit of these bits is per 125 micro second.	1
[11:0]	FS_EOF1_Time	RW	Full-Speed EOF 1 Timing Points The unit of these bits is one clock period of UCLK. This value should not be greater than $\frac{125\ us}{UCLK\ period}$.	0

Vangotech

8.1.6.13. HC High-Speed (HS) EOF1 Timing Point Register

The formula of the High-Speed (HS) EOF1 timing is showed as below:

HS EOF1 timing = (125 μs - (Clock period of UCLK) * HS_EOF[11:0]))

These bits should be adjusted according to the maximum packet size. Any improper configuration may cause the controller to hang.

Table 8-13. HC High-Speed (HS) EOF1 Timing Point Register (HS_EOF, Address = 0x048)

Bit	Name	Type	Description	Default Value
[31:22]	-	Rsvd	Reserved	0
[21:20]	siTDtravSM	RO	siTD Traversal State Machine 0b00: IDLE 0b01: Fetch siTD 0b10: EXE 0b11: Turnaround	0
[19:16]	siTDdmaSM	RO	siTD DMA Data Movement State Machine 0h0: IDLE 0h1: Turnaround 0h4: Start1 0h5: Start2 0h6: Start3 0h8: ALL 0h9: ONE 0ha: TWO 0hc: Write back	0
[15:12]	-	Rsvd	Reserved	0
[11:0]	HS_EOF1_Time	RW	High-Speed EOF 1 Timing Points This register is used to configure the HS EOF1 timing point before the next SOF instead of using the options of the register 0x040 bits[3:2]. If these bits are non-zero, the EOF1 timing point will be set according to this register. Otherwise, this register will be disabled and the EOF1 timing point will be set according to the register 0x040 bits[3:2]. The unit of these bits is a clock period of UCLK. This value should not be greater than	0

Vangotech

			$\frac{125\ us}{UCLK\ period}$	

8.1.6.14. OTG Control Status Register

Table 8-14. OTG Control Status Register (OTG_CSR, Address = 0x080)

Bit	Name	Type	Description	Default Value
[31:29]	-	Rsvd	Reserved	0
[28:27]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
26	-	Rsvd	Reserved	0
[25:24]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
[23:22]	SPD_TYP (HOV, POV)	RO	Speed Type These bits indicate the current operating speed type. 0b10: HS 0b00: FS 0b01: LS 0b11: Reserved	0
21	ID (HOV, POV)	RO	Current ID This bit records the current ID of FOTG210. 0: A device 1: B device	1
20	CROLE (HOV, POV)	RO	Current Role This bit records the current role of FOTG210. 0: Host 1: Device	1
19	VBUS_VLD (HOV)	RO	Vbus Valid of A Device This bit indicates whether or not the voltage on	0

Vangotech

			Vbus is above the A-device Vbus valid threshold.	
18	A_SESS_VLD	RO	Session Valid of A Device This bit indicates whether or not the voltage on Vbus is above the A-device session valid threshold.	0
17	B_SESS_VLD (POV)	RO	Session Valid of B Device This bit indicates whether or not the voltage on Vbus is above the B-device session valid threshold.	0
16	B_SESS_END (POV)	RO	Session End of B Device This bit indicates whether or not the voltage on Vbus is below the B-device session end threshold.	1
[15:14]	-	RsvdP	Reserved and preserved. Software shall preserve the value read from these bits for writes to these bits.	0
13	IDPULUP (POV, HOV)	RW	ID pullup control This bit controls the UTMI output u_idpulup. 0: Drive 0 to u_idpulup 1: Drive 1 to u_idpulup	0
12	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
11	HDISCON_FLT_SEL (HOV)	RW	This bit selects a timer to filter out the HDISCON noise from UTMI+. 0: Approximated to be 135 μs 1: Approximated to be 270 μs	0
10	VBUS_FLT_SEL (HOV, POV)	RW	This bit selects a timer to filter out the VBUS_VLD noise from UTMI+. 0: Approximated to be 135 μs 1: Approximated to be 472 μs	0
9	ID_FLT_SEL	RW	This bit selects a timer to filter out the ID noise from UTMI+. 0: Approximated to be 3 ms 1: Approximated to be 4 ms	0
8	A_SRP_RESP_TYP	RW	SRP Response Type This bit determines the SRP type to which A device should respond. 0: A device responds to the Vbus pulsing 1: A device responds to the data-line pulsing	0
7	A_SRP_DET_EN	RW	SRP Detection Enable of A Device This bit controls A device to detect SRP from B	0

Vangotech

			device. 0: SRP detection is disabled. 1: SRP detection is enabled.	
6	A_SET_B_HNP_EN	RW	This bit indicates the current role is A device and the HNP function of B device has been enabled. This bit should be set and cleared by software. 0: No effect 1: HNP feature of B device has been enabled. This bit is valid when the current role is A device. This bit will be cleared to '0' only after A device issues a USB reset.	0
5	A_BUS_DROP (HOV)	RW	Bus Drop of A Device This bit determines if the A device wants to power-down Vbus. Writing this bit to '1' clears BUS_REQ of A device.	1
4	A_BUS_REQ (HOV)	RW	Bus Request of A Device This bit determines if A device should control the bus. 0: A device stops driving Vbus and bus traffic. 1: A device drives Vbus and generates bus traffic.	0
3	-	Rsvd	Reserved	0
2	B_DSCHRG_VBUS	RW	Discharge Vbus of B Device This bit is used to determine if discharging Vbus is required after the Vbus pulsing during SRP. 0: Vbus will not be discharged after the Vbus pulsing. 1: Vbus will be discharged for 50 ms after the Vbus pulsing.	0
1	B_HNP_EN	RW	This bit indicates that B device has been enabled to perform HNP. This bit can be cleared only after B device is reset by the host. 0: Disable HNP 1: Enable HNP	0
0	B_BUS_REQ	RW	Bus Request of B Device This bit determines if B device should take control of the bus. After the SRP pulsing is finished, this bit will be automatically cleared by the hardware. 0: B device stops driving Vbus and generating the bus traffic. 1: B device requests to take control of the bus.	0

Vangotech

8.1.6.15. OTG Interrupt Status Register

This register defines the interrupt status of FOTG210. The interrupt status is not masked by interrupt enable. That is, if the corresponding events happen, the corresponding status bit will be set to '1' even if the corresponding interrupt enable bit is '0'.

Table 8-15. OTG Interrupt Status Register (OTG_ISR, Address = 0x084)

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	APLGRMV	RW1C	**Mini-A Plug Remove** This bit is set to '1' once the mini-A plug is removed. Writing '1' clears this bit while writing '0' has no effect.	0
11	A_WAIT_CON (HOV)	RW1C	**A-device Wait for Connection** This bit is set to '1' once the A-device is waiting for peripheral connection. Writing '1' clears this bit while writing '0' has no effect.	0
10	OVC (HOV)	RW1C	**Over Current Detection** This bit will be set to '1' when Vbus will not reach VBUS_VLD within the expected time. Writing '1' clears this bit while writing '0' has no effect. This bit is valid only when the current role is A-device.	0
9	IDCHG	RW1C	**ID Change** This bit will be set to '1' when the current ID of FOTG210 changes either from A-device to B-device or from B-device to A-device. Writing '1' clears this bit while writing '0' has no effect.	0
8	RLCHG	RW1C	**Role Change** This bit will be set to '1' when the current role of FOTG210 changes from host to peripheral or from peripheral to host. Writing '1' clears this bit while writing '0' has no effect.	0
7	-	Rsvd	Reserved	0
6	B_SESS_END (POV)	RW1C	B_Sess_End This register will be set to '1' when B_SESS_END is high. Writing '1' clears this register while writing '0'	0

Vangotech

			has no effect.	
5	A_VBUS_ERR (HOV)	RW1C	Vbus Error of A Device This bit will be set to '1' when the OTG state machine moves to the "VBUS_ERROR" state. Writing '1' clears this bit while writing '0' has no effect.	0
4	A_SRP_DET	RW1C	A Device Detects SRP from B Device This bit will be set to '1' when A device detects SRP from B device. Writing '1' clears this bit while writing '0' has no effect.	0
[3:1]	-	Rsvd	Reserved	0
0	B_SRP_DN	RW1C	SRP Done of B Device This bit will be set to '1' after B device has completed the SRP signaling. Writing '1' clears this bit while writing '0' has no effect.	0

8.1.6.16. OTG Interrupt Enable Register

This register defines the interrupt enable of the interrupt events. This register will not mask the interrupt status. It will only mask the interrupt generation.

Table 8-16. OTG Interrupt Enable Register (OTG_IER, Address = 0x088)

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	APLGRMV_EN	RW	APLGRMV interrupt enable	0
11	A_WAIT_CON_EN (HOV)	RW	A_WAIT_CON interrupt enable	0
10	OVC_EN (HOV)	RW	OVC interrupt enable	0
9	IDCHG_EN	RW	IDCHG interrupt enable	0
8	RLCHG_EN	RW	RLCHG interrupt enable	0
7	-	Rsvd	Reserved	0
6	B_SESS_END_EN (POV)	RW	B_SESS_END interrupt enable	0
5	A_VBUS_ERR_EN (HOV)	RW	A_VBUS_ERR interrupt enable	0
4	A_SRP_DET_EN	RW	A_SRP_DET interrupt enable	0
[3:1]	-	Rsvd	Reserved	0
0	B_SRP_DN_EN	RW	B_SRP_DN interrupt enable	0

8.1.6.17. Global HC/OTG/DEV Interrupt Status Register

Table 8-17. Global HC/OTG/DEV Interrupt Status Register (GLB_ISR, Address = 0x0C0)

Bit	Name	Type	Description	Default Value

Vangotech

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
2	HC_INT	RO	HC Interrupt This bit will be set to '1' when an interrupt is issued from the host controller block.	0
1	OTG_INT	RO	OTG Interrupt This bit will be set to '1' when an interrupt is issued from the OTG controller block.	0
0	DEV_INT	RO	Device Interrupt This bit will be set to '1' when an interrupt is issued from the device controller block.	0

8.1.6.18. Global Mask of HC/OTG/DEV Interrupt Register

Table 8-18. Global Mask of HC/OTG/DEV Interrupt Register (GLB_INT, Address = 0x0C4)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0
3	INT_POLARITY	RW	This bit controls the polarity of the system interrupt signal sys_int_n, the default is active low. 0: Active low (Default) 1: Active high	0
2	MHC_INT	RW	This bit masks the interrupt bits of the HC Interrupt. 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
1	MOTG_INT	RW	This bit masks the interrupt bits of the OTG Interrupt. 0: Enable the corresponding interrupt 1: Disable he corresponding interrupt	0
0	MDEV_INT	RW	This bit masks the interrupt bits of the Device Interrupt. 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

8.1.6.19. Revision Information Register

This register indicates the revision value of FOTG210.

Table 8-19. Revision Information Register (REVISION, Address = 0x0E0)

Bit	Name	Type	Description	Default Value
[31:0]	REVISION	RO	BCD encoding of the revision value of FOTG210 0x0001_0506 indicates version 1.5.6.	0001_2800

Vangotech

8.1.6.20. Configured Information Register

This register indicates the current HW configured information.

Table 8-20. Configured Information Register (FEATURE, Address = 0x0E4)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
15	HOST_ONLY	RO	When this bit is '1', FOTG210 has been configured to be host ONLY.	0
14	DEV_ONLY	RO	When this bit is '1', FOTG210 has been configured to be device ONLY.	0
[13:9]	EP_NUM	RO	This field indicates the supported number of endpoints.	8
[8:5]	FIFO_NUM	RO	This field indicates the number of FIFOs configured.	4
[4:0]	DMABUFSIZE	RO	This field indicates the DMA buffer size.	8

8.1.6.21. Device Main Control Register

Table 8-21. Device Main Control Register (DEV_CTL, Address = 0x100)

Bit	Name	Type	Description	Default Value
[31:27]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
26	LPM_ACCEPT	RW	Accept LPM Transaction This bit is effective only when LPM_EN = 1 1: FOTG210 responds the LPM transaction with ACK. 0: FOTG210 responds the LPM transaction with NYET.	0
25	LPM_EN	RW	Enable LPM Capability 1: LPM feature is enabled. FOTG210 responds the LPM transaction according to the LPM_ACCEPT bit. 0: LPM feature is disabled. FOTG210 responds the LPM transaction with STALL.	0
24	-	Rsvd	Reserved	0
[23:20]	LPM_BESL	RO	LPM BESL This field is updated by BESL field of received LPM token.	0
[19:16]	LPM_BESL_MIN	RW	Minimum BESL The minimum BESL value supported by FOTG210. If LPM_BESL is less than LPM_BESL_MIN, FOTG210 will respond the LPM transaction with NYET even when LPM_ACCEPT is 1.	0
[15:12]	LPM_BESL_MAX	RW	Maximum BESL The maximum BESL value supported by FOTG210. If LPM_BESL is greater than LPM_BESL_MAX, FOTG210	0xF

Vangotech

			will respond the LPM transaction with NYET even when LPM_ACCEPT is 1.	
[11:10]	IDLE_DEGLITCH (HOV)	RW	Idle Line State Deglitch The tolerance for glitches at the Idle line state can be configured by this register. During the Idle state, FOTG210 monitors the line state and determines the end-of-idle depends on different settings. 0b00: Treat a non-idle cycle as end-of-idle 0b01: Treat two continuous non-idle cycles as end-of-idle 0b10: Treat three continuous non-idle cycles as end-of-idle 0b11: Treat four continuous non-idle cycles as end-of-idle Please note that these bits should be set carefully according to the electrical instability due to cable.	0b01
9	FORCE_FS	RW	Force the device to full-speed If this bit is set to 1, the high-speed negotiation will be disabled when the reset from USB host starts, which forces FOTG210 to full-speed in the device mode. If this bit is 0, the FOTG210 will remain at high-speed in the device mode. Please note that users can only modify this bit before the unplug bit is set; otherwise, it will cause unexpected results. The default value of this bit is set to '1' if USB 1.1 wrapper configuration is enabled.	0 1(100 wrapper)
8	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
7	SYSBUS_WIDTH (HOV)	RW	System Bus Width 1: System bus width is 8 bits. 0: System bus width is 32 bits. Please note that setting this bit will directly affect the access time when AP needs to obtain the SETUP 8 bytes from SETUP_CMD_RPORT (0x1D0). If this bit is set to '1', AP shall access the register 0x1D0 eight times to get the whole SETUP 8 bytes. Otherwise, the command can be obtained by reading the register twice.	0
6	HS_EN (HOV)	RO	High-Speed Status 1: Device is in the High-Speed mode. 0: Device is in the Full-Speed mode.	0

Vangotech

5	CHIP_EN (HOV)	RW	Chip Enable Writing '1' will enable the write cycle of the FIFO controller. Please do not write '0' in the normal operation mode.	1
4	SFRST (HOV)	RW	Device Software Reset Writing '1' will set the software-initiated reset to the FOTG210 device. This bit cannot be set when FOTG210 is in the suspend mode because u_clk is stopped. By setting this bit, the Chirp sequence will be terminated, the device address will be cleared, and the Frame Number Register will be cleared. Please note that the data FIFO status will not be cleared (Software reset will be self-cleared).	0
3	GOSUSP	RW	Go Suspend Writing '1' will activate the suspend mode of PHY.	0
2	GLINT_EN (HOV)	RW	Global Interrupt Enable Writing '1' will enable all interrupts. Individual interrupt can be masked by setting the corresponding bits in the interrupt mask register (Index 0x144 ~ 0x14C).	0
1	HALF_SPEED (HOV)	RW	Half Speed Enable 1: The FIFO controller asserts ACK to DMA for every two clock cycles. 0: The FIFO controller asserts ACK to DMA continuously. This bit is set to '1' when SCLK is slower than 30 MHz. SCLK should be greater than 15 MHz at least.	0
0	CAP_RMWAKUP	RW	Capability of Remote Wakeup Writing '1' indicates that FOTG210 has the capability of being wakened up from L2 by the "wakeup" signal.	0

8.1.6.22. Device Address Register

Table 8-22. Device Address Register (DEV_ADR, Address = 0x104)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	AFT_CONF	RW	After Set Configuration Writing '1' indicates that the device has successfully executed the SET_CONFIGURATION command. The FOTG210 device will not respond to any non-control transfer before this bit is set.	0 0(U)
[6:0]	DEVADR	RW	Device Address This bit records the latest USB device address for each	0 0(U)

			SET_ADDRESS.	

8.1.6.23. Device Test Register

Table 8-23. Device Test Register (DEV_TST, Address = 0x108)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	TST_MOD_TYP(HOV)	RW	Test Mode Type FOTG210 provides two configurations for protocol timers in test mode. This bit is valid only when TST_MOD is set to 1. 0: Applies normal test mode timer settings (Default) 1: Applies mutual test mode timer settings	0
6	DISGENSOF	RW	Disable the generation of SOF Please always use the SOF issued by the host instead of the self-generation SOF.	0
5	TST_MOD (HOV)	RW	Test Mode Writing '1' will turn this bit on. When this bit is set to '1', FOTG210 will enter the test mode. In the normal mode, FOTG210 uses a counter for 10 ms to detect a USB reset. The count is a large number. In the test mode, FOTG210 will use a smaller counter for the USB reset detection to save the test cycles on test machine.	0
4	TST_DISTOG (HOV)	RW	Disable Toggle Sequence Writing '1' will disable the toggle sequence.	0
3	-	Rsvd	Reserved	0
2	TST_CLREA	RW1C	Clear External Side Address Writing '1' then a '0' will clear the external side address for the loop-back test (This bit is self-clear).	0
1	TST_LPCX	RW	Loop-back Test for Control Endpoint Writing '1' indicates the loop-back test for control transfers.	0
0	TST_CLRFF (HOV)	RW	Clear FIFO Writing '1' will clear all the FIFO counters and location counters of the FIFO controller (This bit is self-clear).	0

8.1.6.24. Device SOF Frame Number Register

Table 8-24. Device SOF Frame Number Register (DEV_SFN, Address = 0x10C)

Bit	Name	Type	Description	Default

Vangotech

				Value
[31:14]	-	Rsvd	Reserved	0
[13:11]	USOFN	RO	SOF Micro Frame Number Bits[2:0] These bits record the micro frame number of the High-Speed mode.	00 (S)
[10:0]	SOFN	RO	SOF Frame Number Bits[10:0] These bits record the frame number of the High-Speed and Full-Speed modes.	00 (S)

8.1.6.25. Device SOF Mask Timer Register

Table 8-25. Device SOF Mask Timer Register (DEV_SMT, Address = 0x110)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
[15:0]	SOFMT	RW	SOF Mask Timer Time count since the last SOF in the 30 MHz clock bit. Software shall manually set the value back to this field after SFRST. The recommended values for the high-speed and full-speed devices are 0x44C and 0x2710, respectively.	0x44C 0 (S)

8.1.6.26. PHY Test Mode Selector Register

Table 8-26. PHY Test Mode Selector Register (PHY_TST, Address = 0x114)

Bit	Name	Type	Description	Default Value
[31:5]	-	Rsvd	Reserved	0
4	TST_PKT	RW	Test Mode for Packet Upon writing '1' to this bit, FOTG210 will repetitively send the packet defined in the UTMI specification to the transceiver. After the set_feature command shows the test mode and the index, Test_Packet, that has been decoded, this bit will be asserted.	0
3	TST_SE0NAK	RW	Upon writing '1', the D+/D- lines will be set to HS, the quiescent state. The device will only respond to a valid HS IN token and will always respond to the IN token with NAK.	0
2	TST_KSTA	RW	Upon writing '1', the D+/D- lines will be set to the high-speed K state.	0
1	TST_JSTA	RW	Upon writing '1', the D+/D- lines will be set to the high-speed J state.	0

Bit	Name	Type	Description	Default Value
0	UNPLUG	RW	When UNPLUG is set to '1', the device controller will set PHY in the Non-Driving mode to emulate the detachment of a device even if it is really plugged. The USB host will not detect the plugging of a device. Such an event is called "soft-detachment". After a hardware reset, the UNPLUG will be '1' and the device will therefore be soft-detached. When the USB host detects the attachment of a device, the PHY must drive D+ and D- in the manner defined in the USB specification. In order to let the PHY drive D+ and D-, AP should clear UNPLUG after the hardware reset. If the AP does not clear the UNPLUG bit, the device will always be soft-detached and the USB host will never detect the attachment of the device.	1 0(U)

8.1.6.27. Device Vendor-Specific I/O Control Register

Table 8-27. Device Vendor-Specific I/O Control Register (DEV_VCTL, Address = 0x118)

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0
5	VCTLOAD_N (HOV)	RW	Vendor-Specific Test Mode Control Load This bit controls the active low output u_vctload_n to PHY. Writing '1' to this bit will make u_vctload_n output a '1', When this bit is cleared, the u_vctload_n outputs will be set to '0'.	1
[4:0]	VCTL (HOV)	RW	Vendor-Specific Test Mode Control The programmed value is delivered to PHY via the "u_vctl" output.	0

8.1.6.28. Device CX Configuration Status Register

Table 8-28. Device CX Configuration Status Register (DEV_CXCFG, Address = 0x11C)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	VSTA (HOV)	RO	Vendor-Specific Test Mode Status	HwInit

8.1.6.29. Device CX Configuration and FIFO Empty Status Register

Table 8-29. Device CX Configuration and FIFO Empty Status Register (DEV_CXCFE, Address = 0x120)

Bit	Name	Type	Description	Default

				Value
31	-	Rsvd	Reserved	0
[30:24]	CX_FNT	RO	CX FIFO Byte Count	0
[23:12]	-	Rsvd	Reserved	0
[11:8]	F_EMP	RO	FIFO EMPTY These bits are for FW to check if FIFO is fully empty.	0b1111
[7:6]	-	Rsvd	Reserved	0
5	CX_EMP	RO	CX FIFO Empty '1' indicates that the endpoint 0 FIFO is empty.	1
4	CX_FUL	RO	CX FIFO Full '1' indicates that the endpoint 0 FIFO is full.	0
3	CX_CLR	RW	Clear CX FIFO Data Writing '1' will clear the data in the endpoint 0 FIFO. Please note that for endpoint 0, all data in FIFO will be cleared regardless of whether the previous SETUP or IN or OUT transaction has been completed or not (This bit is self-clear).	0
2	CX_STL	RW	Stall CX Writing '1' to this bit can stall endpoint 0. The stall status will be cleared with the next setup transaction. This bit will be cleared automatically when the transaction of endpoint 0 is finished. Upon detecting the bus reset, the firmware should clear this bit. Note: When setting this bit, CX_DONE must be set in the same write operation.	0 0(U)
1	TST_PKDONE	RW	Data Transfer is Done for Test Packet The firmware has completely sent the whole test patterns to the endpoint 0 FIFO for a PHY test by writing '1' to this bit. This bit is cleared by a hardware reset.	0
0	CX_DONE	RW	Data Transfer Done for CX The firmware has finished the whole packet transaction for endpoint 0 by writing '1' to this bit. This bit is cleared by a hardware reset or by a p_endcx or p_comfail internal signal.	0 0(U)

8.1.6.30. Device Idle Counter Register

Table 8-30. Device Idle Counter Register (DEV_ICR, Address = 0x124)

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
[2:0]	IDLE_CNT	RW	These bits control the timing delay from the time indicated in the GOSUSP bit of the main control	0

Vangotech

			register to the time when the device enters the suspend mode. The timing delay is denoted as tsusp_delay below. 0b000: tsusp_delay = 0 ms 0b001: tsusp_delay = 1 ms 0b010: tsusp_delay = 2 ms 0b011: tsusp_delay = 3 ms 0b100: tsusp_delay = 4 ms 0b101: tsusp_delay = 5 ms 0b110: tsusp_delay = 6 ms 0b111: tsusp_delay = 7 ms Note: USB 2.0 specifications define TSUSP to mandate the device that should enter the suspend mode no later than 10 ms after D+/D-is continuously at the idle state. The firmware programmer should be cautious in programming the value of tsusp_delay.	

8.1.6.31. Device Mask of Interrupt Group Register

Table 8-31. Device Mask of Interrupt Group Register (DEV_MIGR, Address = 0x130)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0
3	MINT_G3	RW	Mask of interrupt of source group 3 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
2	MINT_G2	RW	Mask of interrupt of source group 2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
1	MINT_G1	RW	Mask of interrupt of source group 1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
0	MINT_G0	RW	Mask of interrupt of source group 0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

8.1.6.32. Device Mask of Interrupt Source Group 0 Register

Table 8-32. Device Mask of Interrupt Source Group 0 Register (DEV_MISG0, Address = 0x134)

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0
5	MCX_COMABORT_INT	RW	Mask interrupt of the control transfer command abort	0

			0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	
4	MCX_COMFAIL_INT	RW	Mask interrupt of the host emits extra IN or OUT data interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
3	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
2	MCX_OUT_INT	RW	Mask the interrupt bits of endpoint 0 for OUT 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
1	MCX_IN_INT	RW	Mask the interrupt bits of endpoint 0 for IN 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
0	MCX_SETUP_INT	RW	Mask endpoint 0 setup data received interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

8.1.6.33. Device Mask of Interrupt Source Group 1 Register

Table 8-33. Device Mask of Interrupt Source Group 1 Register (DEV_MISG1, Address = 0x138)

Bit	Name	Type	Description	Default Value
[31:20]	-	Rsvd	Reserved	0
19	MF3_IN_INT	RW	Mask the IN interrupt bits of FIFO 3 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
18	MF2_IN_INT	RW	Mask the IN interrupt bits of FIFO 2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
17	MF1_IN_INT	RW	Mask the IN interrupt bits of FIFO 1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
16	MF0_IN_INT	RW	Mask the IN interrupt bits of FIFO 0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
[15:8]	-	Rsvd	Reserved	0
7	MF3_SPK_INT	RW	Mask the Short Packet interrupt of FIFO 3 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
6	MF3_OUT_INT	RW	Mask the OUT interrupt of FIFO 3 0: Enable the corresponding interrupt	1

Vangotech

			1: Disable the corresponding interrupt	
5	MF2_SPK_INT	RW	Mask the Short Packet interrupt of FIFO 2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
4	MF2_OUT_INT	RW	Mask the OUT interrupt of FIFO 2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
3	MF1_SPK_INT	RW	Mask the Short Packet interrupt of FIFO 1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
2	MF1_OUT_INT	RW	Mask the OUT interrupt of FIFO 1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
1	MF0_SPK_INT	RW	Mask the Short Packet interrupt of FIFO 0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1
0	MF0_OUT_INT	RW	Mask the OUT interrupt of FIFO 0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	1

8.1.6.34. Device Mask of Interrupt Source Group 2 Register

Table 8-34. Device Mask of Interrupt Source Group 2 Register (DEV_MISG2, Address = 0x13C)

Bit	Name	Type	Description	Default Value
[31:12]	-	Rsvd	Reserved	0
11	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
10	MDev_Wakeup_byVBUS	RW	Mask Dev_Wakeup_byVBUS interrupt Mask active Dev_Wakeup_byVBUS interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
9	MDev_Idle (HOV)	RW	Mask Dev_Idle interrupt Mask active the Dev_Idle interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
8	MDMA_ERROR (HOV)	RW	Mask DMA Error Interrupt Mask active the DMA Error interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

Vangotech

7	MDMA_CMPLT (HOV)	RW	Mask DMA Completion Interrupt Mask active the DMA Completion interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
6	MRX0BYTE_INT	RW	Mask Received Zero-length Data Packet Interrupt Mask active Received Zero-length Data Packet interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
5	MTX0BYTE_INT	RW	Mask Transferred Zero-length Data Packet Interrupt Mask active Transferred Zero-length Data Packet interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
4	MSEQ_ABORT_INT	RW	Mask ISO Sequential Abort Interrupt Mask active ISO Sequential Abort interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
3	MSEQ_ERR_INT	RW	Mask ISO Sequential Error Interrupt Mask active Received ISO Sequential Error interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
2	MRESM_INT	RW	Mask Resume Interrupt Mask active Resume State Change Interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
1	MSUSP_INT	RW	Mask Suspend Interrupt Mask active Suspend State Change Interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
0	MUSBRST_INT	RW	Mask USB Reset Interrupt Mask the Bus Reset Interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

Vangotech

8.1.6.35. Device Interrupt Group Register

Table 8-35. Device Interrupt Group Register (DEV_IGR, Address = 0x140)

Bit	Name	Type	Description	Default Value
[31:4]	-	Rsvd	Reserved	0
3	INT_G3	RO	This bit indicates some interrupts occurred in Group 3.	0
2	INT_G2 (HOV)	RO	This bit indicates some interrupts occurred in Group 2.	0
1	INT_G1	RO	This bit indicates some interrupts occurred in Group 1.	0
0	INT_G0	RO	This bit indicates some interrupts occurred in Group 0.	0

8.1.6.36. Device Interrupt Source Group 0 Register

Table 8-36. Device Interrupt Source Group 0 Register (DEV_ISG0, Address = 0x144)

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0
5	CX_COMABT_INT	RW1C	This bit indicates that a command abort event has happened, which a new SETUP arrives before the last control transfer is done. For the interrupts recorded in this source register, the command abort interrupt has the highest priority. For a command abort interrupt, the AP should only clear the CX_COMABT_INT bit. All other operations will be unnecessary and should be avoided. In general, the command abort interrupt will be accompanied by CX_SETUP_INT. AP should first serve the command abort interrupt to clear CX_COMABT_INT because CXF FIFO is frozen for AP to access when CX_COMABT_INT remains as '1'. In order to get 8-byte for SETUP for the command abort, AP should clear CX_COMABT_INT first.	0 0(S)
4	CX_COMFAIL_INT	RO	This bit indicates that the control transfer has been abnormally terminated. This bit will be asserted when the device receives extra IN/OUT token at the data stage of the control transfer. Once this bit is asserted, it will be kept at '1' before AP sets the CX_STL bit of the CX_Config_Status register. After setting the CX_STL bit of the CX_Config_Status	0

Vangotech

			register, the AP should set the CX_DONE bit of the CX_Config_Status register.	
3	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
2	CX_OUT_INT	RO	This bit indicates that the control transfer contains valid data for the control-write transfers. This bit will remain asserted until the firmware starts to read data from control transfer FIFO (CXF) of device.	0
1	CX_IN_INT	RO	This bit indicates that the firmware should write data for the control-read transfer to control transfer FIFO. For control transfer reads with length less than or equal to 64 bytes, this bit will never be asserted. The firmware will decode the 8-byte data sent at the SETUP stage of control transfer. The firmware should write the first payload of data into the control transfer FIFO if the 8-byte indicates the control-read transfer without asserting this bit. This bit will be asserted only when the length of the control-read transfer is greater than 64 bytes and the USB host has successfully received the data of previous packet. For example, for a 65-byte control-read transfer, the firmware should automatically write the first 64 bytes after it decodes eight byte of the SETUP data. The firmware will be interrupted to write the 65th byte when the USB host ACKs to the first 64 bytes. This bit will remain asserted until FW starts to write data into the control transfer FIFO of device.	0
0	CX_SETUP_INT	RO	The occasion for asserting this interrupt is the token of data or status stage in control transfer is received by controller. It indicates SW shall convey the SETUP 8 bytes out from CXF. This bit will remain asserted until the firmware starts to read data from the control transfer FIFO of device.	0

8.1.6.37. Device Interrupt Source Group 1 Register

Table 8-37. Device Interrupt Source Group 1 Register (DEV_ISG1, Address = 0x148)

Bit	Name	Type	Description	Default Value
[31:20]	-	Rsvd	Reserved	0

19	F3_IN_INT	RO	This bit becomes '1' to indicate that FIFO 3 is ready to be written. This bit is cleared under the following two conditions: ● A maximum-size-packet is received in FIFO 3. ● End of the DMA transaction	0
18	F2_IN_INT	RO	This bit becomes '1' to indicate that FIFO 2 is ready to be written. This bit is cleared under the following two conditions: ● A maximum-size-packet is received in FIFO 2. ● End of the DMA transaction	0
17	F1_IN_INT	RO	This bit becomes '1' to indicate that FIFO 1 is ready to be written. This bit is cleared under the following two conditions: ● A maximum-size-packet is received in FIFO 1. ● End of the DMA transaction	0
16	F0_IN_INT	RO	This bit becomes '1' to indicate that FIFO 0 is ready to be written. This bit is cleared under the following two conditions: ● A maximum-size-packet is received in FIFO 0. ● End of the DMA transaction	0
[15:8]	-	Rsvd	Reserved	0
7	F3_SPK_INT	RO	This bit becomes '1' when the short packet data are received in FIFO 3. This bit is cleared once the DMA master reads FIFO 3.	0
6	F3_OUT_INT	RO	This bit becomes '1' when FIFO 3 is ready to be read. This bit is cleared when all data in FIFO 3 are read out.	0
5	F2_SPK_INT	RO	This bit becomes '1' when the short packet data are received in FIFO 2. This bit is cleared once the DMA master reads FIFO 2.	0
4	F2_OUT_INT	RO	This bit becomes '1' when FIFO 2 is ready to be read. This bit is cleared when all data in FIFO 2 are read out.	0
3	F1_SPK_INT	RO	This bit becomes '1' when the short packet data are received in FIFO 1. This bit is cleared once the DMA master reads FIFO 1.	0
2	F1_OUT_INT	RO	This bit becomes '1' when FIFO 1 is ready to be read. This bit is cleared when all data in FIFO 1 are read out.	0
1	F0_SPK_INT	RO	This bit becomes '1' when short packet data are received in FIFO 0. This bit is cleared once the DMA master reads FIFO 0.	0
0	F0_OUT_INT	RO	This bit becomes '1' when FIFO 0 is ready to be read. This bit is cleared when all data in FIFO 0 are read out.	0

Vangotech

8.1.6.38. Device Interrupt Source Group 2 Register

Table 8-38. Device Interrupt Source Group 2 Register (DEV_ISG2, Address = 0x14C)

Bit	Name	Type	Description	Default Value
[31:11]	-	Rsvd	Reserved	0
10	Dev_Wakeup_byVBUS	RO	Device Wakeup by Vbus Interrupt When the device is at the idle state and the Vbus signal is high, this bit will be set to 1.	0
9	Dev_Idle (HOV)	RO	Device Idle Interrupt This bit can be activated under the following two conditions: ● Both A-device and B-device state machines are at the idle state. ● The SessEnd signal is high.	1
8	DMA_ERROR (HOV)	RW1C	DMA Error Interrupt The DMA operation cannot be finished normally and an error signal has been received. When CPU initiates DMA to fill up or read out FIFO of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit will not be set if Virtual Multi-DMA Channel is enabled. Software shall refer to VDMA_ERROR_Fn in DEV_ISG3 (0x328) as error response reception instead. This bit is not affected by a USB bus reset.	0 0 (S)
7	DMA_CMPLT (HOV)	RW1C	DMA Completion Interrupt The DMA operation has finished normally. When CPU initiates the DMA to fill up or read out FIFO of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware. This bit will not be set if Virtual Multi-DMA Channel is enabled. Software shall refer to VDMA_CMPLT_Fn in DEV_ISG3 (0x328) as VDMA mission completion instead. This bit is not affected by a USB bus reset.	0 0 (S)
6	RX0BYTE_INT	RW1C	Received Zero-length Data Packet Interrupt The device receives a zero-length data packet from the USB host. This bit will be set when the device receives a	0 0 (S)

Vangotech

			zero-length data packet from the USB host. The firmware may further check the register 0x150 to determine which endpoint should receive the zero-length data packet from the USB host. When the interrupt occurs, the device will NAK the next OUT transaction to the same endpoint until the corresponding bit (In 0x150) is cleared by the firmware. Firmware should clear the register 0x150, then clear this bit. This bit is not affected by a USB bus reset.	
5	TX0BYTE_INT	RW1C	Transferred Zero-length Data Packet Interrupt The device returns a zero-length data packet to the USB host. This bit will be set when the device returns a zero-length data packet to the USB host. The firmware should further check the register 0x154 to determine which endpoint return a zero-length data packet to the USB host. After AP served the interrupt request, this bit must be cleared by the firmware. Firmware should clear the register 0x154, then clear this bit. This bit is not affected by a USB bus reset.	0 0 (S)
4	ISO_SEQ_ABORT_INT	RW1C	ISO Sequential Abort Interrupt High bandwidth isochronous sequential abort When the device detects an incomplete DATA PID sequence during a micro-frame, this bit will be set. For example, the condition that a device detects an MDATA followed by an SOF will be regarded as "sequential abort". The firmware should further check the register 0x154 to determine which endpoint should receive the isochronous sequential abort. After AP serviced the interrupt request, this bit must be cleared by the firmware. This bit is not affected by a USB bus reset.	0 0 (S)
3	ISO_SEQ_ERR_INT	RW1C	Isochronous Sequential Error Interrupt High bandwidth isochronous sequential error When a device detects a DATA PID sequence error of an isochronous transaction in the high bandwidth, this bit will be set. Any sequence that is out of order will be taken as a "sequence error". The firmware should further check the register 0x154 to determine which	0 0 (S)

Vangotech

			endpoint should receive the isochronous sequential error. After AP served the interrupt request, this bit must be cleared by the firmware. This bit is not affected by a USB bus reset.	
2	RESM_INT	RW1C	Resume Interrupt Resume-state-change interrupt bit When the device detects the resume event from the host, this bit will be set. After AP served the interrupt request, this bit must be cleared by the firmware. When a USB bus reset occurs, this bit will also be cleared.	0 0 (S)
1	SUSP_INT	RW1C	Suspend Interrupt Suspend-state-change interrupt bit When the USB bus remains in an idle state for over 3 ms, this bit will be set. This bit must be cleared before the firmware sets the "GOSUSP" bit of the 0x100 register. This bit will also be cleared when the USB bus reset or resume occurs.	0 0 (S) 0 (U)
0	USBRST_INT	RW1C	USB Reset Interrupt Bus reset interrupt bit. When the device detects the USB bus reset from the host, this bit will be set. When AP serves the interrupt request, this bit must be cleared by the firmware.	0 0 (S) 1 (U)

8.1.6.39. Device Receive Zero-Length Data Packet Register

All bits are status bits that respond to the zero-length data packet received by an endpoint.

Table 8-39. Device Receive Zero-Length Data Packet Register (DEV_RXZ, Address = 0x150)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	RX0BYTE_EP8	RW1C	Endpoint 8 receives a zero-length data packet.	0
6	RX0BYTE_EP7	RW1C	Endpoint 7 receives a zero-length data packet.	0
5	RX0BYTE_EP6	RW1C	Endpoint 6 receives a zero-length data packet.	0
4	RX0BYTE_EP5	RW1C	Endpoint 5 receives a zero-length data packet.	0
3	RX0BYTE_EP4	RW1C	Endpoint 4 receives a zero-length data packet.	0
2	RX0BYTE_EP3	RW1C	Endpoint 3 receives a zero-length data packet.	0
1	RX0BYTE_EP2	RW1C	Endpoint 2 receives a zero-length data packet.	0
0	RX0BYTE_EP1	RW1C	Endpoint 1 receives a zero-length data packet.	0

8.1.6.40. Device Transfer Zero-length Data Packet Register

All bits are status bits that respond to the zero-length data packet sent by an endpoint.

Vangotech

Table 8-40. Device Transfer Zero-length Data Packet Register (DEV_TXZ, Address = 0x154)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
7	TX0BYTE_EP8	RW1C	Endpoint 8 transfers zero-length data packet. Only write 1'b1 is allowed.	0
6	TX0BYTE_EP7	RW1C	Endpoint 7 transfers zero-length data packet. Only write 1'b1 is allowed.	0
5	TX0BYTE_EP6	RW1C	Endpoint 6 transfers zero-length data packet. Only write 1'b1 is allowed.	0
4	TX0BYTE_EP5	RW1C	Endpoint 5 transfers zero-length data packet. Only write 1'b1 is allowed.	0
3	TX0BYTE_EP4	RW1C	Endpoint 4 transfers zero-length data packet. Only write 1'b1 is allowed.	0
2	TX0BYTE_EP3	RW1C	Endpoint 3 transfers zero-length data packet. Only write 1'b1 is allowed.	0
1	TX0BYTE_EP2	RW1C	Endpoint 2 transfers zero-length data packet. Only write 1'b1 is allowed.	0
0	TX0BYTE_EP1	RW1C	Endpoint 1 transfers zero-length data packet. Only write 1'b1 is allowed.	0

8.1.6.41. Device Isochronous Sequential Error/Abort Register

Table 8-41. Device Isochronous Sequential Error/Abort Register (DEV_ISE, Address = 0x158)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
23	ISO_SEQ_ERR_EP8	RW	Endpoint 8 encounters an isochronous sequential error.	0
22	ISO_SEQ_ERR_EP7	RW	Endpoint 7 encounters an isochronous sequential error.	0
21	ISO_SEQ_ERR_EP6	RW	Endpoint 6 encounters an isochronous sequential error.	0
20	ISO_SEQ_ERR_EP5	RW	Endpoint 5 encounters an isochronous sequential error.	0
19	ISO_SEQ_ERR_EP4	RW	Endpoint 4 encounters an isochronous sequential error.	0
18	ISO_SEQ_ERR_EP3	RW	Endpoint 3 encounters an isochronous sequential error.	0
17	ISO_SEQ_ERR_EP2	RW	Endpoint 2 encounters an isochronous sequential error.	0
16	ISO_SEQ_	RW	Endpoint 1 encounters an isochronous sequential error.	0

Vangotech

	ERR_EP1			
[15:8]	-	Rsvd	Reserved	0
7	ISO_ABT_ ERR_EP8	RW	Endpoint 8 encounters an isochronous sequential abort.	0
6	ISO_ABT_ ERR_EP7	RW	Endpoint 7 encounters an isochronous sequential abort.	0
5	ISO_ABT_ ERR_EP6	RW	Endpoint 6 encounters an isochronous sequential abort.	0
4	ISO_ABT_ ERR_EP5	RW	Endpoint 5 encounters an isochronous sequential abort.	0
3	ISO_ABT_ ERR_EP4	RW	Endpoint 4 encounters an isochronous sequential abort.	0
2	ISO_ABT_ ERR_EP3	RW	Endpoint 3 encounters an isochronous sequential abort.	0
1	ISO_ABT_ ERR_EP2	RW	Endpoint 2 encounters an isochronous sequential abort.	0
0	ISO_ABT_ ERR_EP1	RW	Endpoint 1 encounters an isochronous sequential abort.	0

8.1.6.42. Device IN Endpoint n MaxPacketSize Register

Table 8-42. Device IN Endpoint n MaxPacketSize Register (One per Endpoint, n = 1 ~ 8) (DEV_INMPS, Address = 0x160 + 4(n -1))

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
15	TX0BYTE_I EPn	RW	Transfer a Zero-length Data Packet from Endpoint n to USB Host This bit should be set after the last packet of a transaction is sent to FIFO. After the endpoint data in FIFO are transferred, the device will return a zero-length data packet to the next IN transaction to the same endpoint. The AP should not send the next packet to the same endpoint until TX0BYTE_INT for the endpoint occurs. This bit will be automatically cleared by the hardware when TX0BYTE_INT occurs.	0
[14:13]	TX_NUM_H BW_IEPn	RW	Transaction Number of High Bandwidth Endpoint n TX_NUM_HBW_IEPn[14:13] (Only valid for the isochronous transfer) 0b00, 0b01: Indicate that the endpoint n is in non-high-bandwidth	0

			0b10: Two transactions for each micro-frame 0b11: Three transactions for each micro-frame	
12	RSTG_IEPn	RW	Reset Toggle Sequence for IN Endpoint n The firmware resets the toggle bit of indexed endpoint n by writing '1' to this bit. This bit should also be cleared by the firmware.	0
11	STL_IEPn	RW	Stall IN Endpoint n The indexed endpoint n can be stalled by writing '1' to this bit. The stalled status of the indexed endpoint n can be cleared by writing '0' to this bit. Before setting this bit, users should check the FIFO empty register (0x120) and make sure that the related FIFO is empty.	0 0(U)
[10:0]	MAXPS_IEPn	RW	Max Packet Size of IN Endpoint n The maximum packet size of endpoint n capable of sending or receiving data that is smaller than or equal to this size. Note: This size must not exceed the FIFO size.	0x200

8.1.6.43. Device OUT Endpoint n MaxPacketSize Register

Table 8-43. Device OUT Endpoint n MaxPacketSize Register (One per Endpoint, n = 1 ~ 8) (DEV_OUTMPS, Address = 0x180 + 4(n -1))

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	RSTG_OEPn	RW	Reset Toggle Sequence for OUT Endpoint n The firmware resets the toggle bit of the indexed endpoint n by writing '1' to this bit. This bit should also be cleared by the firmware.	0
11	STL_OEPn	RW	Stall OUT Endpoint n Writing '1' to STL_OEP will stall endpoint n. The stalled status of the indexed endpoint n can be cleared by writing '0' to this bit.	0 0(U)
[10:0]	MAXPS_OEPn	RW	Max Packet Size of OUT Endpoint n The maximum packet size of endpoint n capable of sending or receiving data that is smaller than or equal to this size.	0x200

8.1.6.44. Device Endpoint 1 ~ 4 Map Register

Please note that FNO_OEPn and FNO_IEPn shall be correctly initialized by the software based on the total FIFO block numbers. For example, the valid range shall be between 0 and 3 if there are a total of four FIFO blocks included in the device controller.

Table 8-44. Device Endpoint 1 ~ 4 Map Register (DEV_EPMAP0, Address = 0x1A0)

Bit	Name	Type	Description	Default Value
[31:30]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[29:28]	FNO_OEP4	RW	FIFO Number of OUT Endpoint 4 This register maps logical OUT endpoint 4 to a physical FIFO number.	0b11
[27:26]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[25:24]	FNO_IEP4	RW	FIFO Number of IN Endpoint 4 This register maps logical IN endpoint 4 to a physical FIFO number.	0b11
[23:22]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[21:20]	FNO_OEP3	RW	FIFO Number of OUT Endpoint 3 This register maps logical OUT endpoint 3 to a physical FIFO number.	0b11
[19:18]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[17:16]	FNO_IEP3	RW	FIFO Number of IN Endpoint 3 This register maps logical IN endpoint 3 to a physical FIFO number.	0b11
[15:14]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[13:12]	FNO_OEP2	RW	FIFO Number of OUT Endpoint 2 This register maps logical OUT endpoint 2 to a physical FIFO number.	0b11
[11:10]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[9:8]	FNO_IEP2	RW	FIFO Number of IN Endpoint 2 This register maps logical IN endpoint 2 to a physical FIFO number.	0b11
[7:6]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for	0

Vangotech

			writing to these bits.	
[5:4]	FNO_OEP1	RW	FIFO Number of OUT Endpoint 1 This register maps logical OUT endpoint 1 to a physical FIFO number.	0b11
[3:2]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[1:0]	FNO_IEP1	RW	FIFO Number of IN Endpoint 1 This register maps logical IN endpoint 1 to a physical FIFO number.	0b11

8.1.6.45. Device Endpoint 5 ~ 8 Map Register

Please note that FNO_OEPn and FNO_IEPn shall be correctly initialized by the software based on the total FIFO block numbers. For example, the valid range shall be 0 to 3 if there are a total of four FIFO blocks included in the device controller.

Table 8-45. Device Endpoint 5 ~ 8 Map Register (DEV_EPMAP1, Address = 0x1A4)

Bit	Name	Type	Description	Default Value
[31:30]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[29:28]	FNO_OEP8	RW	FIFO Number of OUT Endpoint 8 This register maps logical OUT endpoint 8 to a physical FIFO number.	0b11
[27:26]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[25:24]	FNO_IEP8	RW	FIFO Number of IN Endpoint 8 This register maps logical IN endpoint 8 to a physical FIFO number.	0b11
[23:22]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[21:20]	FNO_OEP7	RW	FIFO Number of OUT Endpoint 7 This register maps logical OUT endpoint 7 to a physical FIFO number.	0b11
[19:18]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[17:16]	FNO_IEP7	RW	FIFO Number of IN Endpoint 7 This register maps logical IN endpoint 7 to a physical FIFO	0b11

Vangotech

Bit	Name	Type	Description	Default Value
			number.	
[15:14]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[13:12]	FNO_OEP6	RW	FIFO Number of OUT Endpoint 6 This register maps logical OUT endpoint 6 to a physical FIFO number.	0b11
[11:10]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[9:8]	FNO_IEP6	RW	FIFO Number of IN Endpoint 6 This register maps logical IN endpoint 6 to a physical FIFO number.	0b11
[7:6]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[5:4]	FNO_OEP5	RW	FIFO Number of OUT Endpoint 5 This register maps logical OUT endpoint 5 to a physical FIFO number.	0b11
[3:2]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits.	0
[1:0]	FNO_IEP5	RW	FIFO Number of IN Endpoint 5 This register maps logical IN endpoint 5 to a physical FIFO number.	0b11

8.1.6.46. Device FIFO Map Register

Table 8-46. Device FIFO Map Register (DEV_FMAP, Address = 0x1A8)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:28]	Dir_FIFO3	RW	FIFO 3 Direction Data transfer direction 0b00: Out 0b01: In 0b10: Bidirectional 0b11: Not allowed	0
[27:24]	EPNO_FIFO3	RW	Endpoint Number of FIFO 3 This register maps physical FIFO 3 to an endpoint number.	0b1111

Vangotech

			That is to say, the packet belongs the endpoint in this field will be restored in FIFO 3.	
[23:22]	-	Rsvd	Reserved	0
[21:20]	Dir_FIFO2	RW	FIFO 2 Direction Data transfer direction 0b00: Out 0b01: In 0b10: Bidirectional 0b11: Not allowed	0
[19:16]	EPNO_FIFO2	RW	Endpoint Number of FIFO 2 This register maps physical FIFO 2 to an endpoint number. That is to say, the packet belongs to the endpoint in this field will be restored in FIFO 2.	0b1111
[15:14]	-	Rsvd	Reserved	0
[13:12]	Dir_FIFO1	RW	FIFO 1 Direction Data transfer direction 0b00: Out 0b01: In 0b10: Bidirectional 0b11: Not allowed	0
[11:8]	EPNO_FIFO1	RW	Endpoint Number of FIFO 1 This register maps physical FIFO 1 to an endpoint number. That is to say, the packet belongs to the endpoint in this field will be restored in FIFO 1.	0b1111
[7:6]	-	Rsvd	Reserved	0
[5:4]	Dir_FIFO0	RW	FIFO 0 Direction Data transfer direction 0b00: Out 0b01: In 0b10: Bidirectional 0b11: Not allowed	0
[3:0]	EPNO_FIFO0	RW	Endpoint Number of FIFO 0 This register maps physical FIFO 0 to an endpoint number. That is to say, the packet belongs to the endpoint in this field will be restored in FIFO 0.	0b1111

8.1.6.47. Device FIFO Configuration Register

Table 8-47. Device FIFO Configuration Register (DEV_FCFG, Address = 0x1AC)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0

Vangotech

29	EN_F3	RW	Enable FIFO 3 Writing '1' to this bit indicates that FIFO is enabled.	0
28	BLKSZ_F3	RW	Block Size of FIFO 3 BLKSIZE_F3 = 0: The maximum packet size of transferring packets is smaller than or equal to 512 bytes. BLKSIZE_F3 = 1: The maximum packet size of transferring packets is smaller than or equal to 1024 bytes and greater than 512 bytes. Please note that the byte counts above are exemplified by the controller with 2-kB SRAM FIFO configuration. It is unnecessary to turn this bit on if 4-kB SRAM is used instead.	0
[27:26]	BLKNO_F3	RW	Block Number of FIFO 3 BLKNUM_F3 = 0b00: Single block BLKNUM_F3 = 0b01: Double blocks BLKNUM_F3 = 0b10: Triple blocks BLKNUM_F3 = 0b11: Reserved	0
[25:24]	BLK_TYP_F3	RW	Transfer Type of FIFO 3 This register indicates the transfer type used for the FIFOn transfer. TYP_F3 = 0b00: Reserved TYP_F3 = 0b01: Isochronous type TYP_F3 = 0b10: Bulk type TYP_F3 = 0b11: Interrupt type	0
[23:22]	-	Rsvd	Reserved	0
21	EN_F2	RW	Enable FIFO 2 Writing '1' to this bit indicates that FIFO is enabled.	0
20	BLKSZ_F2	RW	Block Size of FIFO 2 BLKSIZE_F2 = 0: The maximum packet size of transferring packets is smaller than or equal to 512 bytes. BLKSIZE_F2 = 1: The maximum packet size of transferring packets is smaller than or equal to 1024 bytes and greater than 512 bytes. Please note that the byte counts above are exemplified by the controller with 2-kB SRAM FIFO configuration. It is unnecessary to turn this bit on if 4-kB SRAM is used instead.	0
[19:18]	BLKNO_F2	RW	Block Number of FIFO 2 BLKNUM_F2 = 0b00: Single block BLKNUM_F2 = 0b01: Double blocks BLKNUM_F2 = 0b10: Triple blocks BLKNUM_F2 = 0b11: Reserved	0

Vangotech

[17:16]	BLK_TYP_F2	RW	Transfer Type of FIFO 2 This register indicates the transfer type used for the FIFO 2 transfer. TYP_F2 = 0b00: Reserved TYP_F2 = 0b01: Isochronous type TYP_F2 = 0b10: Bulk type TYP_F2 = 0b11: Interrupt type	0
[15:14]	-	Rsvd	Reserved	0
13	EN_F1	RW	Enable FIFO 1 Writing '1' to this bit indicates that FIFO is enabled.	0
12	BLKSZ_F1	RW	Block Size of FIFO 1 BLKSIZE_F1 = 0: The maximum packet size of transferring packets is smaller than or equal to 512 bytes. BLKSIZE_F1 = 1: The maximum packet size of transferring packets is smaller than or equal to 1024 bytes and greater than 512 bytes. Please note that the byte counts above are exemplified by the controller with 2-kB SRAM FIFO configuration. It is unnecessary to turn this bit on if 4kB SRAM is used instead.	0
[11:10]	BLKNO_F1	RW	Block Number of FIFO 1 BLKNUM_F1 = 0b00: Single block BLKNUM_F1 = 0b01: Double blocks BLKNUM_F1 = 0b10: Triple blocks BLKNUM_F1 = 0b11: Reserved	0
[9:8]	BLK_TYP_F1	RW	Transfer Type of FIFO 1 This register indicates the transfer type used for the FIFO 1 transfer. TYP_F1 = 0b00: Reserved TYP_F1 = 0b01: Isochronous type TYP_F1 = 0b10: Bulk type TYP_F1 = 0b11: Interrupt type	0
[7:6]	-	Rsvd	Reserved	0
5	EN_F0	RW	Enable FIFO 0 Writing '1' to this bit indicates that FIFO is enabled.	0
4	BLKSZ_F0	RW	Block Size of FIFO 0 BLKSIZE_F0 = 0: The maximum packet size of transferring packets is smaller than or equal to 512 bytes. BLKSIZE_F0 = 1: The maximum packet size of transferring packets is smaller than or equal to 1024 bytes and greater than 512 bytes. Please note that the byte counts above are exemplified by	0

Vangotech

			the controller with 2-kB SRAM FIFO configuration. It is unnecessary to turn this bit on if 4-kB SRAM is used instead.	
[3:2]	BLKNO_F0	RW	Block Number of FIFO 0 BLKNUM_F0 = 0b00: Single block BLKNUM_F0 = 0b01: Double blocks BLKNUM_F0 = 0b10: Triple blocks BLKNUM_F0 = 0b11: Reserved	0
[1:0]	BLK_TYP_F0	RW	Transfer Type of FIFO 0 This register indicates the transfer type used for the FIFO 0 transfers. TYP_F0 = 0b00: Reserved TYP_F0 = 0b01: Isochronous type TYP_F0 = 0b10: Bulk type TYP_F0 = 0b11: Interrupt type	0

8.1.6.48. Device FIFO n Instruction and Byte Count Register

Table 8-48. Device FIFO n Instruction and Byte Count Register (One per FIFO, n = 0 ~ 3) (DEV_FIBC, Address = 0x1B0 + 4n)

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	FFRST (HOV)	RW	FIFO n Reset FIFO can be reset by the firmware by setting this bit. This bit will be automatically cleared.	0
11	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
[10:0]	BC_Fn	RO	OUT FIFO n Byte Count BC_Fn[10:0] indicates the byte number of data stored in FIFO n for OUT EP.	0

8.1.6.49. Device DMA Target FIFO Number Register

Note: Only one of the DMA_TFN bits can be set at the same time.

Table 8-49. Device DMA Target FIFO Number Register (DMA_TFN, Address = 0x1C0)

Bit	Name	Type	Description	Default Value
[31:5]	-	Rsvd	Reserved	0
4	ACC_CXF (HOV)	RW	Accessing Control Transfer FIFO When this bit is set to '1', the DMA target FIFO will be the control transfer FIFO.	0

Bit	Name	Type	Description	Default Value
3	ACC_F3 (HOV)	RW	Accessing FIFO 3 When this bit is set to '1', the DMA target FIFO will be FIFO 3.	0
2	ACC_F2 (HOV)	RW	Accessing FIFO 2 When this bit is set to '1', the DMA target FIFO will be FIFO 2.	0
1	ACC_F1 (HOV)	RW	Accessing FIFO 1 When this bit is set to '1', the DMA target FIFO will be FIFO 1.	0
0	ACC_F0 (HOV)	RW	Accessing FIFO 0 When this bit is set to '1', the DMA target FIFO will be FIFO 0.	0

8.1.6.50. Device DMA Controller Parameter Setting 0 Register

Note: This register is valid only when "FOTG210_IF_AXI" is enabled by using hardware configuration. Otherwise, this register will be reserved as 0 and read only.

Table 8-50. Device DMA Controller Parameter Setting 0 Register (DMA_CPS0, Address = 0x1C4)

Bit	Name	Type	Description	Default Value
[31:20]	-	Rsvd	Reserved	0
19	DST_WD (HOV)	RW	Destination Data Width Select This bit controls the AXI beat size when fixed address mode is enabled. 0: DMA accesses in 32-bit beats 1: DMA accesses in 64-bit beats This field is only valid when "FOTG210_AXI_64BIT" is enabled by using hardware configuration.	1
18	BUF_LD_EN (HOV)	RW	Load Enable for Bufferable Bit of AWCAHCE and ARCAHCE If this bit is set to 1, the Bufferable bit of AXI signals AWCACHE and ARCACHE is set according to the corresponding bit in fields AWCACHE and ARCACHE. Otherwise, the Bufferable bit is hardware controlled.	0
[17:16]	ARLOCK (HOV)	RW	User Definition of AXI ARLOCK This field controls the AXI ARLOCK signal.	0
[15:13]	ARPORT (HOV)	RW	User Definition of AXI ARPROT This field controls the AXI ARPROT signal.	0
[12:9]	ARCACHE (HOV)	RW	User Definition of AXI ARCACHE This field controls the AXI ARCACHE signal.	0
[8:7]	AWLOCK	RW	User Definition of AXI AWLOCK	0

Vangotech

	(HOV)		This field controls the AXI AWLOCK signal.	
[6:4]	AWPROT (HOV)	RW	User Definition of AXI AWPROT This field controls the AXI AWPROT signal.	0
[3:0]	AWCACHE (HOV)	RW	User Definition of AXI AWCACHE This field controls the AXI AWCACHE signal.	0

8.1.6.51. Device DMA Controller Parameter Setting 1 Register

Table 8-51. Device DMA Controller Parameter Setting 1 Register (DMA_CPS1, Address = 0x1C8)

Bit	Name	Type	Description	Default Value
31	DevPhy_Suspend (HOV)	RW	Device Transceiver Suspend Mode Active high Place the transceiver in the suspend mode that draws the minimal power from the power supplies. This bit is only used in the device mode. This bit is in the system clock domain instead of the UCLK clock domain. Please note that it has no effect and cannot be written to 1 when the device is powered. Software is allowed to enable this bit only when BVALID is low.	0
30	L1_WAKEUP	RW	Wakeup from L1 While this bit is set to 1'b1, FOTG210 starts to perform device resume to exit L1. After L1 exit is done, this bit would be cleared as 1'b0	0
29	UNDEF_LEN_BURST (HOV)	RW	AHB Undefined Length Burst This bit is used for configuring the burst behavior of AHB Initiator Interface. This field exists only when "FOTG210_IF_AHB" is enabled by using hardware configuration. Otherwise, this register will be reserved as 0 and read only. 0: AHB Master initiates at most 4-beat burst transfers 1: AHB Master initiates undefined length burst transfers	1
[28:25]	R_HPROT (HOV)	RW	User Definition of AHB HPROT This signal is reserved for users to control the AHB HPROT signal. This field exists only when "FOTG210_IF_AHB" is enabled by using hardware configuration. Otherwise, this register will be reserved as 0 and	0b0011

			read only.	
[24:8]	DMA_LEN (HOV)	RW	DMA Length These bytes indicate the total bytes moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'. During handling the control transfer, the maximum length must not exceed 64. Please note that FOTG210 AXI master would issue the outstanding commands before the read data channel is ready or the write data channel is valid. For IN transfer, If DMA_LEN is larger than FIFO size, the read data channel may not be ready unless FIFO is not full. For OUT transfer, if DMA_LEN is larger than FIFO size, the write data channel may not be valid unless FIFO is not empty.	0
[7:5]	-	Rsvd	Reserved	0
4	CLRFIFO_DMAABORT (HOV)	RW	Clear FIFO when DMA_ABORT This bit will be set to '1' when combined with the DMA_ABORT bit to clear the content in FIFO after completing the DMA abort. If users need to abort DMA and clear the contents in FIFO, this bit must be set to '1' together with DMA_ABORT. The contents in FIFO will not be cleared if this bit is cleared to '0'.	0
3	DMA_ABORT (HOV)	RW	DMA Abort This bit forces the DMA abort during the DMA active. This bit is set to 1 to stop the DMA data movement and will be cleared when DMA is stopped. Please note this bit is only valid in the device mode. Please note that DMA_START and DMA_ABORT cannot be set simultaneously. Doing so will result in unexpected outcomes. Setting this bit to 1 when the DMA_START is 0 will have no effect.	0
2	DMA_IO (HOV)	RW	DMA I/O to I/O This bit forces the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to '1', DMA_LEN must be an integer multiple of DWORD (Four bytes) and DMA_MADDR must be	0

Vangotech

			aligned to the boundary of DWORD (Four bytes).	
1	DMA_TYPE (HOV)	RW	DMA Type The transfer type of data moving 0: FIFO to Memory 1: Memory to FIFO	0
0	DMA_START (HOV)	RW	DMA Start This register informs the DMA controller to initiate a DMA transfer. This bit is set to start the transfer and is cleared when the DMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either DMA completion or DMA error. Please note that DMA_LEN, DMA_TYPE, and DMA_MADDR must be configured before DMA_START is set.	0

8.1.6.52. Device DMA Controller Parameter Setting 2 Register

Table 8-52. Device DMA Controller Parameter Setting 2 Register (DMA_CPS2, Address = 0x1CC)

Bit	Name	Type	Description	Default Value
[31:0]	DMA_MADDR (HOV)	RW	DMA Memory Address The starting address of memory to request a DMA transfer	0

8.1.6.53. Device DMA Controller Parameter Setting 3 Register

Table 8-53. Device DMA Controller Parameter Setting 3 Register (DMA_CPS3, Address = 0x1D0)

Bit	Name	Type	Description	Default Value
[31:0]	SETUP_CMD_RPORT	RO	Control Transfer Setup Command Read Port CPU reads the 8-byte setup command from this port instead of programming DMA to move this data. Users should note that CPU must read one word each time when it reads this port (A half-word or byte read is not allowed). Before reading this port, the DMA target FIFO (0x1C0) must be set to control the transfer FIFO even though DMA is not used to move this data. Each time the CPU reads this port, the internal FIFO pointer will be increased. Under the condition that the pointer increases erroneously, users should not set the target FIFO to the control transfer FIFO while using the	0

Vangotech

			ICE to scan all ports. Please note that CPU is not allowed to obtain 8-byte setup command from here when VDMA_EN is set.	

8.1.6.54. Device DMA Controller Parameter Setting 4 Register

Table 8-54. Device DMA Controller Parameter Setting 4 Register (DMA_CPS4, Address = 0x1D4)

Bit	Name	Type	Description	Default Value
[31:17]	-	Rsvd	Reserved	0
[16:0]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0

8.1.6.55. Device Virtual DMA CXF Parameter Setting 1 Register

Table 8-55. Device Virtual DMA CXF Parameter Setting 1 Register (VDMA_CXFPS1, Address = 0x300)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
[24:8]	VDMA_LEN_CXF	RW	Virtual DMA Length for CXF These bytes indicate that the total bytes need to be moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'. This field will be handed over to the hardware and be modified according to the DMA transfer status since VDMA_START_CXF is set. After VDMA_CMPLT_CXF or VDMA_ERROR_CXF is asserted, the byte counts represent the remaining bytes that have not been moved by VDMA.	0
[7:5]	-	Rsvd	Reserved	0
[4:3]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
2	VDMA_IO_CXF	RW	Virtual DMA I/O to I/O for CXF This bit will force the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to	0

			'1', both VDMA_LEN_CXF and the control endpoint maximum packet size must be an integer multiple of DWORD (Four bytes) and VDMA_MADDR_CXF must be aligned to the boundary of DWORD (Four bytes).	
1	VDMA_TYPE_CXF	RW	Virtual DMA Type for CXF The transfer type of data moving 0: CX FIFO to Memory 1: Memory to CX FIFO	0
0	VDMA_START_CXF	RW	Virtual DMA Start for CXF This register informs the DMA controller to initiate a VDMA transfer. This bit is set to start the transfer and is cleared when the VDMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either VDMA completion or VDMA error. Please note that VDMA_LEN_CXF, VDMA_TYPE_CXF, and VDMA_MADDR_CXF must be configured before VDMA_START_CXF is set.	0

8.1.6.56. Device Virtual DMA CXF Parameter Setting 2 Register

Table 8-56. Device Virtual DMA CXF Parameter Setting 2 Register (VDMA_CXFPS2, Address = 0x304)

Bit	Name	Type	Description	Default Value
[31:0]	VDMA_MADDR_CXF	RW	Virtual DMA Memory Address for CXF The starting address of memory to request a VDMA transfer This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_CXF is set.	0

8.1.6.57. Device Virtual DMA FIFO0 Parameter Setting 1 Register

Table 8-57. Device Virtual DMA FIFO0 Parameter Setting 1 Register (VDMA_F0PS1, Address = 0x308)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
[24:8]	VDMA_LEN_F0	RW	Virtual DMA Length for FIFO0 These bytes indicate that the total bytes need to be moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'.	0

Vangotech

			This field will be handed over to the hardware and be modified according to the DMA transfer status since VDMA_START_F0 is set. After VDMA_CMPLT_F0 or VDMA_ERROR_F0 is asserted, the byte counts represent the remaining bytes that have not been moved by VDMA.	
[7:5]	-	Rsvd	Reserved	0
[4:3]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
2	VDMA_IO_F0	RW	Virtual DMA I/O to I/O for FIFO0 This bit will force the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to '1', both VDMA_LEN_F0 and the endpoint maximum packet size mapped to F0 must be an integer multiple of DWORD (Four bytes) and VDMA_MADDR_F0 must be aligned to the boundary of DWORD (Four bytes).	0
1	VDMA_TYPE_F0	RW	Virtual DMA Type for FIFO0 The transfer type of data moving 0: FIFO0 to Memory 1: Memory to FIFO0	0
0	VDMA_START_F0	RW	Virtual DMA Start for FIFO0 This register informs the DMA controller to initiate a VDMA transfer. This bit is set to start the transfer and is cleared when the VDMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either VDMA completion or VDMA error. Please note that VDMA_LEN_F0, VDMA_TYPE_F0, and VDMA_MADDR_F0 must be configured before VDMA_START_F0 is set.	0

8.1.6.58. Device Virtual DMA FIFO0 Parameter Setting 2 Register

Table 8-58. Device Virtual DMA FIFO0 Parameter Setting 2 Register (VDMA_F0PS2, Address = 0x30C)

Bit	Name	Type	Description	Default Value
[31:0]	VDMA_MADDR_F0	RW	Virtual DMA Memory Address for FIFO0 The starting address of memory to request a VDMA transfer	0

Vangotech

			This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F0 is set.	

8.1.6.59. Device Virtual DMA FIFO1 Parameter Setting 1 Register

Table 8-59. Device Virtual DMA FIFO1 Parameter Setting 1 Register (VDMA_F1PS1, Address = 0x310)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
[24:8]	VDMA_LEN_F1	RW	Virtual DMA Length for FIFO1 These bytes indicate that the total bytes need to be moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'. This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F1 is set. After VDMA_CMPLT_F1 or VDMA_ERROR_F1 is asserted, the byte counts represent the remaining bytes that have not been moved by VDMA.	0
[7:5]	-	Rsvd	Reserved	0
[4:3]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
2	VDMA_IO_F1	RW	Virtual DMA I/O to I/O for FIFO1 This bit forces the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to '1', both VDMA_LEN_F1 and the endpoint maximum packet size mapped to F1 must be an integer multiple of DWORD (Four bytes) and VDMA_MADDR_F1 must be aligned to the boundary of DWORD (Four bytes).	0
1	VDMA_TYPE_F1	RW	Virtual DMA Type for FIFO1 The transfer type of data moving 0: FIFO1 to Memory 1: Memory to FIFO1	0
0	VDMA_START_F1	RW	Virtual DMA Start for FIFO1 This register informs the DMA controller to initiate a VDMA transfer. This bit is set to start the transfer and is	0

Vangotech

			cleared when the VDMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either VDMA completion or VDMA error. Please note that VDMA_LEN_F1, VDMA_TYPE_F1, and VDMA_MADDR_F1 must be configured before VDMA_START_F1 is set.	

8.1.6.60. Device Virtual DMA FIFO1 Parameter Setting 2 Register

Table 8-60. Device Virtual DMA FIFO1 Parameter Setting 2 Register (VDMA_F1PS2, Address = 0x314)

Bit	Name	Type	Description	Default Value
[31:0]	VDMA_MADDR_F1	RW	Virtual DMA Memory Address for FIFO1 The starting address of memory to request a VDMA transfer This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F1 is set.	0

8.1.6.61. Device Virtual DMA FIFO2 Parameter Setting 1 Register

Table 8-61. Device Virtual DMA FIFO2 Parameter Setting 1 Register (VDMA_F2PS1, Address = 0x318)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
[24:8]	VDMA_LEN_F2	RW	Virtual DMA Length for FIFO2 These bytes indicate that the total bytes need to be moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'. This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F2 is set. After VDMA_CMPLT_F2 or VDMA_ERROR_F2 is asserted, byte counts here represent the remaining bytes that have not been moved by VDMA.	0
[7:5]	-	Rsvd	Reserved	0
[4:3]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0

Vangotech

2	VDMA_IO_F2	RW	Virtual DMA I/O to I/O for FIFO2 This bit will force the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to '1', both VDMA_LEN_F2 and the endpoint maximum packet size mapped to F2 must be an integer multiple of DWORD (Four bytes) and VDMA_MADDR_F2 must be aligned to the boundary of DWORD (Four bytes).	0
1	VDMA_TYPE_F2	RW	Virtual DMA Type for FIFO2 The transfer type of data moving 0: FIFO2 to Memory 1: Memory to FIFO2	0
0	VDMA_START_F2	RW	Virtual DMA Start for FIFO2 This register informs the DMA controller to initiate a VDMA transfer. This bit is set to start the transfer and is cleared when the VDMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either VDMA completion or VDMA error. Please note that VDMA_LEN_F2, VDMA_TYPE_F2, and VDMA_MADDR_F2 must be configured before VDMA_START_F2 is set.	0

8.1.6.62. Device Virtual DMA FIFO2 Parameter Setting 2 Register

Table 8-62. Device Virtual DMA FIFO2 Parameter Setting 2 Register (VDMA_F2PS2, Address = 0x31C)

Bit	Name	Type	Description	Default Value
[31:0]	VDMA_MADDR_F2	RW	Virtual DMA Memory Address for FIFO2 The starting address of memory to request a VDMA transfer This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F2 is set.	0

8.1.6.63. Device Virtual DMA FIFO3 Parameter Setting 1 Register

Table 8-63. Device Virtual DMA FIFO3 Parameter Setting 1 Register (VDMA_F3PS1, Address = 0x320)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0

Vangotech

[24:8]	VDMA_LEN_F3	RW	Virtual DMA Length for FIFO3 These bytes indicate that the total bytes need to be moved by the DMA controller. The unit is byte. The maximum length will be 128 KB - 1 and must not be configured to '0'. This field will be handed over to hardware and modified according to the VDMA transfer status since VDMA_START_F3 is set. After VDMA_CMPLT_F3 or VDMA_ERROR_F3 is asserted, the byte counts represent the remaining bytes that have not been moved by VDMA.	0
[7:5]	-	Rsvd	Reserved	0
[4:3]	-	RsvdP	Reserved and preserved Software shall preserve the value read from these bits for writing to these bits	0
2	VDMA_IO_F3	RW	Virtual DMA I/O to I/O for FIFO3 This bit will force the DMA controller not to toggle an address. This bit will be set when DMA targets the I/O device instead of the system memory. If this register is set to '1', both VDMA_LEN_F3 and the endpoint maximum packet size mapped to F3 must be an integer multiple of DWORD (Four bytes) and VDMA_MADDR_F3 must be aligned to the boundary of DWORD (Four bytes).	0
1	VDMA_TYPE_F3	RW	Virtual DMA Type for FIFO3 The transfer type of data moving 0: FIFO3 to Memory 1: Memory to FIFO3	0
0	VDMA_START_F3	RW	Virtual DMA Start for FIFO3 This register informs the DMA controller to initiate a VDMA transfer. This bit is set to start the transfer and is cleared when the VDMA operation is completed. Please note that this bit cannot be cleared by the software; it can only be cleared by the hardware in the case of either VDMA completion or VDMA error. Please note that VDMA_LEN_F3, VDMA_TYPE_F3, and VDMA_MADDR_F3 must be configured before VDMA_START_F3 is set.	0

8.1.6.64. Device Virtual DMA FIFO3 Parameter Setting 2 Register

Table 8-64. Device Virtual DMA FIFO3 Parameter Setting 2 Register (VDMA_F3PS2, Address = 0x324)

Bit	Name	Type	Description	Default Value
[31:0]	VDMA_MADDR_F3	RW	Virtual DMA Memory Address for FIFO3 The starting address of memory to request a VDMA transfer This field will be handed over to the hardware and be modified according to the VDMA transfer status since VDMA_START_F3 is set.	0

8.1.6.65. Device Interrupt Source Group 3 Register

Table 8-65. Device Interrupt Source Group 3 Register (DEV_ISG3, Address = 0x328)

Bit	Name	Type	Description	Default Value
[31:21]	-	Rsvd	Reserved	0
20	VDMA_ERROR_F3	RW1C	Virtual DMA Error Interrupt for FIFO3 The VDMA operation cannot be finished normally and an error signal has been received. When CPU initiates VDMA to fill up or read out FIFO3 of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
19	VDMA_ERROR_F2	RW1C	Virtual DMA Error Interrupt for FIFO2 The VDMA operation cannot be finished normally and an error signal has been received. When CPU initiates VDMA to fill up or read out FIFO2 of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
18	VDMA_ERROR_F1	RW1C	Virtual DMA Error Interrupt for FIFO1 The VDMA operation cannot be finished normally and an error signal has been received. When CPU initiates VDMA to fill up or read out FIFO1 of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
17	VDMA_ERROR_F0	RW1C	Virtual DMA Error Interrupt for FIFO0 The VDMA operation cannot be finished normally and	0

Vangotech

			an error signal has been received. When CPU initiates VDMA to fill up or read out FIFO0 of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	
16	VDMA_ERROR_CXF	RW1C	Virtual DMA Error Interrupt for CXF The VDMA operation cannot be finished normally and an error signal has been received. When CPU initiates VDMA to fill up or read out CX FIFO of the device, and the DMA controller receives an error response from the system bus, this bit will be set. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
[15:5]	-	Rsvd	Reserved	0
4	VDMA_CMPLT_F3	RW1C	Virtual DMA Completion Interrupt for FIFO3 The VDMA operation has finished normally. When CPU initiates the VDMA to fill up or read out FIFO3 of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
3	VDMA_CMPLT_F2	RW1C	Virtual DMA Completion Interrupt for FIFO2 The VDMA operation has finished normally. When CPU initiates the VDMA to fill up or read out FIFO2 of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
2	VDMA_CMPLT_F1	RW1C	Virtual DMA Completion Interrupt for FIFO1 The VDMA operation has finished normally. When CPU initiates the VDMA to fill up or read out FIFO1 of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0
1	VDMA_CMPLT_F0	RW1C	Virtual DMA Completion Interrupt for FIFO0 The VDMA operation has finished normally. When CPU initiates the VDMA to fill up or read out FIFO0 of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware.	0

Vangotech

			This bit is not affected by a USB bus reset.	
0	VDMA_CMPLT_CXF	RW1C	Virtual DMA Completion Interrupt for CXF The VDMA operation has finished normally. When CPU initiates the VDMA to fill up or read out CX FIFO of the device, this bit will be set after the mission is complete. This bit can only be cleared by the firmware. This bit is not affected by a USB bus reset.	0

8.1.6.66. Device Mask of Interrupt Source Group 3 Register

Table 8-66. Device Mask of Interrupt Source Group 3 Register (DEV_MISG3, Address = 0x32C)

Bit	Name	Type	Description	Default Value
[31:21]	-	Rsvd	Reserved	0
20	MVDMA_ERROR_F3	RW	Mask Virtual DMA Error Interrupt for FIFO3 Mask active the VDMA Error interrupt for FIFO3 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
19	MVDMA_ERROR_F2	RW	Mask Virtual DMA Error Interrupt for FIFO2 Mask active the VDMA Error interrupt for FIFO2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
18	MVDMA_ERROR_F1	RW	Mask Virtual DMA Error Interrupt for FIFO1 Mask active the VDMA Error interrupt for FIFO1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
17	MVDMA_ERROR_F0	RW	Mask Virtual DMA Error Interrupt for FIFO0 Mask active the VDMA Error interrupt for FIFO0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
16	MVDMA_ERROR_CXF	RW	Mask Virtual DMA Error Interrupt for CXF Mask active the VDMA Error interrupt for CXF 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
[15:5]	-	Rsvd	Reserved	0
4	MVDMA_CMPLT_F3	RW	Mask Virtual DMA Completion Interrupt for FIFO3 Mask active the VDMA Completion interrupt for FIFO3 0: Enable the corresponding interrupt	0

Vangotech

			1: Disable the corresponding interrupt	
3	MVDMA_CMPLT_F2	RW	Mask Virtual DMA Completion Interrupt for FIFO2 Mask active the VDMA Completion interrupt for FIFO2 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
2	MVDMA_CMPLT_F1	RW	Mask Virtual DMA Completion Interrupt for FIFO1 Mask active the VDMA Completion interrupt for FIFO1 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
1	MVDMA_CMPLT_F0	RW	Mask Virtual DMA Completion Interrupt for FIFO0 Mask active the VDMA Completion interrupt for FIFO0 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0
0	MVDMA_CMPLT_CXF	RW	Mask Virtual DMA Completion Interrupt for CXF Mask active the VDMA Completion interrupt for CXF 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

8.1.6.67. Device Virtual DMA Control Register

Table 8-67. Device Virtual DMA Control Register (VDMA_CTRL, Address = 0x330)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	VDMA_EN	RW	Virtual Multi-DMA Channel Enable 0: Virtual multi-DMA channel disable 1: Virtual multi-DMA channel enable Enable the virtual DMA scheme to manage VDMA parameter settings for each FIFO (0x300~0x324) automatically. Round-robin arbitration scheme is applied between VDMA tasks for CXF, FIFO0, FIFO1, FIFO2, and FIFO3. Software shall guarantee that DMA is out of service prior to setting this bit to 1. Clear this bit from 1 to 0 make the whole VDMA parameter	0

			setting registers be cleared as well. A DMA abort procedure is also applied by hardware if data is under conveying. Software is able to program DMA_CPS1 again after DMA_START goes to 0.	

8.1.6.68. Device LPM Capability Register

Table 8-68. Device LPM Capability Register (LPM_CAP, Address = 0x334)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	LPM_WAKEUP_EN	RO	Enable LPM Wakeup Capability This bit is updated by the bRemoteWake field of the received LPM token	0

8.1.6.69. Device Interrupt Source Group 4 Register

Table 8-69. Device Interrupt Source Group 4 Register (DEV_ISG4, Address = 0x338)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	L1_INT	RW1C	L1 Interrupt L1-state-change interrupt bit When FOTG210 received LPM token and entered L1, this bit will be set to 1.	0

8.1.6.70. Device Mask of Interrupt Source Group 4 Register

Table 8-70. Device Mask of Interrupt Source Group 4 Register (DEV_MISG4, Address = 0x33C)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	ML1_INT	RW	Mask L1 Interrupt Mask active L1 State Change Interrupt bit 0: Enable the corresponding interrupt 1: Disable the corresponding interrupt	0

Vango Tech

8.2. SSP

8.2.1. Overview

FTSSP010 is a synchronous serial port interface that allows the host processor to serve as a master or a slave. Various devices can be connected to this controller by using the serial protocol. FTSSP010 supports the Synchronous Serial Port (SSP) from Texas Instruments, the Serial Peripheral Interface (SPI) from Motorola, MICROWIRE from National Semiconductor, I2S from Philips and partial LCD driver serial interface formats. The serial data formats may range from 4 bits to 128 bits in length.The SSP controller can use the on-chip DMA to directly transfer data between the external serial device and the system memory without the intervention from the processor.

8.2.2. Features

- Compliant with AMBA 2.0 APB
- Supports TI SSP, Motorola SPI, National Semiconductor MICROWIRE, and Philips I2S frame formats
- Maximum frequency of data transfer bit rate: 20 MHz
- Supports partial LCD driver 4-line serial interface
- Supports independent SSP clock for easy bit clock generation
- Supports master mode and slave mode
- Supports internally or externally controlled serial bit clock
- Supports internally or externally controlled frame/sync. format
- Programmable frame/sync. polarity
- Programmable serial bit clock polarity, phase, and frequency
- Programmable serial bit data sequence (MSB or LSB first)
- Programmable I2S format including zero bit padding and right/left justification
- Programmable threshold interrupt of transmit/receive FIFO
- Independently programs interrupt enable/disable
- Independently configures transmit and receive data FIFO depths
- Selectable serial protocol interfaces

8.2.3. Functional Block Diagram

Figure 8-2 shows the functional block diagram of the Synchronous Serial Port (SSP) controller.

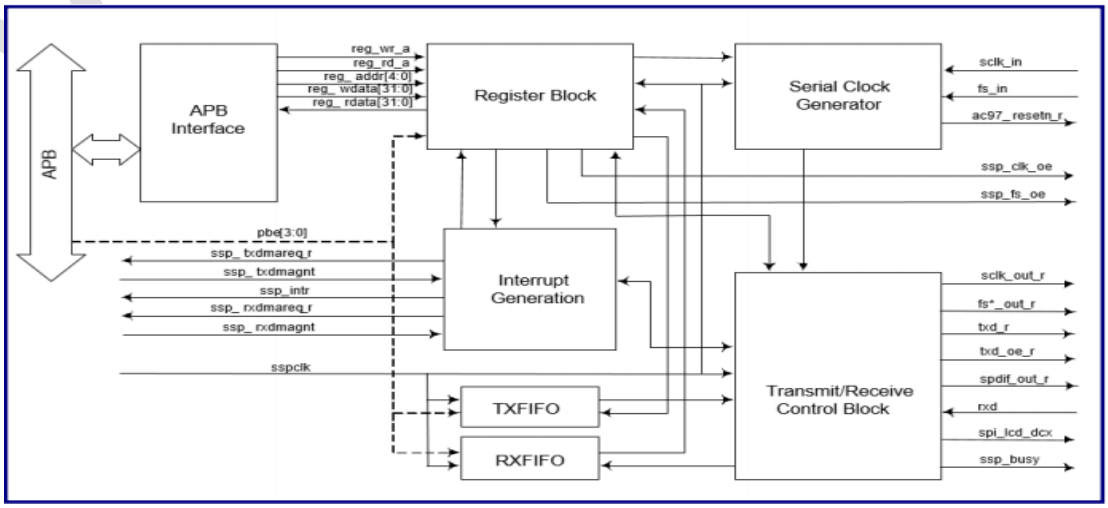


Figure 8-2. Functional Block Diagram

Each block shown in Figure 8-2 is described below:

APB Interface

The AMBA APB interface is used to decode signals from the APB to read/write signals to FIFO. The APB interface of the SSP controller provides the active-high, byte-enable input (As the dashed lines shown in Figure 8-2). If the APB bridge can generate the byte-enable signal, it can directly connect to the APB interface, and the timing of the byte-enable signal should be the same as the timing of PWRITE. If the standard APB is used, four byte-enable inputs can be removed by un-defining FTSSP010_BYTE_ENABLE.The byte-enable information will affect the operation of FIFO.

Register Block

The register block accepts the register read/write from the APB interface. This register block also provides the property information to other blocks of the SSP controller.

Interrupt Generation Control

The interrupt generation control block collects information (FIFO underrun/overrun, FIFO threshold) from the transmit/receive control block and provides the value to the register block. If the interrupt condition matches, the ssp_intr signal will be asserted.

This block also generates the DMA request signals, if necessary.

Serial Clock Generator

This block generates a serial clock for communication. The format of the serial clock is defined through the control register 0. Generally, the clock will not start if the SSP controller is not enabled.Once the SSP controller is enabled and the clock running condition is matched (For example, when TI SSP is specified and the transmit FIFO is not empty, or when I2S is specified), the serial clock will start running. The operating frequency of the serial clock depends on the setting of the SCLKDIV register.

This block also sends the serial clock and frame/sync. information to the data transmit/receive control block.

Transmit/Receive Control

The main function of this block is to perform the parallel-to-serial transmission or handle the serial-to-parallel reception from the external devices.

If the master mode is specified and the transmit FIFO is not empty, the data in the transmit FIFO will be read and shifted out via the txd_r pin. If a whole word is completely shifted out and the transmit FIFO still contains valid data, the next data will be read and shifted out. The received data can be shifted in via the rxd pin. Once the reception is complete, the received word will be written into the receive FIFO, and the receive control logic will continue to receive the next word. If the half-duplex protocol is specified, the receive control logic will not start until the transmission is completed.If the slave mode is specified, the SSP controller will start to transmit/receive data when full-duplex protocol is specified.

8.2.4. Initialization/Application Information

In a series of transfer transfers, user users can not change the transfer size while read ing /writ ing a data. This section provides the initialization sequence of the SSP controller. First of all, the user has to determine whether to use the master or slave mode for the application. Once the mode is determined, please follow the appropriate SSP initialization sequence provided in the following sections.

8.2.4.1. Initialization in Master Mode

The following steps outline the initialization sequence of the data transmission/reception to/from slave in the master mode:

1. Specify the frame format, set the operation mode to master, and specify the polarity of the frame output.
   - If Motorola SPI is specified, users can simultaneously specify SPIFSPO, LSB, SCLKPO, and SCLKPH.
   - If Philips I2S is specified, users can simultaneously specify FSDIST, FSPO, LSB, and FSJSTFY.
2. Specify the bit rate by setting the serial clock divider and serial data length. If the Intel AC-link frame format is specified, the serial data length will be ignored.
3. Fill the transmitted data to the transmit FIFO
4. Enable the transmit or receive interrupt and the DMA request, if required
5. Enable both the SSP and transmit data output. If the Motorola SPI, National Semiconductor MICROWIRE, or Philips I2S frame format is specified, the TXEN and RXEN signals should be enabled.
6. If the Serial Data Length (SDL) is set to be larger than 31 (32 bits) and the transmit FIFO is empty, SSP should be disabled and go to Step 3 for the next transfer. If SDL is set to be less than or equals to 31 and the transmit FIFO is empty, SSP will stop providing fs_out_r and sclk_out_r until users re-fill the transmitted data to the transmit FIFO.

After the initialization in the master mode, the SSP controller will start transmitting/receiving data without user intervention. The SSP controller will report the FIFO status or transmission/reception to/from users if the corresponding interrupt is enabled. The SSP controller will request DMA to transfer data if DMA is enabled. To get an empty interrupt, users can set TFTHOD = 1. When there is only one entry in the FIFO and TFTHOD = 1, then users will get the threshold interrupt. When the transmit logic reads last data out, users will also get the threshold interrupt. Please note that the SSP controller will report that the transmit FIFO is empty before the transmission is complete and the transmit logic is still busy transmitting the last data out. Users have to monitor the busy flag of the SSP controller to confirm the completion of the transmission.

The frame/sync. output is sensitive to the FIFO status. If the last data in the transmit FIFO are transmitted and no more data are filled, frame/sync. will return to the idle state. For an application that is sensitive to the action of frame/sync., users must monitor and make sure that things are under control. For example, the transfer of Atmel AT25080 EEPROM should consist of the command and data phases, and the transfer can be a continue transfer if the transmit FIFO does not run out of data. If the transmit FIFO runs out of data, the chip-enable (Usually frame/sync.) will return to the de-select state, and the user has to restart the next transfer by heading the command phase.

8.2.4.2. Initialization in Slave Mode

The following steps show the initialization sequence for transmitting/receiving data to/from the master in the slave mode:

1. Specify the frame format, set the operation mode to slave, and specify the polarity of the frame output.
   - If Motorola SPI is specified, users should simultaneously specify SCLKPO and SCLKPH.
   - If Philips I2S is specified, users should also specify FSDIST.
   - The SSP controller cannot function as a slave in the Intel AC-link frame format.

Vangotech

2. Specify the serial data length. In the National Semiconductor MICROWIRE or Philips I2S frame format, users can specify the padding data length, if available.
3. Fill the transmit FIFO, if necessary
4. Enable the interrupt of transmit or receive and the DMA request, if necessary
5. Enable SSP and turn on the transmit data output enable, if allowed. If the Motorola SPI, National Semiconductor MICROWIRE, or Philips I2S frame format is specified, the TXEN and RXEN signals should be enabled.

In the slave mode, the external master controls the action of the SSP controller. The SSP controller will operate according to the frame/sync. and serial clock input.

8.2.4.3. Disable/Enable Program Sequence

When FTSSP010 is used in a disabling sequence and then enabling, please note that:

- The transmit FIFO must be cleared before disabling FTSSP010.
- The receive FIFO must be cleared before enabling FTSSP010.

8.2.4.4. SPI Flash Controller Write Command Program Sequence

1. Specify the frame format as SPI and FLASH application indications. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the bit rate by setting the serial clock divider and serial data length in SSP control register 1
3. Enable SSPEN, TXDOE, and TXEN, specify FSOS, and set FS to "LOW" in SSP control register 2
4. Enable the transmit interrupt and DMA request if required
5. Fill the transmitted data into the transmit FIFO
6. Check TFVE and BUSY in the SSP status register; TFVE = 0 and BUSY = 0 indicate that the SSP controller has completed the data transfer, SSPEN and TXEN should then be disabled, FS should be set to "HIGH" to stop the write command
7. Check the SPI FLASH status register; the SSP controller cannot issue a new command unless the write command has completed

8.2.4.5. SPI Flash Controller Read Command Program Sequence

1. Specify the frame format as SPI and FLASH application indications. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the bit rate by setting the serial clock divider and serial data length in SSP control register 1
3. Clear the transmit/receive FIFO, fill the read command code into the transmit FIFO
4. Enable SSPEN, TXDOE, TXEN, and RXEN, specify FSOS, and set FS to "LOW" in SSP control register 2
5. Enable the receive FIFO threshold interrupt and DMA request, if required
6. After all data are received, users should disable SSPEN, TXEN, and RXEN, and set FS to "HIGH" to stop the read command

8.2.4.6. SPI Flash Controller Read Command Control by Write Program Sequence

1. Specify the frame format as SPI, the FLASH application indication, and the FLASHTX bit. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the bit rate by setting the serial clock divider and serial data length in SSP control register 1
3. Clear the transmit/receive FIFO.

4. Enable SSPEN, TXDOE, TXEN, and RXEN; specify FSOS; and set FS to "LOW" in SSP control register 2
5. Enable the transmit/receive FIFO threshold interrupt and DMA request, if required
6. Fill transmit data into transmit FIFO, one transmit data will exchange one receive data. After all data are transmit and received, users should disable SSPEN, TXEN, and RXEN, and set FS to "HIGH" to stop the read command

8.2.4.7. LCD Driver 3-line Serial Interface Write Command & D/CX Control by Write Data Program Sequence

1. Specify the frame format as SPI. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the serial data length including D/CX information (1-bit) and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - If the data length is 8-bit, the total transfer length is 9-bit. The SDL register is set to 8.
3. Clear the transmit FIFO, fill the write command/data into the transmit FIFO
   - If the MSB of the data word will be transmitted, D/CX information must be set on MSB bit.
   - Figure 8-3 shows the write transaction when SDL = 8.
4. Enable SSPEN, TXDOE, and TXEN in SSP control register 2

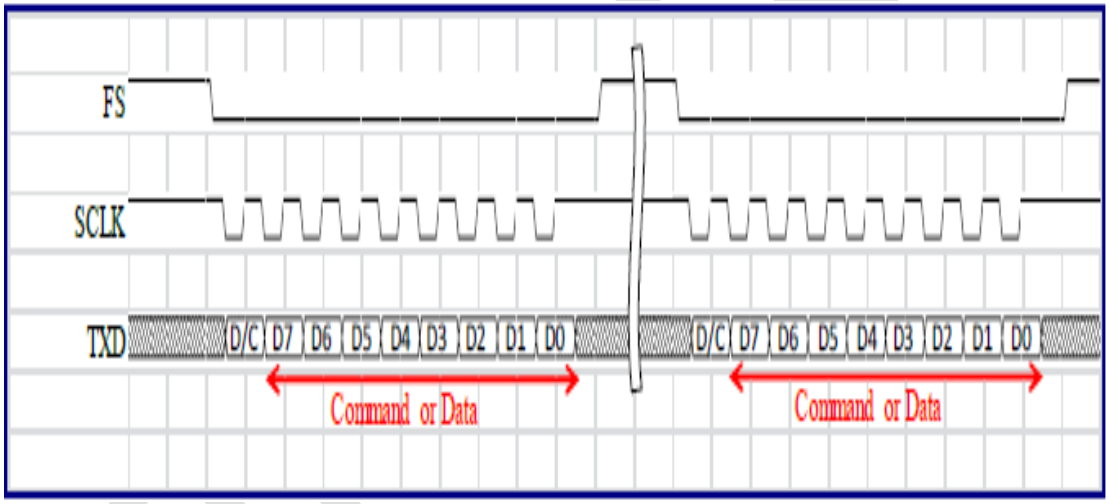


Figure 8-3. LCD Driver 3-line Serial Write Transaction

8.2.4.8. LCD Driver 3-line Serial Interface Write Command & D/CX Control by Register Program Sequence

1. Specify the frame format as SPI and the LCDDCXS & LCDDCX value. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the serial data length including D/CX information (1-bit) and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - Support data length = 1, 2, 3, 4 byte, SDL must be set to 8, 17, 26, 35. Users should input data by bytes format without D/CX bit. The controller will auto insert D/CX bit value.
   - If the SDL register is set to '8'. The total transfer length is 9-bit, one bit for D/CX and 8-bit for transmit data.
   - If the SDL register is set to '17'. The total transfer length is 18-bit, the total transfer is combined by two LCD command/data values.

- If the SDL register is set to '26'. The total transfer length is 27-bit, the total transfer is combined by three LCD command/data values.
- If the SDL register is set to '35'. The total transfer length is 36-bit, the total transfer is combined by four LCD command/data values.

3. Clear the transmit FIFO, fill the 1 ~ 4 byte write command/data value into the transmit FIFO
   - The MSB bit D/CX information will be transmitted by LCDDCX value. Then, transmit 8-bit write command/data value.
   - Figure 8-3 shows the write transaction when SDL = 8.
4. Enable SSPEN, TXDOE, and TXEN in SSP control register 2

8.2.4.9. LCD Driver 4-line Serial Interface Write Command & D/CX Control by Write Data Program Sequence

1. Specify the frame format as SPI. Set the operation mode as master, and have SCLKPO and SCLKPH be set in SSP control register 0
2. Specify the serial data length and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - Only support data length = 8, SDL must be set to 7.
3. Clear the transmit FIFO, fill the write command/data into the transmit FIFO
   - The valid data length is 8, and the 8th bit presents the D/CX value.
   - For example, Figure 8-4 shows the 4-line serial write transaction.
   - The first TX data is 'h0XX, D/CX pin is zero.
   - The second TX data is 'h1XX, D/CX pin is one.
4. Enable SSPEN, TXDOE, and TXEN in SSP control register 2

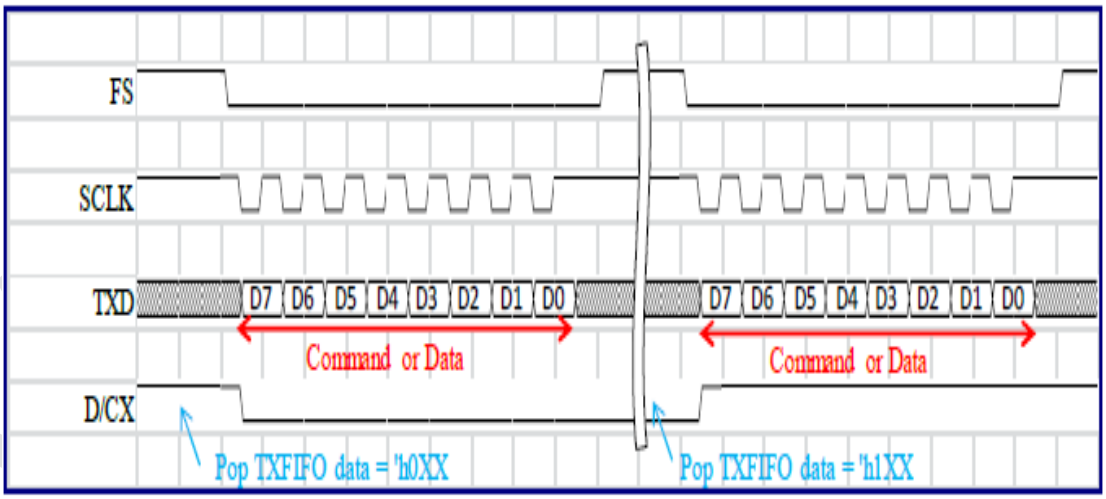


Figure 8-4. LCD Driver 4-line Serial Write Transaction

8.2.4.10. LCD Driver 4-line Serial Interface Write Command & D/CX Control by Register Program Sequence

5. Specify the frame format as SPI and the LCDDCXS & LCDDCX value. Set the operation mode as master, and have SCLKPO & SCLKPH be set in SSP control register 0
6. Specify the serial data length and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - Support data length = 1, 2, 3, 4 byte, SDL must be set to 7, 15, 23, 31.

- The SDL should not set to 8. If the SDL is set to 8, the LCDDCX value will replace 8th bit.

7. Clear the transmit FIFO, fill the write command/data value into the transmit FIFO
   - The valid data length is 1, 2, 3, 4 byte, and the D/CX value is controlled by LCDDCX value.
8. Enable SSPEN, TXDOE, and TXEN in SSP control register 2

8.2.4.11. LCD Driver 3-line Serial Interface Read Command Program Sequence

1. Specify the frame format as MICROWIRE and the operation mode as master in SSP control register 0
2. Specify the padding data length including D/CX information (1-bit) for command length, the serial data length for read data length, and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - If the command length is 8-bit, the total transfer length is 9-bit. The PDL register is set to 8.
   - If the read data length is 8-bit and no stall cycle between command and read data, the SDL register is set to 6.
   - If the read data length is 8-bit and one stall cycle between command and read data, the SDL register is set to 7.
3. Clear the transmit/receive FIFO, fill the command/data into the transmit FIFO
4. Enable SSPEN, TXDOE, TXEN, and RXEN in SSP control register 2

8.2.4.12. LCD Driver 4-line Serial Interface Read Command Program Sequence

1. Specify the frame format as MICROWIRE and the operation mode as master in SSP control register 0
2. Specify the padding data length include D/CX information (1-bit) for command length, the serial data length for read data length, and the bit rate by setting the serial clock divider and serial data length in SSP control register 1
   - Only support command length = 8, PDL must be set to 7.
   - If the read data length is 8-bit and no stall cycle between command and read data, the SDL register is set to 6.
   - If the read data length is 8-bit and one stall cycle between command and read data, the SDL register is set to 7.
3. Clear the transmit/receive FIFO, fill the command into the transmit FIFO
   - The valid data length is 8, and the 8th bit presents the D/CX value.
4. Enable SSPEN, TXDOE, TXEN, and RXEN in SSP control register 2

8.2.5. Register List

Module Name	Base Address
SPI0	0x54B0_0000
SPI1	0x54C0_0000
SPI2	0x54D0_0000
SPI3	0x5650_0000
SPI4	0x5660_0000
SPI5	0x5670_0000
SPI6	0x5680_0000
SPI7	0x2210 0000

Module Name	Base Address

Vangotech

Module Name	Base Address
I2S_0	0x54A0 0000
I2S_1	0x5640 0000

Register Name	Offset	Description
SSPCR0	000	SSP control register 0
SSPCR1	004	SSP control register 1
SSPCR2	008	SSP control register 2
SSPStatus	00C	SSP status register
IntrCR	010	SSP interrupt control register
IntrStatus	014	SSP interrupt status register
TxRxDR	018	SSP transmit/receive data register
SSPCR3	01C	SSP control register 3
SSPRevision	060	SSP revision register
SSPFeature	064	SSP feature register

8.2.6. Register Description

The following subsections provide the detailed information of the SSP control registers.

8.2.6.1. SSP Control Register 0

This register controls the generation and behavior of frame/sync.

Table 8-71. SSP Control Register 0 (SSPCR0, Offset = 0x00)

Bit	Name	Type	Description	Default Value
[31:22]	-	Rsvd	Reserved and read as zero	0
21	LCDDCX	RW	LCD D/CX value If LCDDCXS is set to '1', this bit will output to LCD D/CX bit/signal.	0
20	LCDDCXS	RW	LCD D/CX Select If this bit is set to '0', the LCD D/CX bit/signal is controlled by TxRxDR bit 8. If this bit is set to '1', the LCD D/CX bit/signal is controlled by SSPCR0 LCDDCX bit. The below content describes more detail. In 3-line mode and the data length is 1, 2, 3, or 4 bytes, the SDL must be set to 8, 17, 26, or 35. In this mode, users don't need to insert D/CX bit. The controller will auto-insert D/CX bit by register LCDDCX setting. In 4-line mode and the data length is 1, 2, 3, or 4 bytes, the SDL must be set to 7, 15, 23, or 31. LCD D/CX signal will output by LCDDCX setting.	0

Vantotech

19	SPICONTX	RW	SPI Continuous Transfer control If this bit is set to '1', the SPI continuous transfer behavior can continuously transfer without frame/sync. is asserted when the Transmit FIFO is not empty.	0
18	FLASHTX	RW	FLASH Transmit control If this bit is set to '1' and both TXEN/RXEN are enabled, sclk will stop when no transmit data in FLASH mode. If this bit is set to '0' and both TXEN/RXEN are enabled, sclk will still trigger when no transmit data in FLASH mode until the RX FIFO is full. This bit will take effect only when the SPI frame format and FLASH bit are specified.	0
17	FSFDBK	RW	This bit indicates that fs_in is the internal feedback from fs_out_r. If the master mode is specified, FTSSP010 will need fs_in for reference. If this bit is set to '1', fs_in will be the internal feedback from fs_out_r. If this bit is set to '0', fs_in will be inputted from input pin.	0
16	SCLKFDBK	RW	This bit indicates that sclk_in is the internal feedback from sclk_out_r. If the master mode is specified, FTSSP010 will need sclk_in for reference. If this bit is set to '1', sclk_in will be the internal feedback from sclk_out_r. If this bit is set to '0', sclk_in will be inputted from the input pin.	0
15	SPIFSPO	RW	Frame/Sync. polarity for the SPI mode It is different to FSPO in bit 5. If this bit is set to '1', frame/sync. will be regarded as active high. If this bit is set to '0', frame/sync. will be regarded as active low. This bit will take effect only when the SPI frame format is specified.	0
[14:12]	FFMT	RW	Frame format This register defines the pre-defined frame format according to the following encodings:	0

Vangotech

			000: Texas Instrument Synchronous Serial Port (SSP) 001: Motorola Serial Peripheral Interface (SPI) 010: National Semiconductor MICROWIRE 011: Philips I2S 100 ~ 111: Not defined If these bits are set, the operation of SSP will be unpredictable.	
11	FLASH	RW	This bit indicates that the current application is SPI Flash. This bit will take effect only when the SPI frame format is specified. When this bit is set to '1', the Frame/Sync will control by SSPCR2 FS bit. When RXEN is enabled and FLASHTX is set to '0', the controller will continuous receive data until the RX FIFO is full. When TXEN and RXEN are enabled, and FLASHTX is set to '1', the controller will continuous transmit/receive data until TX FIFO is empty or RX FIFO is full.	0
10	-	Rsvd	Reserved and read as zero	0
[9:8]	FSDIST	RW	Frame/Sync. and data distance These bits define the relationship between the frame/sync. and the data bits. If these bits are set to zero, the first bit of serial data will be valid in the first cycle of frame/sync. If non-zero is specified, FSDIST will define the number of SCLK cycles between the first bit of the valid received data and the frame start. For standard I2S frame format, these bits must be set to '1'. These bits will take effect only when the I2S frame format is specified.	0x1
7	LBM	RW	Loopback mode If this bit is set to '1', the transmitted data will be connected to the received data internally. This bit is used for self-test only. If this bit is set to '0', SSP will operate in the normal mode and the transmitted/received data will be independent.	0
6	LSB	RW	Bit sequence indicator If this bit is set to '0', the most significant bit (MSB) of a data word will be transmitted or received first. If this bit is set to '1', the least significant bit (LSB) of a data word will be transmitted or received first. This bit will take effect only when the I2S or SPI frame	0

Vangotech

			format is specified.	
5	FSPO	RW	Frame/Sync. polarity If this bit is set to '0', the frame/sync. will be regarded as active high. If this bit is set to '1', the frame/sync. will be regarded as active low. This bit will take effect only when the I2S or MWR frame format is specified.	0
4	FSJSTFY	RW	Data justify If this bit is set to '0' and the Padding Data Length (PDL) in the control bit is not zero, the number of zeros specified in PDL will be appended in the back of the serial data. If this bit is set to '1', the padding data will be in front of the serial data. This bit will be valid only when the I2S frame format is specified.	0
[3:2]	OPM	RW	Operation mode If the SSP, SPI, or MICROWIRE frame format is specified, these bits will define the operation modes as follows: 00, 01: Slave mode 10, 11: Master mode If the I2S frame format is specified, these bits will define the operation modes as follows: 11: Master stereo mode 10: Master mono mode 01: Slave stereo mode 00: Slave mono mode	0x3
1	SCLKPO	RW	SCLK polarity If this bit is set to '0', SCLK will remain low when SSP is idle. If this bit is set to '1', SCLK will remain high when SSP is idle. This bit will take effect only if Motorola SPI is specified.	0
0	SCLKPH	RW	SCLK phase This bit defines the relationship between SCLK and frame/sync. If this bit is set to '0', SCLK will start toggling after one SCLK cycle time when frame/sync. is activated. If this bit is set to '1', SCLK will start running after half an SCLK cycle time when frame/sync. is activated.	0

Vangotech

			This bit will take effect only if Motorola SPI is specified.	

8.2.6.2. SSP Control Register 1

This register defines the clock divider and the data length of transfer.Table 8-72 shows the detailed format of SSP control register 1.

Table 8-72. SSP Control Register 1 (SSPCR1, Offset = 0x04)

Bit	Name	Type	Description	Default Value
[31:24]	PDL	RW	Padding data length These bits are relevant only when the Philips I2S or National Semiconductor MICROWIRE frame format is specified. In the Philips I2S frame format, the actual data length is equal to these bits. In National Semiconductor MICROWIRE frame format, the actual data length equals to these bits plus one.	0
23	-	Rsvd	Reserved and read as zero	0
[22:16]	SDL	RW	Serial data length These bits define the bit length of a transmitted/received data word. The actual data length equals to these bits plus one. The minimum SDL value should not be fewer than four. If National Semiconductor MICROWIRE is specified, these bits will define the bit length of the second phase (Receive in the master mode or transmit in the slave mode). If SPDIF is specified, these bits can only be set to F, 13, or 17. If the I2S frame format is specified, these bits can be set to the maximum of 127.	0x07
[15:0]	SCLKDIV	RW	SCLK divider These bits define the serial clock divider. These bits should larger than 2, 4 or 6 by setting.	0x8000

If the Philips I^2S frame format is specified, PDL will be used to define the bit length of the padding bits in front/back of a data word. If PDL is set to '0', no padding bits will be inserted. If a non-zero value is specified, the actual number of zeroes will be inserted/appended in front/back of the data word.If the National Semiconductor MICROWIRE frame format is specified, PDL will define the bit length of the first phase (Transmit in the master mode and receive in the slave mode), and the actual data length will be equal to this register plus one. The maximum value of this register should not exceed 31 if National Semiconductor MICROWIRE is specified. The minimum value of this register should be more than one. The minimum value of SDL should not be less than four.

8.2.6.3. SSP Control Register 2

The minimum value of SDL should not be less than four.

This register defines the SSP enable/disable and output enable of transmitted data. Table 8-73 shows the format of SSP control register 2.

Table 8-73. SSP Control Register 2 (SSPCR2, Offset = 0x08)

Bit	Name	Type	Description	Default Value
[31:12]	-	Rsvd	Reserved and read as zero	0
[11:10]	FSOS	RW	Frame/Sync. Output Select When the SPI frame format is specified and these two bits are set as: 00: Frame/Sync. output from the output port, SPIx_CS (x: 0, 1, 2, …, 7) 01: Frame/Sync. output from the output port, SPIx_CS2 10: Frame/Sync. output from the output port, SPIx_CS3 11: Frame/Sync. output from the output port, SPIx_CS4 These bits will take effect only if the SPI frame format is specified.	0
9	FS	RW	Frame Sync. output This bit controls the frame/sync. output level in the SPI frame format. These bits will take effect only if the SPI frame format is specified and SSPCR0 FLASH bit is set to '1'. If this bit is set to '0', the FS output will be "LOW". If this bit is set to '1', the FS output will be "HiGH".	0
8	TXEN	RW	Transmit Function Enable When the I2S, SPI, or MICROWIRE frame format is specified, the transmit and receive functions will work independently. This bit controls the transmit function. 1: Transmit function is enabled. 0: Transmit function is disabled. For the SPI and MICROWIRE frame formats, this bit can only be changed when the SSP controller is idle.	0
7	RXEN	RW	Receive Function Enable When the I2S, SPI, or MICROWIRE frame format is specified, the transmit and receive functions will work independently. This bit controls the receive function. 1: Receive function is enabled. 0: Transmit function is disabled. For the SPI and MICROWIRE frame formats, this bit can only be changed when the SSP controller is idle.	0
6	SSPRST	RW	SSP Reset The software reset of the SSP controller state machine Writing '1' to this bit will reset the SSP controller state	0

Vangotech

			machine. Writing '0' to this bit will have no effect. The read of this bit is always '0'.	
5	-	Rsvd	Reserved and read as zero	0
4	-	Rsvd	Reserved and read as zero	0
3	TXFCLR	RW1C	Transmit FIFO Clear This is a write only bit. If this bit is written as '1', all data in the transmit FIFO will be cleared. Writing this bit as '0' takes no effect. Reading this bit always gets a value of zero.	0
2	RXFCLR	RW1C	Receive FIFO Clear This is a write only bit. If this bit is written as '1', all data in receive FIFO will be cleared. Writing this bit as '0' takes no effect. Reading this bit always gets a value of zero.	0
1	TXDOE	RW	Transmit Data Output Enable This bit is only valid when the slave mode is specified. In the multi-slave system, it is possible for an SSP master to broadcast a message to all slaves within the system while ensuring that only one slave drives data onto its serial output line. In such a system, the rxd line from multiple slaves can be tied together. To operate in such a system, TXDOE may be cleared if the SSP slave is not supposed to drive the txd_r line. When TXDOE is set, SSP will drive data on the transmit data line. When TXDOE is not set, SSP will not drive data on the transmit data line. If the I2S frame format is specified, this bit will define either the receiving (Recording) mode (0) or the simultaneous transmitting/receiving (Playing/Recording) mode (1). If the SPI frame format and slave mode are specified, TXDOE should also be set to high.	1
0	SSPEN	RW	SSP Enable If this bit is set to '1', the SSP controller will start transmitting/receiving data if possible. If this bit is set to '0', the serial data will stop toggling. Please note that when the SSPEN bit is cleared in an active operation, SSP will stop transmitting or receiving after the current word is transmitted or received. If the AC-link frame format is specified, SSP_EN will be cleared if AC97CRST is set. Writing to this bit will be ignored if the AC97 reset is in progress.	0

Writing '1' to SSPRST will reset the SSP state machine even if the normal transmission or reception is in progress. Usually, this bit will be written when the software stops the SSP controller by setting SSPEN to '0'. For the slave mode of the SSP controller, the serial clock will be controlled by the external master and will be stopped under certain circumstances, and will cause the SSP state machine to halt. Writing '1' to this bit puts the SSP controller back to the idle state when carrying out the following operations.

Vangotech

8.2.6.4. SSP Status Register

This register provides the SSP internal status for the user to take corresponding actions. It shows the raw status before interrupt is enabled. Table 8-74 shows the format of this register.

Table 8-74. SSP Status Register (SSPStatus, Offset = 0x0C)

Bit	Name	Type	Description	Default Value
[31:18]	-	Rsvd	Reserved and read as zeros	0
[17:12]	TFVE	RO	Transmit FIFO Valid Entry These bits indicate the number of entries in the transmit FIFO waiting to be transmitted.	0
[11:10]	-	Rsvd	Reserved and read as zero	0
[9:4]	RFVE	RO	Receive FIFO Valid Entry These bits indicate the number of entries in the receive FIFO waiting for DMA or host processor to read them.	0
3	-	Rsvd	Reserved and read as zero	0
2	BUSY	RO	Busy Indicator If this bit is read to '1', it will indicate that SSP is transmitting and/or receiving data. If this bit is read to '0', it will indicate that SSP is idle or disabled.	0
1	TFNF	RO	Transmit FIFO Not Full This bit will be set to '1' whenever the transmit FIFO is available for DMA or host processor to write. This bit will be cleared to '0' when FIFO is completely full.	1
0	RFF	RO	Receive FIFO Full This bit will be set to '1' whenever the receive FIFO is full. This bit will be cleared to '0' when the DMA controller or host processor reads data from the receive FIFO.	0

8.2.6.5. Interrupt Control Register

This register controls the interrupt generation from SSP. Table 8-75 shows the format of the interrupt control register. The receive FIFO threshold depends on RFTHOD and RFTHOD_UNIT. If RFTHOD_UNIT = 1, the actual receive FIFO threshold will equal to RFTHOD plus one. If RFTHOD_UNIT = 0, the actual receive FIFO threshold will equal to RFTHOD.

Table 8-75. Interrupt Control Register (IntrCR, Offset = 0x10)

Bit	Name	Type	Description	Default Value
[31:19]	-	Rsvd	Reserved and read as zeros	0
18	TXCIEN	RW	Transmit Data Complete Interrupt Enable If this bit is set to '1', the interrupt will take place when	0

Vangotech

			transmit is done and TX valid entry is empty. If this bit is set to '0', the interrupt will be disabled.	
17	RFTHOD_UNIT	RW	Receive FIFO Threshold Unit If this bit is set to '1', the RxFIFO threshold = RFTHOD + 1. If this bit is set to '0', the RxFIFO threshold = RFTHOD.	0
[16:12]	TFTHOD	RW	Transmit FIFO Threshold If the valid data in the transmit FIFO is equal to or less than the actual threshold, the DMA request and/or the interrupt will be asserted. If this bit is set to '0', the interrupt will be disabled.	0x02
[11:7]	RFTHOD	RW	Receive FIFO Threshold If the valid data in the receive FIFO is equal to or greater than the actual threshold, the DMA request and/or the interrupt will be asserted. If this bit is set to '0', the interrupt will be disabled.	0x02
6	-	Rsvd	Reserved and read as zero	0
5	TFDMAEN	RW	Transmit DMA Request Enable If this bit is set to '1', the DMA request will be issued when the transmit FIFO threshold is hit. If this bit is set to '0', no DMA request will be issued.	0
4	RFDMAEN	RW	Receive DMA Request Enable If this bit is set to '1', the DMA request will be issued when the receive FIFO threshold is hit. If this bit is set to '0', no DMA request will be issued.	0
3	TFTHIEN	RW	Transmit FIFO Threshold Interrupt Enable If this bit is set to '1', the interrupt will be issued when the valid entries in the transmit FIFO are less than or equal to the threshold value.	0
2	RFTHIEN	RW	Receive FIFO Threshold Interrupt Enable If this bit is set to '1', the interrupt will be issued when the valid entries in the receive FIFO are greater than or equal to threshold value.	0
1	TFURIEN	RW	Transmit FIFO Underrun Interrupt Enable If this bit is set to '1', the transmit FIFO underrun will cause SSP to assert interrupt. If this bit is set to '0', the interrupt will be masked even when the transmit FIFO underrun happens.	0
0	RFORIEN	RW	Receive FIFO Overrun Interrupt Enable If this bit is set to '1', the receive FIFO overrun will cause SSP to assert interrupt.	0

Vangotech

			If this bit is set to '0', the interrupt will be masked even when the receive FIFO overrun occurs.	

8.2.6.6. Interrupt Status Register

This register shows the current interrupt status. Table 8-76 shows the format of the interrupt status register.

Table 8-76. Interrupt Status Register (IntrStatus, Offset = 0x14)

Bit	Name	Type	Description	Default Value
[31:5]	-	Rsvd	Reserved and read as zero	0
5	TXCI	RO	Transmit Data Complete Interrupt This bit is take place when IntrCR TXCIEN is set to '1'. When transmit is done and TX valid entry is empty, this bit will be set to '1'. This bit will never be cleared to '0' until users read this bit.	0
4	-	Rsvd	Reserved and read as zero	0
3	TFTHI	RO	Transmit FIFO Threshold Interrupt If the transmit FIFO is equal to or less than the threshold, this bit will be set to '1'. If the valid entries in the transmit FIFO are larger than TFTHOD, this bit will be automatically cleared.	1
2	RFTHI	RO	Receive FIFO Threshold Interrupt If the receive FIFO is equal to or greater than the threshold, this bit will be set to '1'. This bit will be cleared when the condition above is removed.	0
1	TFURI	RC	Transmit FIFO Underrun Interrupt If the transmit logic tries to retrieve data from the empty transmit FIFO, this bit will be set to '1'. This bit will never be cleared to '0' until users read this bit.	0
0	RFORI	RC	Receive FIFO Overrun Interrupt If the receive logic tries to receive data when the receive FIFO is full, this bit will be set to '1'. This bit will never be cleared to '0' until users read this bit.	0

8.2.6.7. SSP Transmit/Receive Data Register

The SSP controller separates the transmitting and receiving FIFOs, which are 32-bit wide and can be configured separately. The transmitted and received data occupy the same address space. The write operation writes data to the transmit FIFO and the read operation reads data from the receive FIFO. If the size of a serial data length is less than 32 bits, data will be automatically right-justified during reception or transmission. If the size of a serial data length is larger than 32 bits, the data will be processed from LSB to MSB based on the 32-bit data width during receiving or transmitting data.

Vangotech

8.2.6.8. SSP Control Register 3

This register defines the SPI padding cycle between transactions. Table 8-77 shows the format of SSP control register 3.

- For PCL > 0, all SPI modes of operation will generate discrete transactions with the (PCL + 1) SPI clocks between transactions
- For PCL = 0 and SCLKPH = 1 (SCLKPH is the bit [0] of SSP control register 0), a continuous transaction will occur if TXFIFO is filled prior to enabling the TXEN bit (SSP control register, bit [8]).
- For PCL = 0 and SCLKPH = 0, the multiple discrete transactions will occur.

Table 8-77. SSP Control Register 3 (SSPCR3, Offset = 0x1C)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved and read as zeros	0
[23:16]	DPDL	RW	Different Padding Data Length These bits are relevant only when the Philips I2S frame format is specified. In the I2S stereo mode, DPDL is used when FS equals to zero.	0
[15:13]	-	Rsvd	Reserved and read as zeros	0
12	DPDLEN	RW	Different Padding Data Length Enable This bit is relevant only when the Philips I2S frame format is specified.	0
[11:10]	-	Rsvd	Reserved and read as zeros	0
[9:0]	PCL	RW	Padding Cycle Length These bits are relevant only when the SPI frame format is specified. In the master mode, SPI will wait (PCL + 1) SCLK cycles between each successive transfer.	0

8.2.6.9. SSP Revision Register

This register shows the revisions of the current SSP controller. Table 8-78 shows the format of the revision register.

Table 8-78. SSP Revision Register (SSPRevision, Offset = 0x60)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved and read as zero	0
[23:16]	MAJOR_REV	RO	Major revision number	0x01
[15:8]	MINOR_REV	RO	Minor revision number	0x28
[7:0]	REL_REV	RO	Release number	0x0

Vangotech

8.2.6.10. SSP Feature Register

Since the SSP controller is highly configurable, individual functions of the SSP controller can be enabled or disabled before the real design is synthesized. Table 8-79 shows the format of the feature registers.

Table 8-79. SSP Feature Register (SSPFeature, Offset = 0x64)

Bit	Name	Type	Description	Default Value
31	-	Rsvd	Reserved and read as zero	0x0
[30:29]	EXT_FSNUM	RO	These bits extend the frame/sync. output number configuration.	0x0
28	-	Rsvd	Reserved and read as zero	0
27	SSP_FCFG	RO	TI SSP function configurations	0x1
26	SPIMWR_FCFG	RO	Motorola SPI and National Semiconductor MICROWIRE function configurations	0x1
25	I2S_FCFG	RO	Philip I2S function configurations	0x1
24	-	Rsvd	Reserved and read as zero	0
[23:16]	TXFIFO_DEPTH	RO	Transmit FIFO depth configuration The actual depth is the read value plus one.	0x1f
[15:8]	RXFIFO_DEPTH	RO	Receive FIFO depth configuration The actual depth is the read value plus one.	0x1f
[7:0]	FIFO_WIDTH	RO	Transmit/Receive FIFO width The actual width is the read value plus one. This bit is always read as 31.	0x1f

For the function configuration register bits (SSP_FCFG, SPIMWR_FCFG, and I2S_FCFG), when the register value is '0', it indicates that the corresponding frame format function is not configured in the current design so that the control register relevant to the corresponding frame format will take no effect. When the register is set to '1', it indicates that the corresponding frame format function is configured into the current design.

The FIFO depth register indicates the transmit/receive FIFO depth. The actual depth of the corresponding FIFO is the read value plus one. The transmit/receive FIFO threshold in the Interrupt Control Register (ICR) should not exceed this value. Otherwise, there may be an unpredic result.

Vangotech

8.3. SPIF

8.3.1. Overview

FTSPI020 contains one command-based slave port, one XIP (Direct Address Mapped Read) slave port,and one SPI interface controller to execute the SPI Flash command.The command slave port is used to access the register and data port. The XIP slave port is used to read data of direct address mapped.Meanwhile,these two slave ports can be the AMBA AHB bus port.Moreover,FTSPI020 also provides the PIO mode to access data from the AHB data port.

8.3.2. Features

- Supports DTR mode
- Supports synchronous/asynchronous clock modes between core and SPI Flash interface
- Supports CPU PIO mode
- Supports SPI serial mode, Dual/Quad output mode, and Dual/Quad I/O mode
- Supports XIP (Direct Address Mapped Read) port for boot and execution functions
- Supports prefetch (with 2-way associate read cache) function in XIP port in AHB bus
- Supports programmable divider

8.3.3. Block Diagram

Figure 8-5 shows a block diagram of the SPI.

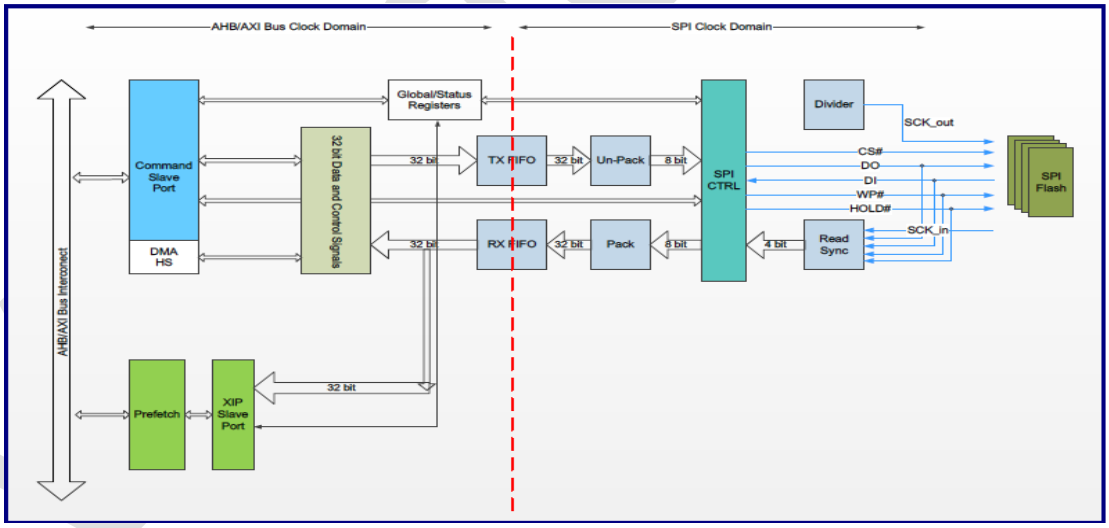


Figure 8-5. SPI Block Diagram

8.3.4. Operations and Functional Descriptions

8.3.4.1. Command Slave Port

FTSPI020 has one command slave port, including the register and data port. Users can access the register or data port through this slave port. Command queue and 0x30 register must be accessed by the WORD size; however, other registers, including the data port, can be accessed by the BYTE, HALFWORD, and WORD size. Besides, the data port can be accessed by DWORD for a data width of 64 bits. Users must

issue the fixed address when accessing the data port. The command slave port has limitation depending on the interface type. The limitation is listed in the following table. Please note that there are some limitations while FTSPI020_IF_AXI is configured:

Table 8-80. the command slave port limitation

Host Type	AHB (32 bits)	AHB (64 bits)	AXI (64 bits)[1]
Transfer Size for Register	Byte, Half WORD, WORD	Byte, Half WORD, WORD	Byte, Half WORD, WORD, Double WORD (ARLen/AWLen must be 0.)
Transfer Size for Data Port (0x100)	Byte, Half WORD, WORD	Byte, Half WORD, WORD, Double WORD	Byte, Half WORD, WORD, Double WORD

[1]The AXI interface types do not support Unaligned Address and Sparse Write Write, but it supportsupports Narrow Transfer.

8.3.4.2. PIO Mode

Users can access the data port via the PIO mode. In the PIO mode, users must poll the RxFIFO/TxFIFO ready register (0x18). When RXFIFO is ready, users can read the whole RxFIFO data or remain data of RxFIFO out. When TxFIFO is ready, users can write the whole data into TxFIFO. Whenever operating in the PIO mode, users have to make sure to poll the status.

8.3.4.3. XIP Slave Port

If FTSPI020_XIP_PORT_ON is configured, FTSPI020 will have one Direct Address Mapped Read (XIP) slave port. Users can directly access the data of system address through this slave port. Besides, users can only access 0x10 register to select the command slave port or XIP port, or read other registers except data register (0x100) at this time. The XIP slave port has limitation depending on the interface type. The limitation is listed in the following table:

Table 8-81. The XIP slave port limitation

Host Type	AHB (32 bits)	AHB (64 bits)	AXI (64 bits)[1]
Transfer Size	Byte, Half WORD, WORD	Byte, Half WORD, WORD, Double WORD	Byte, Half WORD, WORD, Double WORD

[2] The AXI interface typedoesnot support Unaligned Address and Sparse Write, but it supports Narrow Transfer.

Please note that the information of the abort function (0x10, bit[8]) cannot be referred while using the XIP port to read data from Flash (It means that bit[20] of 0x10 is 1), and the abort function cannot abort the XIP port. Meanwhile, the XIP port only supports the read operation; if users issue the write operation to the XIP port, it will lead to unexpected behavior. Users should follow the port switch sequence to do port switching. There must be another main memory in the user system so that the port switching code can be loaded into the main memory first and do port switching on main memory in this release.

8.3.4.4. XIP Prefetch Function

When the macro, FTSPI020_PFC_ON, is defined, the XIP port supports the read prefetch function with 2-way as sociate read cache. Users can decide whether bypass cache or not after hardware reset by the macromacro, FTSPI020_BYPASS_VAL VAL. Due to the SRAM taa may be larger than one period of the bus clock; userusers need to consider the read cycle number by the macro macro, FTSPI020_RD_LATENCY_VAL VAL, or register 0x34.

Besides, if the hardware configuration configurations, FTSPI020_XIP_PORT_ON and FTSPI020_PFC_ON ON, are defined defined; thenthen, there are extra 8 words/double double-words (depend ing on the AHB bus width) direct read read-only zero zero-wait state region, which address range is 0x0 ~ 0x1f/0x3f (depending on the AHB bus width) on F lash. The zero zero-wait state region will be flush if port switching is executedexecuted.

8.3.4.5. SPI Flash Operation

A: Serial Mode

The host controller supports the SPI protocols, mode 0 and mode 3. Users must set the command queue register based on the specification of the SPI Flash. Please refer to the waveforms shown in Figure 8-6 through Figure 8-7 to set the command queue.

If users want to execute the write enable command, please set the instruction code to 06h, write enable, and '1' for the instruction length. Please refer to Figure 8-6 for more details.

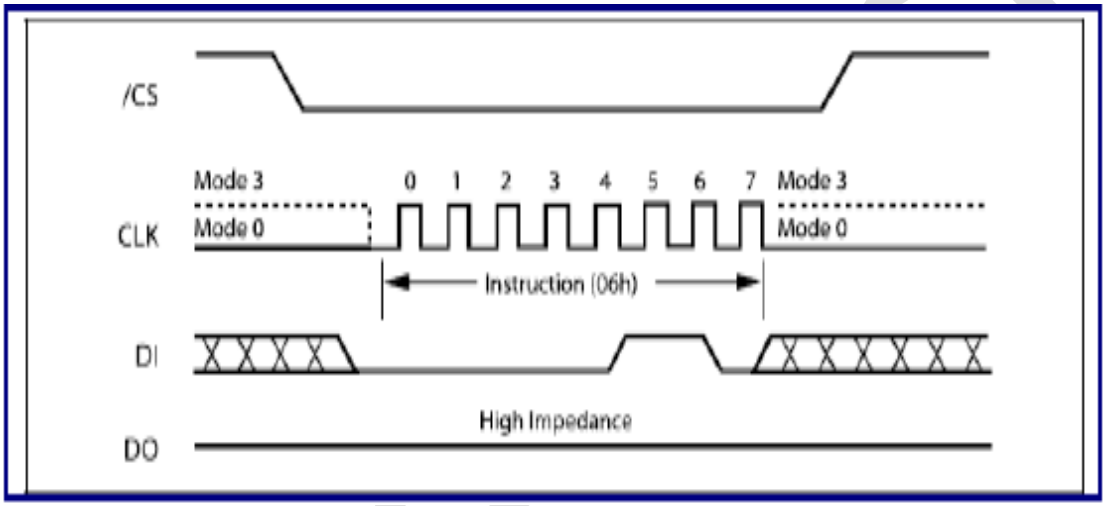


Figure 8-6. Write Enable

If users want to execute the page program command, please set the instruction code to 02h, address register, address length, write enable, and '1' for the instruction length.

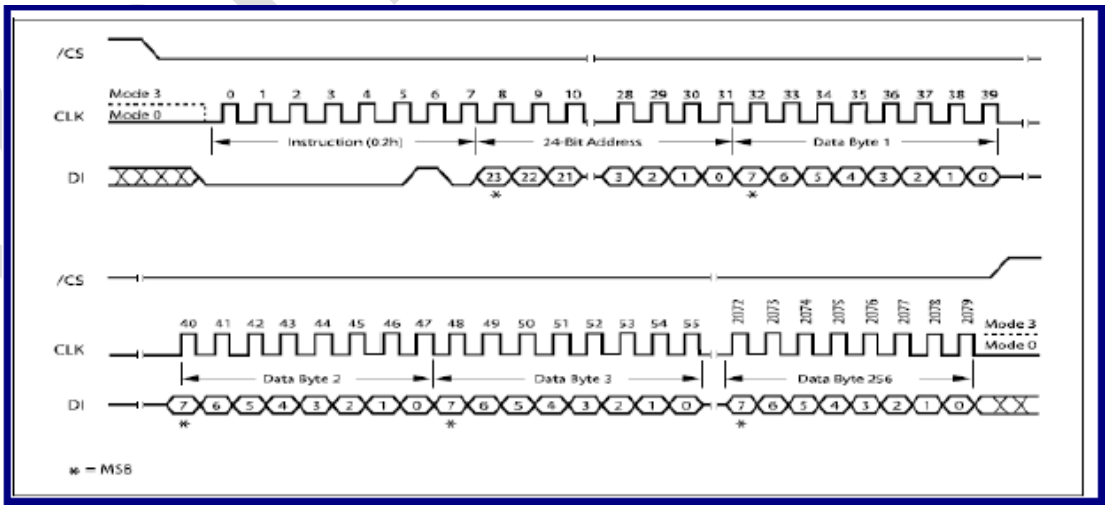


Figure 8-7. Page Program

If users want to execute the read status command, please set the instruction code to 05h/35h, read status enable, optional read status by hardware/software, and '1' for the instruction length. Please refer to Figure 8-8 for more details.

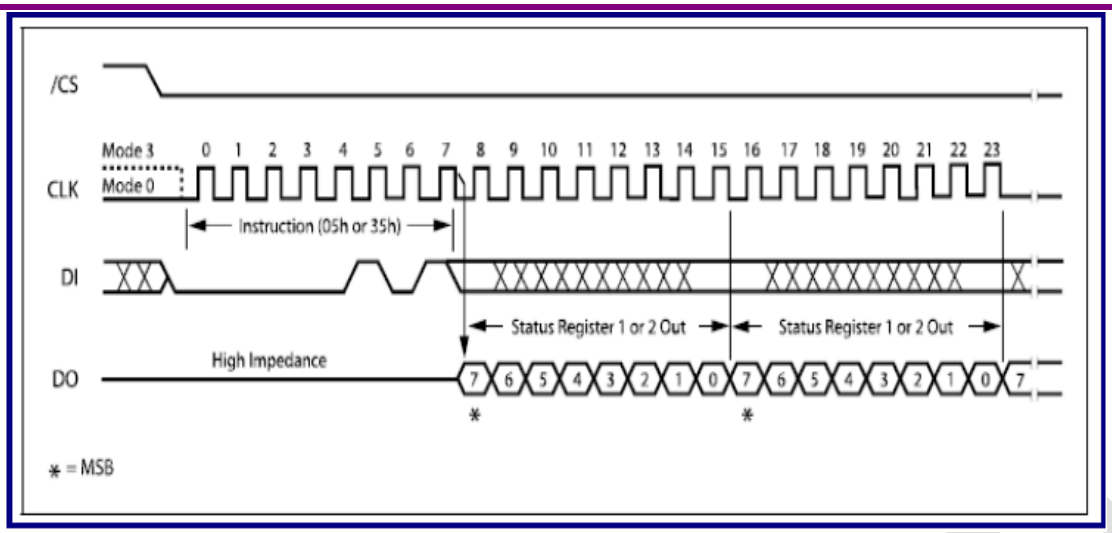


Figure 8-8. Read Status

If users want to execute the read data command, please set the instruction code/length to 1 byte, address/address length to 3 bytes, and write enable to '0'. Please refer to Figure 8-9 for more details.

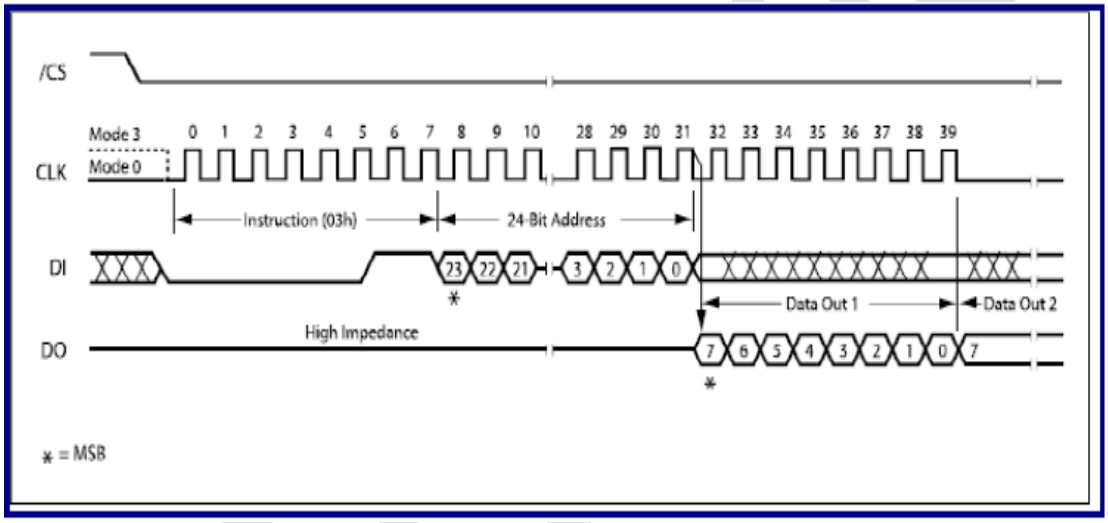


Figure 8-9. Read Data

B: Dual Mode/Dual I/O Mode

If users want to execute the fast read dual output command, please set the instruction code/length to '1', address/address length to 3 bytes, dummy second cycle to 8, and operation mode to dual mode. Please refer to Figure 8-10 for more details.

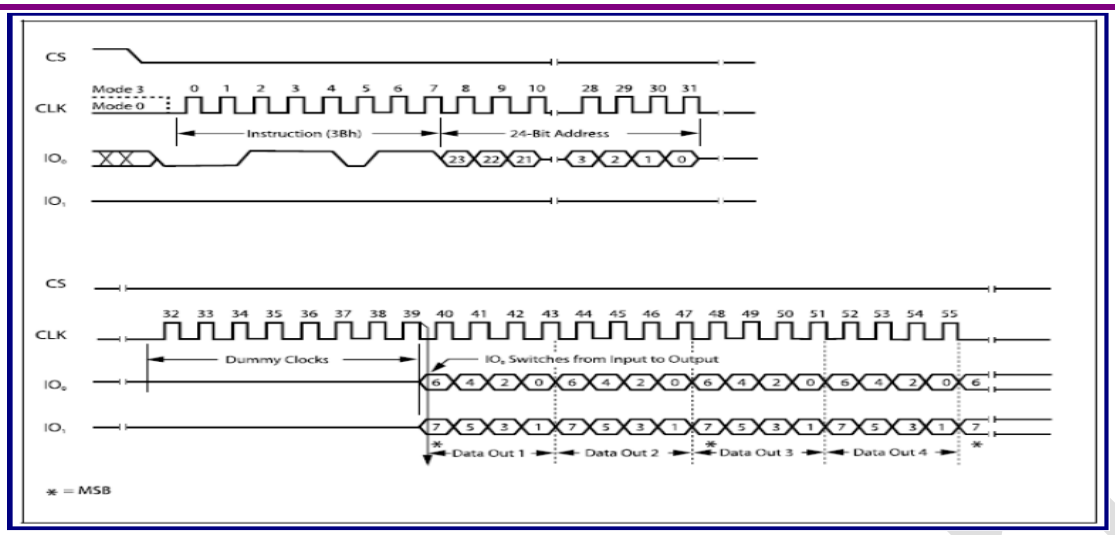


Figure 8-10. Fast Read Dual Output Command

If users want to execute the fast read Dual I/O command, please set the instruction of code/length, address/address length, continuous read mode to enable, continuous read mode code, and operation to dual_io mode. Please refer to Figure 8-11 for more details.

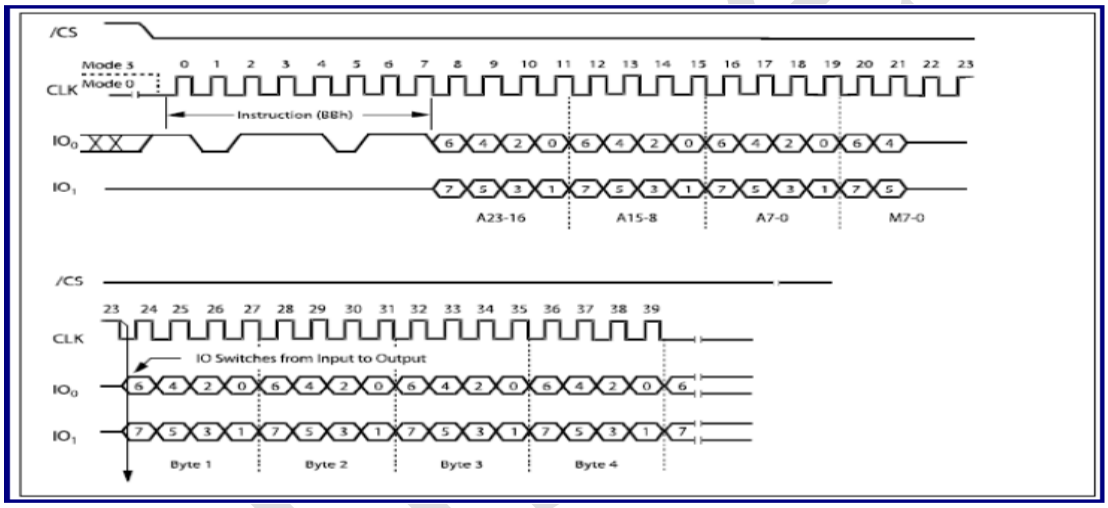


Figure 8-11. Fast Read Dual I/O

C: Quad Mode/Quad I/O Mode

If users want to execute the fast read Quad I/O command, please set the instruction code/length, address/address length, dummy second cycle, and operation mode to the quad mode. Please refer to Figure 8-12 for more details.

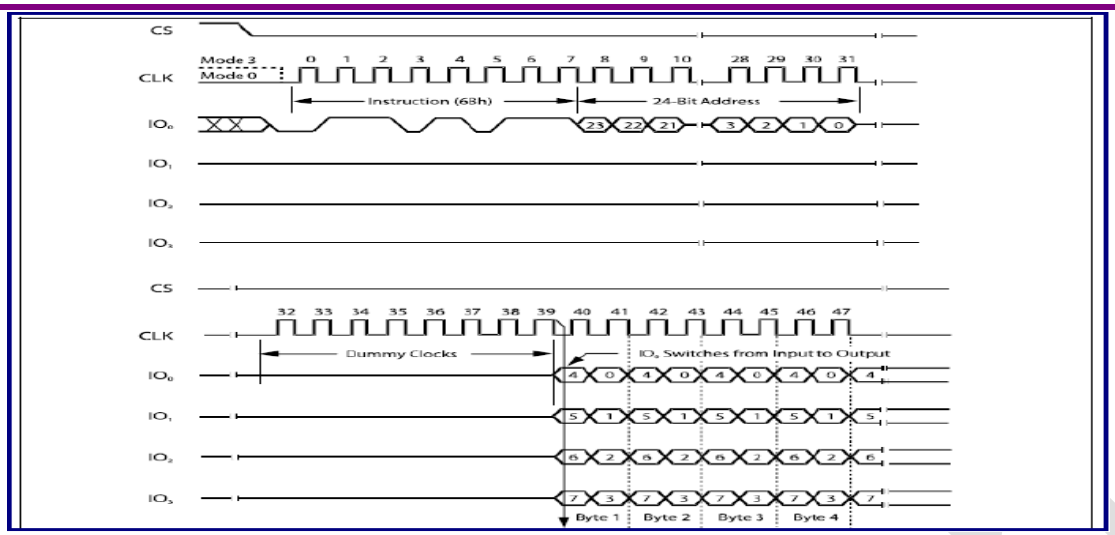


Figure 8-12. Fast Read Quad Output

If users want to execute the fast read Quad I/O command, please set the instruction code/length, address/address length, 4-cycle dummy second, continuous read mode/code, and operation mode to the Quad I/O mode. Please refer to Figure 8-13 for more details.

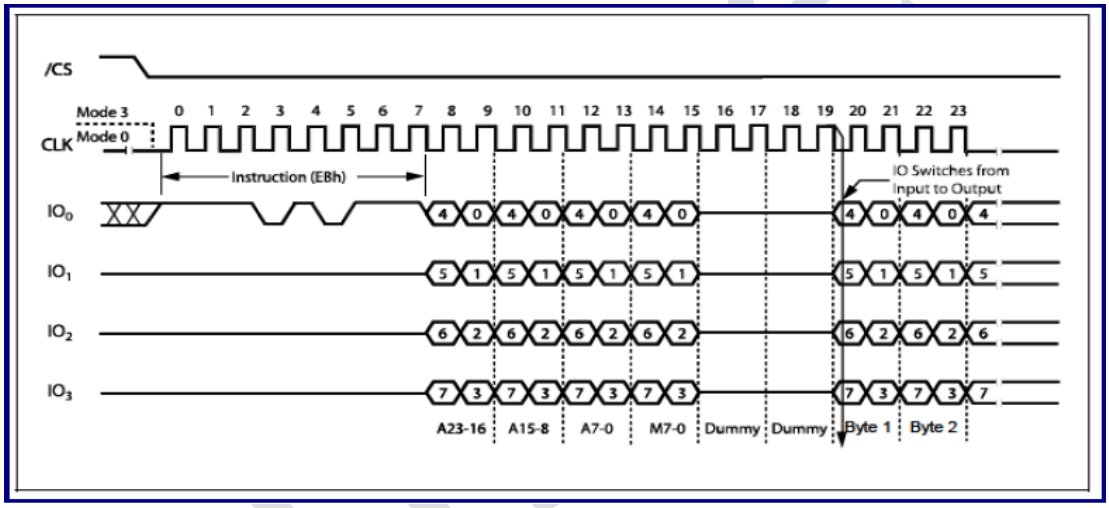


Figure 8-13. Fast Read Quad I/O Command

Note: If entering the continuous read mode, command can only be accepted by the continuous read command, except the reset continuous read mode.

If users want to reset the continuous read mode, please set the instruction length to '1' for resetting to the Quad I/O mode, or set instruction length to '2' for resetting to the Dual I/O mode. Please refer to Figure 8-14 for more details.

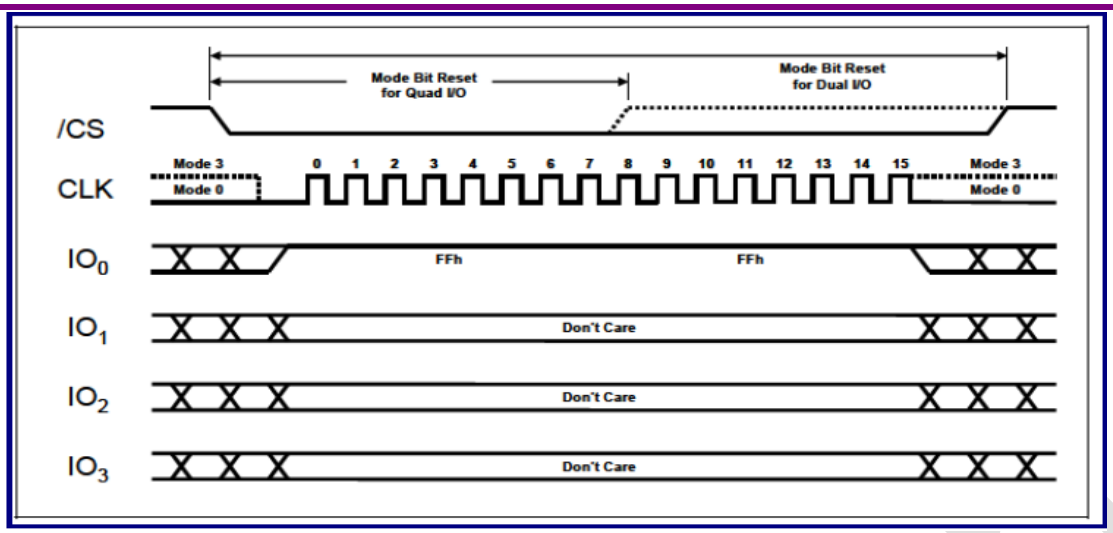


Figure 8-14. Continuous Read Mode Reset

D: DTR Mode

FTSPI020 supports the Double Transfer Rate (DTR) mode and latchs data at the rising edge and falling edge. Users must fill the DTR mode (Offset = 0x0C) and other corresponding registers to excute the SPI command.

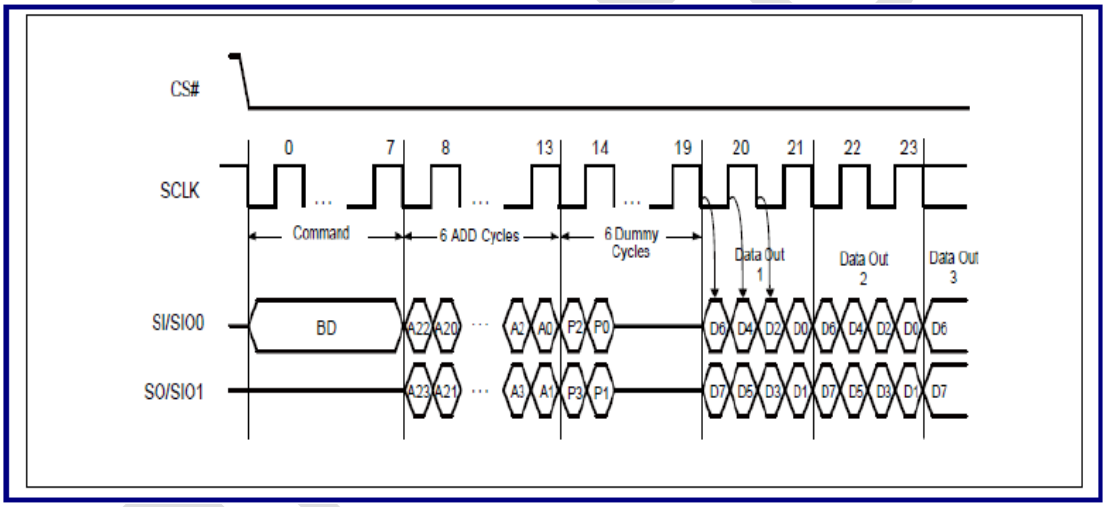


Figure 8-15. Fast Dual I/O DT Read (2DTRD) Sequence (Command BD)

E: AC Timing Relationship

1. The CS active setup time relative to SCK (sck_out) is shown in Figure 8-16 . tCSS is spi_clk period period) * divider (spi_clk_divider from Control Register (CR, offset = 0x0010)) 0x0010)). For example, when users apply the spi_clk period = 10 ns , and the divider is set to 4 4; once CS# falls, after 40 ns, SCK be gins to be activeactive.

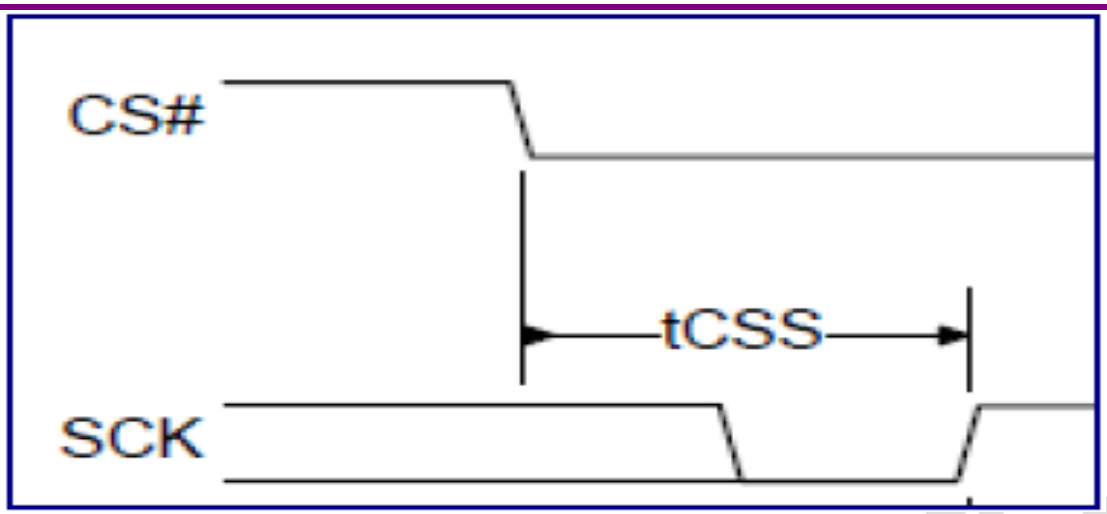


Figure 8-16. CS Active Setup Time relative to SCK

2. For READ and CS# (rise) then SCK (off): It refers to the timing parameters, tCSS and tCSH, in the green rectangle (2) shown in Figure 8-17. However, FTSPI020 will directly apply SCK to high, as the red line (3) shown in Figure 8-17. Since there is no SCK transition, Flash does not check tCSS and tCSH at all, which means that users can ignore the timing parameters here. In other words, those timing parameters are "don't care".
   As for tCSH in the orange rectangle (1), FTSPI020 will apply CS# to high after the SCK rising edge,in a half of the minimum SCK period. For example, the SCLK clock period is 10 ns , and then the fastest CS# will go to high after the SCK rising edge is 5 ns.

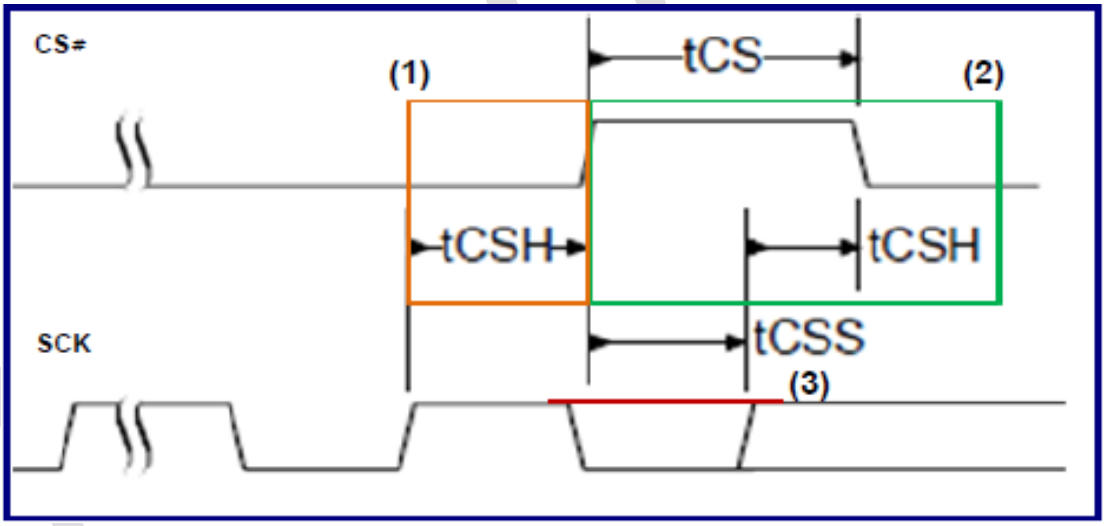


Figure 8-17. CS Active Setup Time relative to SCK

3. For WRITE and CS#(fall) to SCK (on): the same as item 1.
4. For WRITE and CS#(rise) then SCK (off): the same as item 2. However, tCSH in the orange rectangle (1) is one minimum SCK period.

8.3.5. Initialization/Application Information

In a series of transfer transfers, user users can not change the transfer size while read ing /writ ing a data.

8.3.5.1. Data Transfer in PIO Mode at Command-based Slave Port

1. Set the FTSPI020 XIP_port_en control register to '0'
2. Poll the 0x10 abort control register to confirm whether the port switching is done

3. Set one command queue to meet the SPI operation flow and enable the command complete interrupt enable
4. Poll txfifo_ready/rxfifo_ready to write/read data to/from the data port[3]
   - RX:
     1. Polling RX FIFO ready, when it is 1'b1,then go to Step 2a or 2b,respectively (depending on the amount of data).
     2. a. If the amount of data at this transaction is larger than the FIFO size, then this "ready" means that FIFO is full. Just fetch the amount of data that equals to the FIFO size,then go back to Step 1 to poll the ready again. Users do not have to poll the ready status during data fetching.

        b. If the amount of data at this transaction is less than the FIFO size, then this "ready"means that the userdata is ready in FIFO and it isthe last data transaction. Read the user demanding amount and then go to Step 3.
     3. The flow is complete.
   - TX:
     1. Polling RX FIFO ready, when it is 1'b1,then go to Step 2a or 2b,respectively (dependingon the amount of data).
     2. a. If the amount of data at this transaction is larger than the FIFO size, then this "ready" means that FIFO is empty. Just move in the amount of data that equals to the FIFO size, then go back to step 1 to poll the ready again. Users do not have to poll the ready status during data moving.

        b. If the amount of data at this transaction is less than the FIFO size, then this "ready" means that it is the last data transaction. Move the user demanding amount data and then go to Step 3.
     3. The flow is complete.
5. Wait for the command to complete interrupt and to clear the interrupt

[3] For accessing the data port, the size of a transfer must keep the same. For example, when using the AHB host IF , the first transfer HSize is BYTE, and HSize of the next transfers must be kept to BYTE.

8.3.5.2. Read Data Transfer at XIP Slave Port (Port Switching Sequence)

1. Set the XIP port to issuethe quad/dual/address 4-byte mode by 0x30. If the current operation is on the command slave port and users desire to read data from the XIP slave port in Quad/Dual/Address 4-byte mode, please make sure that SPI Flash has already turned on the Quad/Dual/Address 4-byte mode, or it has ability to execute quad/dual/address 4-byte mode transfer first (Optional).
2. Make sure the operation on the current port is finished
3. Poll the bit 8 as 0 and bit 7 as 1 of 0x10 register to confirm whether the controller is ready for being switched
4. Set the 0x10 XIP_port_sel control register to'1'
5. Poll the 0x10 abort control register to confirm whether the port switching is finished
6. Issue the bus read command to read data from SPI Flash
7. After finishing all read operations, users can switch to the command slave port by doing steps from 3 to 6 (Set the 0x10 XIP_port_sel controller register to '0') before using the command slave port

Note: Steps 3 ~ 6 are the port switching sequence.

8.3.5.3. Recommended Boot Sequence

1. Port is switched to XIP as default if the hardware configuration, FTSPI020_XIP_PORT_ON , is defined.
2. CPU fetches the boot code from SPI Flash and initializes DDRC.
3. CPU copies the initial code to the main memory memory.
4. CPU jumps to the main memory and executes the initial code (If it is necessary to do the port switching operation, the port switching software/firmware code must be copied to the main memory before going to Step 5) .
5. CPU configures the SPI Flash to enter the Quad/Dual mode through the command slave port registersregisters.
   - Poll the bit 8 as 0 and bit 7 as 1 of 0x10 r egister to confirm whether the controller is ready for being switched
   - Switch port ports from the XIP port to the command slave port by setting '0' to 0x10 bit[20]
   - Poll the bit 8 as 0 and bit 7 as 1 of 0x10 register to confirm whether the port switching is finished
   - Configure SPI Flash to enter the Quad/Dual mode by setting 0x30 registers
6. Switch port ports from the command slave port to the XIP port by setting 0x10 bit[20] (Optional)
   - Set the XIP port to access the Quad/Dual mode by 0x30 if user users want to do Quad/Dual by the XIP port
   - Poll the bit 8 as 0 and bit 7 as 1 of 0x10 register to confirm whether the controller is ready for being switched
   - Switch port ports from the XIP port to the command slave port by setting '1' to 0x10 bit[20]
   - Poll the bit 8 as 0 and bit 7 as 1 of 0x10 register to confirm whether the port switching is finished.
7. Move the main code from the SPI Flash to the main memory by PIO memory copy (Optional)
8. CPU jumps to the main code.

8.3.5.4. Other Operations

1. Set one command queue to meet the SPI operation flow and enable the command complete interrupt enable
2. Wait for the command to complete interrupt and to clear the interrupt
3. When FTSPI020_ XIP _PORT_ON is configured , the command slave port and XIP port will be selected by the XIP_port_sel at bit 20 of 0x10 register . Meanwhile, the operation rules are listed below:
   1. After changing the XIP_port_sel control register, users must wait until the abort function is finished.
   2. When selecting the command slave port, users can not issue any commands to the XIP port; when selecting the XIP port, users can only issue the read commands to registers (Except the 0x10 register) and the data port (0x100) can not be accessed.
   3. Software must make sure that a transaction has done before port switching.

8.3.6. Register List

Module Name	Base Address
SPIF	0x1200_0000

Register Name	Offset	Description

Register Name	Offset	Description
Cmd_w0	0x0000	Command Queue Word 0
Cmd_w1	0x0004	Command Queue Word 1
Cmd_w2	0x0008	Command Queue Word 2
Cmd_w3	0x000C	Command Queue Word 3
CR	0x0010	Control Register
ACTR	0x0014	AC Timing Register
SR	0x0018	Status Register
ICR	0x0020	Interrupt Control Register
ISR	0x0024	Interrupt Status Register
SPISR	0x0028	SPI Read Status Register
SPIFSR	0x002C	SPI Flash Size Register
XIPCMD	0x0030	SPI XIP Command Word
REVISION	0x0050	Revision Register
FEATURE	0x0054	Feature Register
DR	0x0100	Data Port Register

8.3.7. Register Description

The following subsections provide the detailed descriptions of the control registers.

8.3.7.1. Command Word 0 ~ 3

One command contains four words. One command will be executed by writing the fourth word of a command and will be completely executed by the host SPI controller.

Notes:

1. The command must be executed before reading/writing the data into FIFO.
2. The reserved region of the command queue is not allowed to be filled to '1'.

- For the write command, users must check and clear the command complete interrupt before writing the next command.
- For the read status command, users must check and clear the command complete interrupt before writing the next command.
- For the read data command, users must finish reading the data from RXFIFO; and check and clear the command complete interrupt before writing the next command.

Table 8-82. Command Queue First Word (Cmd_w0, Offset = 0x0000)

Bit	Name	Type	Description	Default Value
[31:0]	SPI Flash Address	RW	SPI Flash address This register decides the values of the SPI Flash address and issues this address to the SPI Flash. The byte of the address is executed by the address length when offset = 0x04.	0

vangotech

Table 8-83. Command Queue Second Word (Cmd_w1, Offset = 0x0004)

Bit	Name	Type	Description	Default Value
[31:29]	-	Rsvd	Reserved	0
28	Continuous Read Mode Enable	RW	The continuous read mode state is located between the address and second dummy state. It can eliminate the instruction state after the second read command in the continuous read mode. Please do not enable this bit while data counter (Cmd_w2) is 0. 1'b0: Disable the continuous read mode 1'b1: Enable the 1-byte continuous read mode	0
[27:26]	-	Rsvd	Reserved	0
[25:24]	Instruction length	RW	Instruction code length When users want to execute a SPI Flash command, the instruction code must be included. Different SPI Flash vendors have different instruction lengths; therefore, users can set this register to meet different behaviors. Instruction code is normally 1 byte; however, if users set the 2-byte instruction length, the host controller will issue this instruction code twice. 2'b00: No instruction code. It can only be used after the contiunous read mode command has been finished. 2'b01: 1-byte instruction code 2'b10: 2-byte instruction code (Repeat the instruction code) 2'b11: Reserved	0x1
[23:16]	dum_2nd_cyc	RW	Second dummy state cycle Second dummy state is located between the address and the data state that excludes the continuous read mode state. Users can check whether the dummy state exists between the address and the data state or not in the SPI Flash specification. The host controller will issue logic 1 in the dummy cycle. 8'd0: No second dummy state 8'd1 ~ 8'd32: 1 dummy second cycle ~ 32	0

Vangotech

Bit	Name	Type	Description	Default Value
			dummy second cycles	
[15:3]	-	Rsvd	Reserved	0
[2:0]	address length	RW	This register decides the SPI Flash address byte number. Users can set this register to decide the address byte ranging from 1 byte to 4 bytes. 3'b000: No address state 3'b001: 1-byte address 3'b010: 2-byte address 3'b011: 3-byte address 3'b100: 4-byte address Others: Reserved For the XIP port access, the value is decided by the SPI_addr_cyc register (0x2800_810C). When the address length is 0, there will be no dummy cycle ([23:16]) and continuous read mode code (0x000C[2:16]) will occur.	0x3

Table 8-84. Command Queue Third Word (Cmd_w2, Offset = 0x0008)

Bit	Name	Type	Description	Default Value
[31:0]	Data counter	RW	Read/Write data counter This register must be set to '0' when performing the read status command. 32'b0: No read/write data 0x1 ~ 0xFFFF_FFFF: 1-byte data ~ 0xFFFF_FFFF data Please note that no matter it is a data read or data write, this register is not allowed to be filled as '0'. However, such as "read status" or "write enable" instruction, this register must set to '0'.	0

Table 8-85. Command Queue Fourth Word (Cmd_w3, Offset = 0x000C)

Bit	Name	Type	Description	Default Value

Vangotech

Bit	Name	Type	Description	Default Value
[31:24]	Instruction Code	RW	Instruction code Users can set this code to execute the SPI Flash command.	0x3
[23:16]	Continuous Read Mode Code	RW	Continuous read mode operate code Users can fill this code to execute the continuous read mode. To do the continuous read mode with this Flash, users have to issue 0xA0 as the mode bit because it does not need the instruction code for the next read command.	0xB0
[15:10]	-	Rsvd	Reserved	0
[9:8]	Start cs	RW	FTSPI020 can connect four SPI Flash (Max.) and this bit is used to select cs. 2'b00: cs 0 2'b01: cs 1 2'b10: cs 2 2'b11: cs 3	0
[7:5]	SPI Operate Mode	RW	SPI operate mode 3'b000: Serial mode 3'b001: Dual mode 3'b010: Quad mode 3'b011: dual_io mode 3'b100: quad_io mode Others: Reserved	0x0
4	DTR Mode	RW	DTR (Double Transfer Rate) mode 1'b0: Disable The read status, write data command, and DTR mode must be set to '0'. 1'b1: Enable The DTR mode is valid at the read data command. The DTR mode will be invalid when hardware configuration is "FTSPI020_DTR_MODE_OFF". The DTR mode does not support the write operation and the DTR_mode register must be set to"1";	0

Vangotech

Bit	Name	Type	Description	Default Value
			otherwise, the sequence will fail.	
3	Read Status	RW	Read the SPI Flash status by using software or hardware It is only available when the read status is enabled and the write enable = 1'b0. Users must issue the SPI Read Status command. 1'b0: Read status by hardware, the controller will poll the status until the status is ready (Not busy) and report the status register. 1'b1: Read status by software, read status once and report the status register until users can read it.	0
2	Read Status Enable	RW	Enable the Read SPI status It is available at write_enable = '0' and users must issue the SPI read status command. 1'b0: Disable 1'b1: Enable	0
1	Write enable	RW	Enable the SPI write data, except for the read data or read status (Read the data return path); users must set the write enable = '1' for other SPI commands. Please note that in the write data or erase Flash command, write enable is set to '1'. Only when in the read data or read status command, write enable must be set to '0'.	0
0	-	Rsvd	Reserved	0

8.3.7.2. Control Register

This register must be set before executing the command, and it is prohibited to set this register when performing the command. Reserved region is not allowed to be filled to '1'.

Table 8-86. Control Register (CR, Offset = 0x0010)

Bit	Name	Type	Description	Default Value

Vangotech

Bit	Name	Type	Description	Default Value
[31:21]	-	Rsvd	Reserved	0
20	XIP_port_sel	RW	XIP port selection Read SPI Flash data from the XIP port 0: Command slave port 1: XIP port When the bit is toggled, FTSPI020 will automatically issue the abort function. Users must wait until the abort function is finished to issue a command.	1
19	-	Rsvd	Reserved	0
[18:16]	rdy_loc	RW	Busy bit of the SPI status Host polls this busy bit and is ready at the HW read status. 3'b000 ~ 3'b111: Bit 0 ~ bit 7	0
[15:9]	-	Rsvd	Reserved	0
8	abort	RW	Flush all commands/FIFOs and reset the state machine When abort occurs, users must fill commands again. This bit will be automatically cleared to zero. It is strongly suggested that there is no need to set this bit when setting bit 20 (Change port).	0
7	XIP_port_idle	RO	XIP port idle status 0: XIP port is busy. 1: XIP port is idle.	0
[6:5]	-	Rsvd	Reserved	0
4	spi_clk_mode	RW	spi clk mode at the IDLE state 0: For mode0, sck_out will be low at the IDLE state. 1: For mode3, sck_out will be high at the IDLE state.	0
[3:2]	-	Rsvd	Reserved	0
[1:0]	spi_clk_divider	RW	spi_clk divider sck_out is divided by spi_clk and factor is listed as follows: 2'b00: Divided by 2 2'b01: Divided by 4 2'b10: Divided by 6 2'b11: Divided by 8	0x3

Vangotech

Bit	Name	Type	Description	Default Value
			Note: When programming the DTR mode, only the divider by 4/8 is allowed.	

8.3.7.3. AC Timing Register

Table 8-87. AC Timing Register (ACTR, Offset = 0x0014)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:4]	Trace_delay	RW	Trace delay for turn around stage 4'b0000 ~ 4'b1111: 1cycle ~ 16 cycles	0x3
[3:0]	cs_delay	RW	cs delay is the time from inactive cs to active cs, which means that the timing is from high to low. Please follow the specification for details. The unit is the sck_out period, and the default setting is 16 sck_out periods. 4'b0000 ~ 4'b1111: 1 cycle ~ 16 cycles	0xF

8.3.7.4. Status Register

Table 8-88. Status Register (SR, Offset = 0x0018)

Bit	Name	Type	Description	Default Value
[31:2]	-	Rsvd	Reserved	0
1	RXFIFO Ready	RO	RxFIFO ready status When RxFIFO is ready, it indicates that RxFIFO is full, and the remained data in RxFIFO will be less than the RXFIFO depth and this is the last data.	0
0	TXFIFO Ready	RO	TxFIFO ready status When TxFIFO is ready, it indicates that TxFIFO will be empty, and users can transfer the data into TxFIFO until it is full.	1

8.3.7.5. Interrupt Control Register

Table 8-89. Interrupt Control Register (ICR, Offset = 0x0020)

Bit	Name	Type	Description	Default Value
[31:2]	-	Rsvd	Reserved	0
1	cmd_cmplt_intr_en	RW	Command complete interrupt enable 1'b0: No interrupt 1'b1: Enable command complete interrupt	0

Vangotech

Bit	Name	Type	Description	Default Value
0	-	-	-	0

8.3.7.6. Interrupt Status Register

Table 8-90. Interrupt Status Register (ISR, Offset = 0x0024)

Bit	Name	Type	Description	Default Value
[31:1]	-	Rsvd	Reserved	0
0	cmd_cmplt_sts	RW1C	Command complete status will be set when the command is complete.	0

8.3.7.7. SPI Read Status Register

Table 8-91. SPI Read Status Register (SPISR, Offset = 0x0028)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	SPI Read Status	RO	Host issues the read SPI Flash status command and stores the return data at this register. Users can read this register to check the SPI Flash status.	0

8.3.7.8. SPI Flash Size Register

This register will exist when the hardware configuration configuration, FTSPI020_ XIP _PORT PORT, is set to ON.

Table 8-92. SPI Flash Size Register (SPIFSR, Offset = 0x002C)

Bit	Name	Type	Description	Default Value
[31:0]	SPI Flash Size	RW	For the direct address mapped function, the address of system is always larger than the address of the SPI Flash. Users need to mask the higher bits of the system address to fit the SPI Flash size.	32'hFF000000

8.3.7.9. XIP Command Word

This register will exist when the hardware configuration configuration, FTSPI020_ XIP _PORT PORT, is set to ON.

Table 8-93. XIP Command Word (XIPCMD, Offset = 0x0030)

Bit	Name	Type	Description	Default Value

Vangotech

31	-	RW	Reserved	0
[30:29]	Start cs	RW	To select which SPI Flash chip to read 2'b00: cs 0 2'b01: cs 1 2'b10: cs 2 2'b11: cs 3	0
28	Quad/Dual I/O Mode Enable for XIP port	RW	Quad/Dual I/O mode allow the address to transmit in 2-bit or 4-bit, depending on the data sheet of the user devices. 1'b0: Disable the Quad/Dual I/O mode 1'b1: Enable the Quad/Dual I/O mode	0
[27:20]	Quad/Dual I/O Mode Code for XIP port	RW	Quad/Dual I/O mode code for the XIP port Users have to issue the code, for example, 0xB0, as the mode bit because every command needs instruction code while the XIP port is reading. The code depends on the data sheet of the user Flash devices.	0xB0
[19:12]	Insruction Code for XIP port	RW	Instruction code Users can set this code to execute the SPI Flash command for XIP read. If this byte set to 0, XIP will transfer in the 1-bit mode.	0
11	SPI Address Length for XIP port	RW	This register decides the SPI Flash address byte number. Users can set this register to decide the address 3 bytes or 4 bytes for the XIP port. 1'b0: 3-byte address 1'b1: 4-byte address	0
[10:8]	SPI Operate Mode for XIP port	RW	SPI operate mode for the XIP port 3b000: Serial mode 3b001: Dual mode 3b010: Quad mode 3b011: dual_io mode 3'b100: quad_io mode Others: Reserved	0
[7:0]	dum_2nd_cyc for XIP port	RW	Second dummy state cycle for the XIP port Second dummy state is located between the address and the data state that excludes the continuous read mode state. Users can check whether the dummy state exists between the address and the data state or not in the SPI Flash specification. The host controller will issue logic 1 in the dummy cycle. 8'd0: No second dummy state	0

Vangotech

			8'd1 ~ 8'd32: 1 dummy second cycle ~ 32 dummy second cycles	

8.3.7.10. XIP Prefetch Function Option

This register will exist when the hardware configuration configurations, FTSPI020_XIP_PORT _ON, FTSPI020_PFC_ON, and FTSPI020_IF_AHB are configured.

Table 8-94. XIP Prefetch Function Option (XIPPFO, Offset = 0x0034)

Bit	Name	Type	Description	Default Value
[2:1]	SRAM_RD_LATENCY	RW	SRAM read latency setting The unit is cycle number. The read access time of SRAM (Taa) depends on the SRAM device character and Taa unit is ns. Besides, users need to translate the Taa time to the cycle number. For example, Taa = 15 ns and the clock period of AHB is 6 ns; then, the field set to Roundup(15/6) – 1 = 2	1
0	bypass_cache	RW	Bypass cache function When setting the bit, FTSPI020 will do invalidate cache function and all read transfers will not lookup cache.	0

8.3.7.11. Revision Register

Table 8-95. Revision Register (REVISION, Offset = 0x0050)

Bit	Name	Type	Description	Default Value
[31:0]	Revision_Number	RO	It indicates this IP version.	0x0001_0600

8.3.7.12. Feature Register

Table 8-96. Feature Register (FEATURE, Offset = 0x0054)

Bit	Name	Type	Description	Default Value
31	HOST_IF	RO	FTSPI020 host interface 0: AHB I/Fs 1: AXI I/Fs	0
30	HOST_IF_DW	RO	FTSPI020 host inteface data width 0: Data width is 32 bits. 1: Data width is 64 bits.	0
29	XIP_PORT	RO	FTSPI020 XIP port	1

Bit	Name	Type	Description	Default Value
			0: Only command-based slave port 1: Command-based slave port and XIP ports	
[28:26]	-	Rsvd	Reserved	0
25	CLK_MODE	RO	FTSPI020 clock mode 0: ASYN mode 1: SYNC mode	0
24	DTR_MODE	RO	FTSPI020 DTR mode 0: Disable 1: Enable	1
[23:19]	AXI_ID_DW	RO	AXI ID width 0: No ID port 1 ~ 19: ID width	8
[18:17]	-	Rsvd	Reserved	0
[15:8]	RXFIFO_DEPTH	RO	FTSPI020 RXFIFO depth	0x1f
[7:0]	TXFIFO_DEPTH	RO	FTSPI020 TXFIFO depth	0x1f

8.3.7.13. Data Port Register

Table 8-97. Data Port Register (DR, Offset = 0x0100)

Bit	Name	Type	Description	Default Value
[31:0]	Data Port	RW	Data port register Users can read/write data from the data port.	0

8.4. SPI2AHB

8.4.1. Overview

The SPI2AHB is a SPI slave to receive read/write command from SPI host and issue read/write transfer at AHB Lite bus.

8.4.2. Features

The embedded Flash controller features are listed as follows：

- AMBA AHB Lite protocol.
- Only support AHB single word access.
- The minimum clock radio between SPI clock and system clock sys_clk is 4.

8.4.3. Functional Block Diagram

The block diagram of SPI2AHB is shown Figure 8-18.

Vangotech

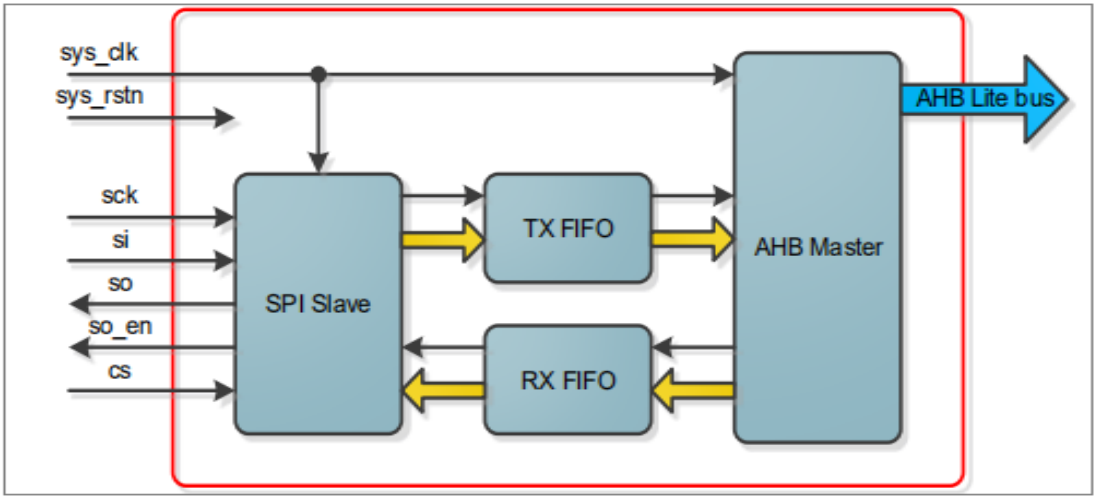


Figure 8-18. Block Diagram

8.4.4. Function Description

8.4.4.1. SPI Timing

The SPI read/write timing is shown in the **Figure 8-19**.

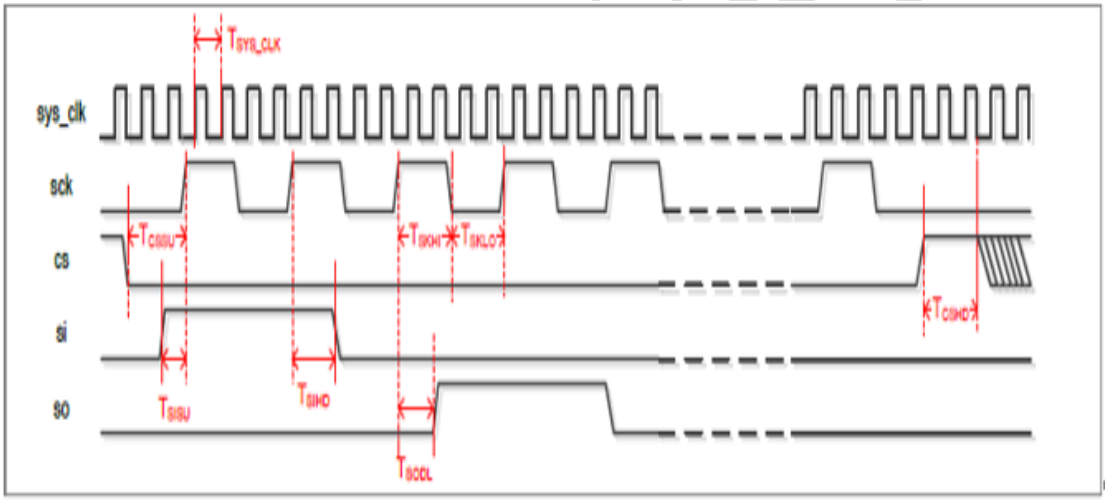


Figure 8-19. SPI Timing Chart

Table 8-98. SPI Bus Specification

Item	Symbol	Min	Typ	Max	Unit
System clock Frequency	F_{SYS_CLK}	-	-	-	-
System clock period	T_{SYS_CLK}	-	1/ F_{SYS_CLK}	-	-
SPI Clock Frequency	F_{SPI}	-	-	1/4	F_{SYS_CLK}
SCK Low Pulse	T_{SKLO}	2	-	-	T_{SYS_CLK}
SCK High Pulse	T_{SKHI}	2	-	-	T_{SYS_CLK}
SDI Setup Time	T_{SISU}	0.5	-	-	T_{SYS_CLK}
SDI Hold Time	T_{SIHD}	0.5	-	-	T_{SYS_CLK}
SDO Output Delay	T_{SODL}	1	-	2	T_{SYS_CLK}
CSB Setup Time	T_{CSSU}	2	-	-	T_{SYS_CLK}
CSB Hold Time	T_{CSHD}	2	-	-	T_{SYS_CLK}

8.4.4.2. SPI Write Protocol

8.4.4.3. SPI Single Write

An example of the single write is 32-bit address and 32-bit data for SPI write transfer is shown as the **Figure 8-20**. The SPI write includes one read/write bit, 32-bits address and 32-bits write data on SI. The most significant bit transfers first. High level of read/write bit indicates a write transaction.

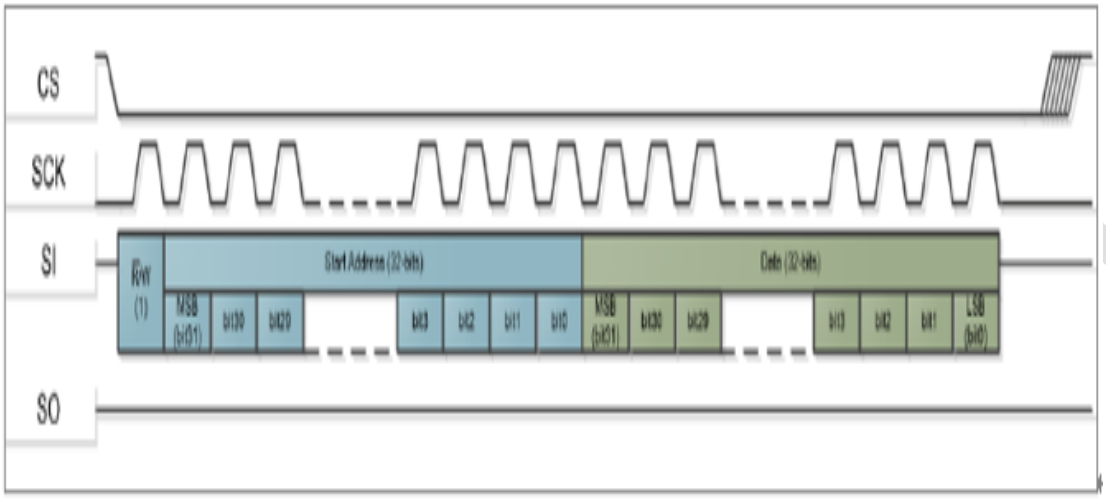


Figure 8-20. SPI Single Write Transfer

8.4.4.4. SPI Burst Write

This design supports write burst mode. Each command starts at CS falling edge and end of riding edge, and first word is address/command, and the consequent data is issued after address. For internal AHB Lite bus, the write address will be increased by 4 when each write data is transferred in the burst mode. An example of write burst 4 is shown in the **Figure 8-21**. For burst 8 and 16, extends write data to 8*32 and 16*32 bits.

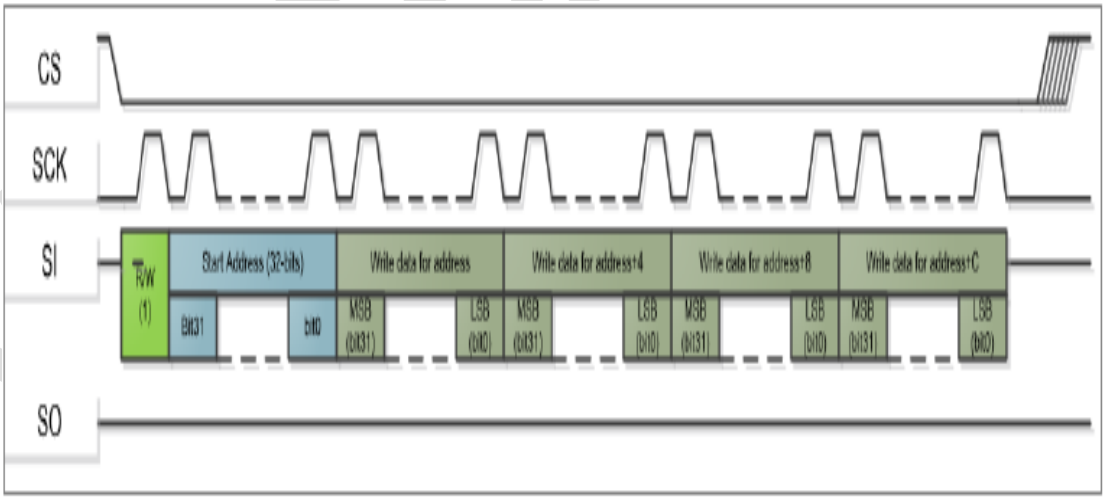


Figure 8-21. SPI Burst Write Transfer

8.4.5. SPI Read Protocol

For SPI protocol, it doesn't allow to insert any wait cycle, so the AHB Lite bus read should be ready within 1 SPI clock, if the AHB clock, sys_clk, is 4 times SPI clock, that's meaning, the AHB transfer should be finish within 4 clock cycles, hready only can low less 4 AHB clock cycles, otherwise, the read data will be incorrect values. Please note that address bit1 and bit0 must be set to 0 for AHB read/write word access.

8.4.5.1. SPI Single Read

An example is shown the SPI read protocol of signal transfer in the Figure 8-22. The SPI read includes one read/write bit, and 32-bits address on X_SPI_DI and 32-bits read data on X_SPI_DO. The most significant bit transfers first. Low level of read/write bit indicates a read transaction.

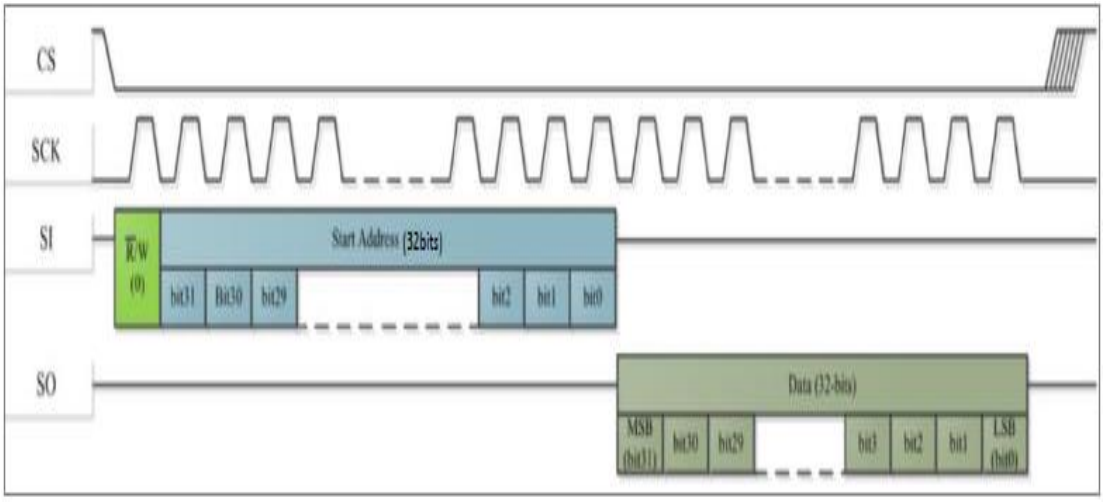


Figure 8-22. SPI Single Read Transfer

8.4.5.2. SPI Burst Read

For SPI burst read, this design supports any read burst. The following diagram is an example of read burst 4. For burst 8 and 16, extends read data to 8*32 and 16*32 bits.

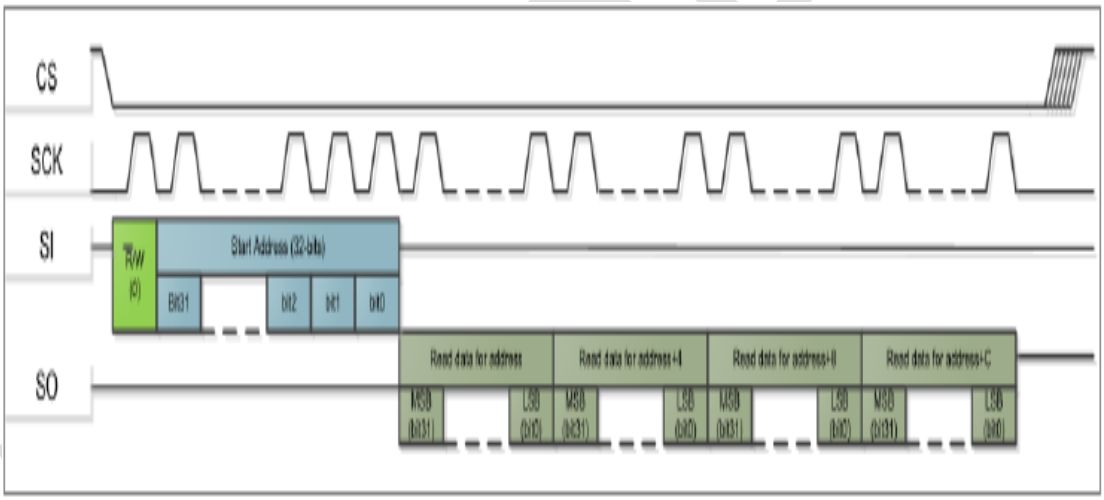


Figure 8-23. SPI Burst Read Transfer

8.5. I2C

8.5.1. Overview

The I2C bus interface controller, FTIIC010, allows the host processor to serve as a master or slave in the I2C bus. Data are transmitted to and received from the I2C bus via a buffered interface.

8.5.2. Features

- Compliant with AMBA 2.0 APB
- Supports Standard, Fast, and Fast+ modes (These 3 modes are so-called F/S-mode.) by programmable clock division register
- Supports HS-mode by programmable clock division register
- Supports 7/10-bit addressing modes
- Supports General-Call and START byte
- Supports glitch suppression
- Supports programmable slave address
- Supports master-TX, master-RX, slave-TX, and slave-RX modes
- Supports multi-master mode

8.5.3. Block Diagram

Figure 8-24 shows the block diagram of the I2C bus interface controller.

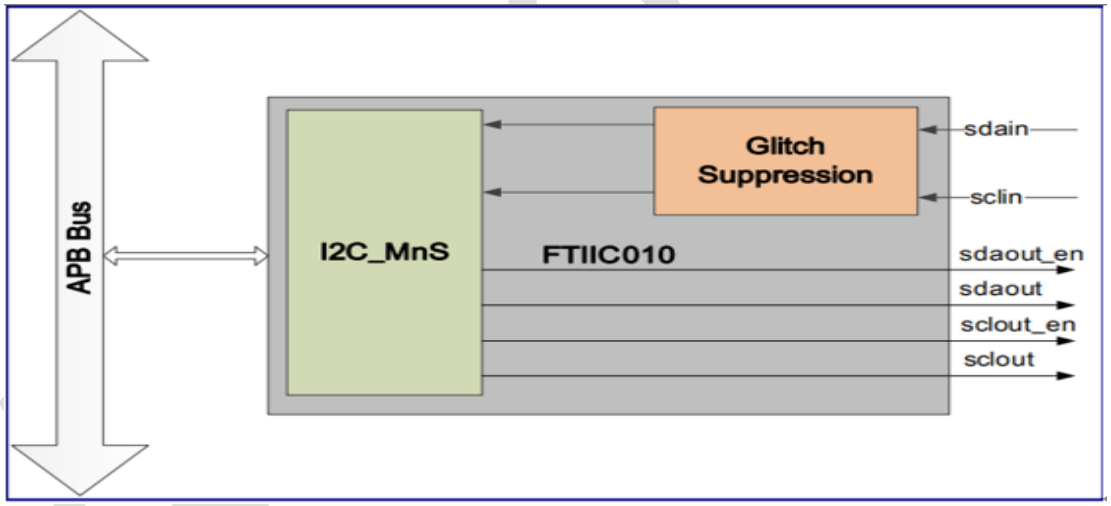


Figure 8-24. Functional Block Diagram

The main building blocks of the I2C bus interface controller include the I2C_MnS control block and Glitch Suppression circuit. Please refer to the following subsections for more details of each block.

8.5.3.1. Glitch Suppression Circuit

Glitch Suppression circuit is used to eliminate the glitch on the I2C bus. Glitches are suppressed according to the GSR internal bus clock period, where GSR is bit[13:10] of the register TGSR at offset 0x14.

8.5.3.2. I2C_MnS Circuit

The I2C_MnS circuit is used to detect sclin and sdain from glitch suppression circuit, generating sclout,sclout_en, and sdaout. This block also contains the register files and decides the status of the bus.

8.5.4. Software Programming Guide

8.5.4.1. Programming Guide for TX/RX Slave Mode

1. Write AR: Set the slave address
2. Write CR: Enable all interrupts and I2C enable, and disable MST_EN
3. Wait for the data receive interrupt - Read SR:
   - If RW (1), TD (1), SAM (1), I2CB (1), and NACK (0), then write 1 to clear the corresponding SR bits, and go to step 4.
   - If RW (0), TD (1), SAM (1), I2CB (1), and NACK (0), then write 1 to clear the corresponding SR bits, and go to step 8.
4. Write DR: Load the data byte to DR for transfer
5. Write CR: Set the TB_EN bit
6. Wait for the data transmit interrupt - Read SR:
   TD (1) or STOP (1), then write 1 to clear the corresponding SR bits.
7. If STOP is set, go to the initial step. If STOP is not set, go to step 4 to load the second data byte.
8. Write CR: Set the TB_EN bit and set NACK if required.
9. Wait for the data receive interrupt - Read SR:
   - If TD (1) and STOP (0): Write 1 to clear the corresponding SR bits, then go to step 9
   - If TD (0) and STOP (1): Write 1 to clear the corresponding SR bits, then go to step 10
10. Read DR: To get data
11. If STOP is set, go to the initial step. If STOP is not set, go to step 8 to get the next data.

Note: If the STOP interrupt occurs, go back to the initial step.

8.5.4.2. Programming Guide for RX in Slave Mode with Repeat-Start

This section is a special case that slave needs a repeat-start (Sr) to switch write to read like the EEPROM behavior in Figure 8-25.

1. Write SAR: Set the slave address
2. Write CR: Enable all interrupts and I2C enable, and disable MST_EN
3. Wait for the data receive interrupt - Read SR:
   TD (1), SAM (1), I2CB (1), and NACK (0), then write 1 to clear the corresponding SR bits
4. Write CR: Set the TB_EN bit and set NACK if required.
5. Wait for the data receive interrupt - Read SR:
   - If TD (1) and START(0): Write 1 to clear corresponding SR bits, then go to step 6
   - If TD (0) and START(1): Write 1 to clear corresponding SR bits, then go to step 7
   - If TD (1), START(1), and SAM (1): Write 1 to clear corresponding SR bits, then go to step 4 listed in Section 8.5.4.1.
   - If TD (1), START(0), and SAM (1): Write 1 to clear corresponding SR bits, then go to step 4 listed in Section 8.5.4.1.
6. Read DR: To get data
7. Go to step 4

vangotech

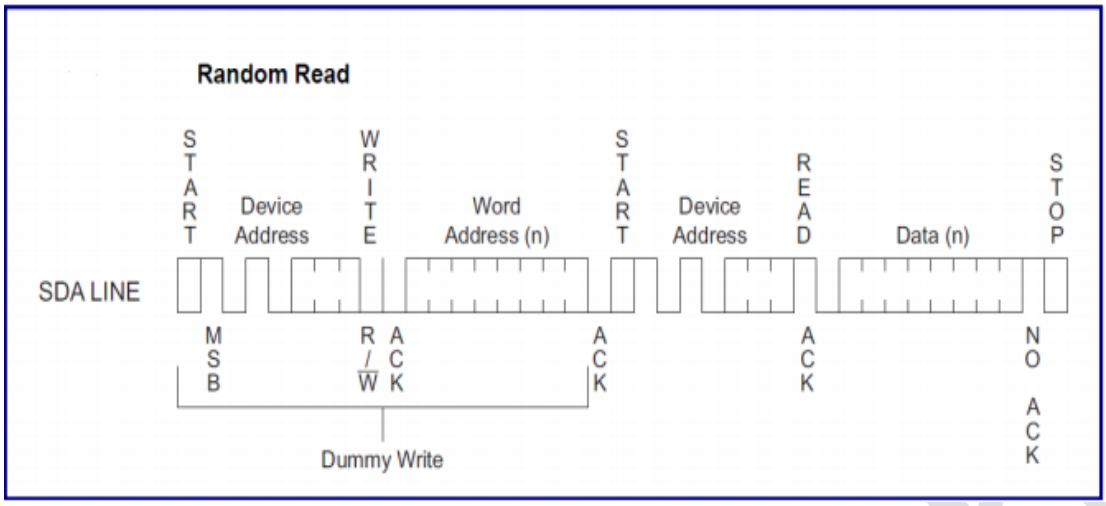


Figure 8-25. Example of Repeat-Start Sequence

Note: If the STOP interrupt occurs, go back to the initial step.

8.5.4.3. Programming Guide for TX in Master Mode

1. Write CDR: Set the clock count
2. Write DR: Target slave address and RW bit for slave
3. Write CR: Enable all interrupts, set MST_EN,set the I2C enable,set the START bit,clear the STOP bit, and set the TB_EN bit
4. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (0), RW (0), and AL (0), then write 1 to clear the corresponding SR bits
5. (a). If the master wants to stop after one byte of data has been transferred:
   - Write DR: First data will be sent.
   - Write CR: Clear the START bit, set the STOP bit, and set the TB_EN bit
   - Wait for the data transmit interrupt - Read SR: TD (1), and AL (0), then write 1 to clear the corresponding SR bits
   - Go to the initial step

   (b). If the master wants to the send the second data byte transfer after the first data byte transfer:
   - Write DR: First data will be sent.
   - Write CR: Clear the START bit, clear the STOP bit, and set the TB_EN bit
   - Wait for the data transmit interrupt - Read SR: TD (1), NACK (0), and AL (0), then write 1 to clear the corresponding SR bits
   - Go to step 5 or step 6
6. If the master wants to send data byte transfer on the same slave or other slaves (i.e. repeat-start): Go to step 2

8.5.4.4. Programming Guide for RX in Master Mode

1. Write CDR: Set the clock count
2. Write DR: Target slave address and RW bit for slave
3. Write CR: Enable all interrupts, set MST_EN, set the I2C enable, set the START bit, clear the STOP bit, and set the TB_EN bit
4. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (0), and AL (0), then write 1 to clear the corresponding SR bits

Vangotech

5. (a). If the master wants to stop after one byte of data has been received:
   - Writ CR: Clear the START bit, set the STOP bit, set the NACK bit, and set the TB_EN bit
   - Wait for the data receive interrupt - Read SR: TD (1), NACK (1), then write 1 to clear the corresponding SR bits
   - Read DR: To get data
   - Go to the initial step

   (b). If the master wants to receive the second data byte transfer after the first data byte transfer:
   - Write CR: Clear the START bit, clear the STOP bit, and set the TB_EN bit
   - Wait for the data receive interrupt - Read SR: TD (1), NACK (0), then write 1 to clear the corresponding SR bits
   - Read DR: To get data
   - Go to step 5 or step 6
6. If the master wants to receive the data byte transfer from the same slave or other slaves (i.e. repeat-start): Go to step 2

8.5.4.5. Programming Guide for TX in Master Mode with HS-Mode (or START Byte)

1. Write CDR: Set COUNT for F/S-mode and set COUNTH for HS-mode
2. Write DR: Reserved address and RW bit (START byte: 0x01 or HS-mode master code: 8'000001XXX) for slave
3. Write CR: Enable all interrupts, set MST_EN, set the I2C enable, set the START bit, clear the STOP bit, and set the TB_EN bit
4. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (1), and AL (0), then write 1 to clear the corresponding SR bits
5. Write DR: Target slave address and RW bit for slave
6. Write CR: Set HS_mode bit to 1 to transfer data in HS-mode by using CDR[31:20], enable all interrupts, set MST_EN, set the I2C enable, set the START bit, clear the STOP bit, and set the TB_EN bit
7. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (0), RW (0), and AL (0), then write 1 to clear the corresponding SR bits
8. (a). If the master wants to stop after one byte of data has been transferred:
   - Write DR: First data will be sent.
   - Write CR: Set HS_mode bit, clear the START bit, set the STOP bit, and set the TB_EN bit
   - Wait for the data transmit interrupt - Read SR: TD (1), and AL (0), then write 1 to clear the corresponding SR bits
   - Go to the initial step

   (b). If the master wants to the send the second data byte transfer after the first data byte transfer:
   - Write DR: First data will be sent.
   - Write CR: Set HS_mode bit, clear the START bit, clear the STOP bit, and set the TB_EN bit
   - Wait for the data transmit interrupt - Read SR: TD (1), NACK (0), and AL (0), then write 1 to clear the corresponding SR bits
   - Go to step 8 or step 9
9. If the master wants to send data byte transfer on the same slave or other slaves (i.e. repeat-

start): Go to step 5

Note: As long as the STOP interrupt is issued by master, I2C bus will exit the HS-mode and recover to F/S-mode.

8.5.4.6. Programming Guide for RX in Master Mode with HS-Mode (or START Byte)

1. Write CDR: Set COUNT for F/S-mode and set COUNTH for HS-mode
2. Write DR: Reserved address and RW bit (START byte: 0x01 or HS-mode master code: 8'000001XXX) for slave
3. Write CR: Enable all interrupts, set MST_EN, set the I2C enable, set the START bit, clear the STOP it, and set the TB_EN bit
4. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (1), and AL (0), then write 1 to clear the corresponding SR bits
5. Write DR: Target slave address and RW bit for slave
6. Write CR: Set HS_mode bit to 1 to transfer data in HS-mode by using CDR[31:20], enable all interrupts, set MST_EN, set the I2C enable, set the START bit, clear the STOP bit, and set the TB_EN bit
7. Wait for the data transmit interrupt - Read SR:
   TD (1), NACK (0), RW (1), and AL (0), then write 1 to clear the corresponding SR bits
8. (a). If the master wants to stop after one byte of data has been received:
   - Writ CR: Set HS_mode bit, clear the START bit, set the STOP bit, set the NACK bit, and set the TB_EN bit
   - Wait for the data receive interrupt - Read SR: TD (1), NACK (1), then write 1 to clear the corresponding SR bits
   - Read DR: To get data
   - Go to the initial step

   (b). If the master wants to receive the second data byte transfer after the first data byte transfer:
   - Write CR: Set HS_mode bit, clear the START bit, clear the STOP bit, and set the TB_EN bit
   - Wait for the data receive interrupt - Read SR: TD (1), NACK (0), then write 1 to clear the corresponding SR bits
   - Read DR: To get data
   - Go to step 8 or step 9
9. If the master wants to receive the data byte transfer from the same slave or other slave (i.e. repeat-start): Go to step 5

Note: As long as the STOP interrupt is issued by master, I2C bus will exit the HS-mode and recover to F/S-mode.

8.5.4.7. Programming Guide for TX/RX in Slave Mode with HS-Mode (or START Byte)

1. Write AR: Set the slave address
2. Write CR: Enable all interrupts and I2C enable, and disable MST_EN
3. Wait for the data receive interrupt - Read SR:
   HSS (1) or SBS (1), then write 1 to clear the corresponding SR bits

Vangotech

4. Wait for the data receive interrupt - Read SR:
   - If RW (1), TD (1), SAM (1), I2CB (1), and NACK (0), then write 1 to clear the corresponding SR bits, go to step 4
   - If RW (0), TD (1), SAM (1), I2CB (1), and NACK (0), then write 1 to clear the corresponding SR bits, go to step 9
5. Write DR: Load the data byte to DR for transfer
6. Write CR: Set the TB_EN bit
7. Wait for the data transmit interrupt - Read SR:
   TD (1) or STOP (1), then write 1 to clear the corresponding SR bits
8. If STOP is set, go to the initial step.If STOP is not set,go to step 5 to load the second data byte.
9. Write CR: Set the TB_EN bit and set NACK if required
10. Wait for the data receive interrupt - Read SR:
    - If TD (1) and STOP (0): Write 1 to clear the corresponding SR bits, then go to step 10
    - If TD (0) and STOP (1): Write 1 to clear the corresponding SR bits, then go to step 11
11. Read DR: To get data
12. If STOP is set, go to the initial step. If STOP is not set, go to step 9 to get the next data.

8.5.4.8. Programming Guide for Master TX Burst Mode

1. Write CDR: Set COUNT and COUNTH
2. Write AR: Set ADDR, MEM_IDX, and ADDR10EN (If users need to enable 10-bit addressing mode).Set M2BIDX_EN and MEM_IDX2 (If users need to transmit 2nd index). [1]
3. Write BSTMR: Set total burst data count (BSTTDC)
4. Write data to DR
5. Write CR: Enable TD interrupts, set HS_mode (if needed), and set CR[25:24] to 2'b01 for TX
6. Wait for the data transmit interrupt - Read SR:
   - If NACK (1), TD (0), then write 1 to clear the corresponding SR bits, TX burst may fail during the data or address transfer, set CR[0] to 1 to reset FTIIC010
   - If AL (1), then write 1 to clear the corresponding SR bits, set CR[0] to 1 to reset FTIIC010
   - If NACK (0), TD (1), then write 1 to clear the corresponding SR bits, TX burst done
   - If NACK (1), TD (1), then write 1 to clear the corresponding SR bits, TX burst may fail at the last data.

[1]

Single/Multi-byte write	S + (SAD + Wr) + ack + IDX + ack + data + ack + … + data + ack + P
Combine format read	S + (SAD + Wr) + ack + IDX + ack + Sr + (SAD + Rd) + ack + data + ack + … + data + nack + P
10-bit addressing write	S + (1st SAD + Wr) + ack + (2nd SAD) + ack + IDX + ack + data + ack + … + data + ack + P
10-bit addressing read	S + (1st SAD + Wr) + ack + (2nd SAD) + ack + IDX + ack + Sr + (1st SAD + Rd) + ack + data + ack + …+ nack + P
HS single/multi-byte write	S + (Master code) + Sr + (SAD + Wr) + ack + IDX + ack + data + ack + … + data + ack + P

HS combine format read	S + (Master code) + Sr + (SAD + Wr) + ack + IDX + ack + Sr + (SAD + Rd) + ack + data + ack + … + data + nack + P

8.5.4.9. Programming Guide for Master RX Burst Mode

1. Write CDR: Set COUNT and COUNTH
2. Write AR: Set ADDR, MEM_IDX,and ADDR10EN(If users need to enable 10-bit addressing mode).Set M2BIDX_EN and MEM_IDX2 (If users need to transmit 2nd index). Please refer to the note above.
3. Write BSTMR: Set total burst data count (BSTTDC), and burst threshold (BSTTHOD)
4. Write CR: Enable TD and BSTTHOD interrupts, set HS_mode (if needed), and set CR[25:24] to 2'b10 for RX
5. Wait for the BSTTHOD or TD interrupt - Read SR:
   - If NACK (1), TD (0), then write 1 to clear the corresponding SR bits, TX burst may fail during the data or address transfer, set CR[0] to 1 to reset the FTIIC010
   - If AL (1), then write 1 to clear the corresponding SR bits, set CR[0] to 1 to reset FTIIC010
   - If BSTTHOD_SR (1) and TD (0), then users (a) read number of BSTTHOD data, (b) write 1 to clear the corresponding SR bits, and (c) go back to step 4. While users read the data, the controller is keeping receiving data at the same time.
   - If BSTTHOD_SR (1) and TD (1), users read the number of BSTTDC data, and then write 1 to clear the corresponding SR bits.
   - If BSTTHOD_SR (0) and TD (1), users read the rest number (BSTTDC - BSTTHOD) of data, and then write 1 to clear the corresponding SR bit

8.5.5. Register List

Module Name	Base Address
I2C_0	0x5550 0000
I2C_1	0x5560 0000
I2C_2	0x5570 0000
I2C_3	0x56F0 0000
I2C_4	0x5700 0000

Register Name	Offset	Description
CR	000	I2C Control Register (CR)
SR	004	I2C Status Register (SR)
CDR	008	I2C Clock Division Register (CDR)
DR	00C	I2C Data Register (DR)
SAR	010	I2C Address Register (AR)
TGSR	014	I2C Setup/Hold Time and Glitch Suppression Setting Register (TGSR)
BMR	018	I2C Bus Monitor Register (BMR)
BSTMR	01C	I2C Burst Mode Register (BSTMR)
REVISION	030	I2C Revision Register

8.5.6. Register Description

The following subsections describe the details of the I2C controller register.

Vangotech

8.5.6.1. I2C Control Register

The host processor uses the I2C Control Register (CR) to control the I2C interface controller for transmitting and receiving data from the I2C bus. Set MST_EN to '0' to enter the slave mode. The START and STOP bits can only be set in the master mode.

Table 8-99. I2C Control Register (Offset: 0x00)

Bit	Name	Type	Description	Default Value
[31:26]	-	Rsvd	Reserved	0
[25:24]	FLOW_CTRL_EN	RW	2'h0: No action. Users need to follow Section 8.5.4.1 ~ Section 8.5.4.7 to issue I2C protocol (Master/Slave). 2'h1: Master TX Burst Mode. Issue a write burst transaction in the master mode. 2'h2: Master RX Burst Mode. Issue a read burst transaction in the master mode.	0
[23:22]	-	Rsvd	Reserved	0
21	SBI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects a START byte in the slave mode.	0
20	HSI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects an HS-mode code in the slave mode.	0
19	HS_mode	RW	This register is used to switch between HS-mode and F/S-mode. When users set this register to 1, the I2C controller will use CDR[31:20] (0x08) to generate the corresponding HS-mode SCL.	0
18	Arb_off	RW	This register is used to ignore the arbitration lose detection in the single I2C master environment. Nonetheless, the bit 10 of SR (0x14) and ISI2C will assert (if CR[13] is 1) to notify CPU that the I2C bus has detected the arbitration lose, but FTIIC010 will still finish the entire I2C protocol.	1
17	Test_bit	RW	Special test mode This bit must be set to '0'.	0
16	SDA_LOW	RW	If it is set, SDAout will be tied to '0'. For a normal case, it is suggested setting this bit to '0'.	0

Vangotech

15	SCL_LOW	RW	If it is set, SCLout will be tied to '0'. For a normal case, it is suggested setting this bit to '0'.	0
14	STARTI_EN	RW	If it is set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects a start condition on the I2C bus.	0
13	ALI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller loses the arbitration in the master mode.	0
12	SAMI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects a slave address that matches the SAR register or a general call address (When GC_EN is set).	0
11	STOPI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects a stop condition happening on the I2C bus.	0
10	NACKI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller detects the non-ACK responses from the I2C bus.	0
9	TDI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller DR register has transmitted/received one data byte from the I2C bus, or has transmitted/received the total data count (bits [13:8] of register offset 0x1C) in the master RX/TX burst mode.	0
8	BSTTHODI_EN	RW	If set, this bit will enable the FTIIC010 I2C controller to interrupt the host processor when the FTIIC010 I2C controller read data reaches the threshold (bits [5:0] of register offset 0x1C) in the master RX burst mode.	0
7	TB_EN	RW	When Transfer Byte Enable (TB_EN) is set, the FTIIC010 I2C controller is ready to receive or transmit one byte. Otherwise, the FTIIC010 I2C controller will insert the wait state by pulling SCLout low in the I2C bus.	0

Vangotech

			Please note that TB_EN will be automatically cleared when the device receive or transmit one byte or detect the stop/start condition.	
6	ACK/NACK	RW	The acknowledge signal sent by the FTIIC010 I2C controller when the FTIIC010 I2C controller is in the master-receive or slave-receive mode. If I2C controller detects stop condition on I2C bus, this bit will auto clear to 0. 0: ACK 1: NACK	0
5	STOP	RW	I2C controller initiates a stop condition after transferring the next data byte on the I2C bus when I2C is in the master mode.	0
4	START	RW	I2C controller initiates a start condition when the I2C bus is idle or initiates a repeated start condition after transferring the next data byte on the I2C bus in the master mode.	0
3	GC_EN	RW	Enable the I2C controller to respond to a general call message as a slave	0
2	MST_EN	RW	Enable the I2C controller clock output for the master mode operation	0
1	I2C_EN	RW	Enable the I2C bus interface controller	0
0	I2C_RST	RW	Reset the I2C controller This bit will be automatically cleared after two PCLK clocks.	0

8.5.6.2. I2C Status Registers

I2C interrupts are signaled through the ISI2C pin. The ISI2C pin will be set when both interrupt enable bits in the I2C Control Register (CR) and the corresponding status register are set. When the I2C controller interrupt is asserted, software reads the SR bits to check the status of the I2C controller and write 1 to clear SR.

SR is used to clear the following interrupts by reading the register status:

- Data transfer (TX/RX) is completed.
- Slave address is detected.
- Bus NACK is detected.
- Start condition is detected.
- Stop condition is detected.
- Arbitration loss is detected.
- General Call is detected.
- START byte is detected

Table 8-100 shows the bit assignments of the I2C status register.

Table 8-100. I2C Status Register(SR,Offset:0x04)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
23	SBS	RW1C	This signal will be set when the I2C controller detects a START byte on the I2C bus.	-
22	HSS	RW1C	This signal will be set when the I2C controller detects an HS-mode code on the I2C bus.	-
[21:12]	-	Rsvd	Reserved	0
11	START	RW1C	This signal will be set when the I2C controller detects a start condition on the I2C bus.	0
10	AL	RW1C	This signal will be set when the I2C controller loses arbitration in the master mode.	0
9	GC	RW1C	This signal will be set when the I2C controller receives slave address that matches the general-call address, and the I2C controller is in the slave mode.	0
8	SAM	RW1C	When the I2C controller is in the slave mode, this signal will be set while the I2C controller is receiving slave address that matches the address in the Address Register (AR), and it should assert while GC is hitting.	0
7	STOP	RW1C	This signal will be set when the I2C controller detects a stop condition in the I2C bus.	0
6	NACK	RW1C	This signal will be set when the I2C controller detects non-ACK responses from the slave device after transmitting one byte of data when the I2C controller is in the master mode.	0
5	TD	RW1C	This signal will be set when data finishes transferring (TX/RX) a byte, or the total data count (bits [13:8] of register offset 0x1C) is finished in the master RX/TX burst mode.	0
4	BSTTHOD_SR	RW1C	This signal will be set when the FTIIC010 I2C controller read data reaches the threshold (bits [5:0] of register offset 0x1C) in the master RX burst mode.	0
3	BB	RO	This signal will be set when the I2C bus is busy, but the I2C controller is not involved in the transaction.	0
2	I2CB	RO	This signal will be set when the I2C controller is busy, i.e. during the time period between START and STOP.	0
1	-	Rsvd	Reserved	0
0	RW	RO	This signal will be set to 1 when the I2C controller serves in the RX or TX mode. 0: Maxter in the TX mode, or slave in the RX mode	0

Vangotech

			1: Maxter in the RX mode, or slave the in TX mode	

8.5.6.3. I2C Clock Division Register

The I2C Clock Division Register (CDR) defines the divided values that are used to generate the I2C SCL clock. This register is used with an internal 32-bit counter. When the MST_EN bit in the control register is set, this counter decrements from the programmable value to zero, and then reloads the programmable value and decrements again. Each time the counter reaches zero, the SCL line transaction starts from high to low or vice versa, depending on the current state. This register can be configured to select the transfer speed needed in the I2C bus.

Please note that since the controller supports the I2C bus speed of up to 3.4 MHz, the minimum PCLK must be larger than 68 MHz.

Table 8-101. I2C Clock Division Register (CDR, Offset: 0x08)

Bit	Name	Type	Description	Default Value
[31:28]	DUTY_OFFSET	RW	This offset can change the I2C clock duty. Users must configure this register offset to fit the LOW period of the SCL clock that is claimed in I2C bus specification. High period = ((COUNT) - DUTY_OFFSET) + GSR + 3) * PCLK period Low period = ((COUNT) + DUTY_OFFSET) + 1) * PCLK period Please note that COUNT must be larger than DUTY_OFFSET and ((COUNT) - DUTY_OFFSET) is equal to or larger than 2.	0
[27:20]	COUNTH	RW	Counter value for the HS mode is used to generate an I2C clock (SCLout) from the internal bus clock, PCLK. The relation between PCLK and the I2C bus clock (SCLout) is as shown in the following equation: SCLout (Hz) = PCLK (Hz)/(2 * COUNTH + GSR + 4) Please note that the minimum value of COUNTH is equal to or larger than GSR + TSR + 4, where GSR and TSR are from the register offset 0x14. For example: If PCLK is 60 MHz, COUNTH is 7, and GSR is 2, and then SCLout is 3.0 MHz.	8
[19:0]	COUNT	RW	Counter value for F/S-mode, is used to generate an I2C clock (SCLout) from the internal bus clock PCLK. The relation between PCLK and the I2C bus clock (SCLout) is shown in the following equation: SCLout (Hz) = PCLK (Hz)/(2 * COUNT + GSR + 4) Please note the minimum value of COUNT is equal to or larger than GSR + TSR + 4, where GSR, TSR are from register offset 0x14.	8

Vangotech

			For example: If PCLK is 20 MHz, COUNT is 7, GSR is 2, and then SCLout is 1 MHz.	

8.5.6.4. I2C Data Register

The I2C Data Register (DR) is used by the host processor to transmit and receive data from the I2C bus.DR is accessed by either the host processor or the I2C controller Shift Register (SHR). Data coming from the I2C bus interface are received by the DR register after a full data byte has been received. Data going out of the I2C bus interface are written into the DR register by the host processor and sent to the serial bus.

When the I2C controller is in the transmit mode, the host processor writes data into the DR register over the APB bus. This occurs when a master transition is initiated or when a data transmit interrupt is signaled. Data are moved from the DR register to the SHR register when the transfer byte enable (TB_EN) in the control register is set. The data transmit interrupt is signaled when one byte of data has been transferred on the I2C bus. If DR is not written by the host processor before the next byte package, the I2C controller will insert a wait state until the host processor writes to DR and sets the transfer byte enable bit (TB_EN).

When the I2C controller is in the receive mode, the processor reads DR register data over the APB bus,and this occurs when the DR received interrupt is signaled. When the I2C controller has received one new data byte, it will automatically clear TB_EN and issue ACK/NACK on the I2C bus. After issuing ACK/NACK, the I2C controller will insert the TD interrupt to processor. Users must set the TB_EN bit again for the next byte transfer on the I2C bus.

Table 8-102. I2C Data Register (Offset: 0x0C)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	DR	RW	Buffer for the I2C bus data transmission and reception	0

8.5.6.5. I2C Address Register

The I2C address register defines the 10-bit I2C controller that transmits slave ID, 1/2 indexes, 10 bit addressing mode in the master RX/TX burst mode, defines slave address which the processor responds to when the I2C controller operates in the slave mode, or defines the slave address transmitted in the master burst mode. The host processor writes this register before enabling the I2C operation. The register is fully programmable to be set to a value rather than fix to the slave peripheral address preexisting in the system.

Table 8-103. I2C Address Register (AR, Offset: 0x10)

Bit	Name	Type	Description	Default Value
[31:24]	MEM_IDX2	RW	2nd transmitted index for the master RX/TX burst mode	0
[23:16]	MEM_IDX	RW	1st transmitted index for the master RX/TX burst mode	0
[15:14]	-	Rsvd	Reserved	0
13	M2BIDX_EN	RW	Set to 1 to transfer 2nd index bytes	0
12	ADDR10EN	RW	Set to 1 to enable 10-bit addressing mode	0
[11:10]	-	Rsvd	Reserved	0
[9:7]	ADDR2	RW	The most significant 3-bit address to which the I2C	0

			controller transmits in the master RX/TX burst mode, or gives a response when I2C operates in the slave mode if ADDR10EN = '1'. When ADDR10EN = '0', the I2C controller ignores these three bits.	
[6:0]	ADDR	RW	In the 7-bit addressing mode (ADDR10EN = '0'), the 7-bit address is for the I2C controller transmitted in the master burst mode, or for the I2C controller gives a response when I2C operates in the slave mode.	0

8.5.6.6. I2C Set/Hold Time and Glitch Suppression Setting Register

The I2C TGSR[13:10] defines the values of the PCLK clock period when the I2C bus interface has a built-in glitch suppression logic. Glitches are suppressed according to TGSR[13:10] * PCLK clock period. For example, with a 66 MHz (15-ns period) PCLK clock, and TGSR[13:10] = 4'b100, glitches of 60 ns or less are suppressed. With a 40 MHz (25-ns period) clock, and TGSR[13:10] = 4'b10,glitches of 50 ns or less are suppressed. This is within the 50-ns glitch suppression specification.The only limitation is COUNT or COUNTH >= 4 + GSR + TSR. Please note that when FTIIC010 is slave, SCL will be released after the wait state and then count TSR+1 cycles.

Table 8-104. I2C Set/Hold Time and Glitch Suppression Setting Register (TGSR, Offset: 0x14)

Bit	Name	Type	Description	Default Value
[31:14]	-	Rsvd	Reserved	0
[13:10]	GSR	RW	These bits define the values of the PCLK clock period when the I2C bus interface has the built-in glitch suppression logic. Glitch is suppressed according to the "GSR * PCLK" clock period.	0x1
[9:0]	TSR	RW	These bits define the delay values of the PCLK clock cycles between SCL and SDA while data or acknowledgement is driven. The actual delay value is GSR + TSR + 4. Please refer to Figure 8-26 for more details. Please note that TSR cannot be set to '0'.	0x1

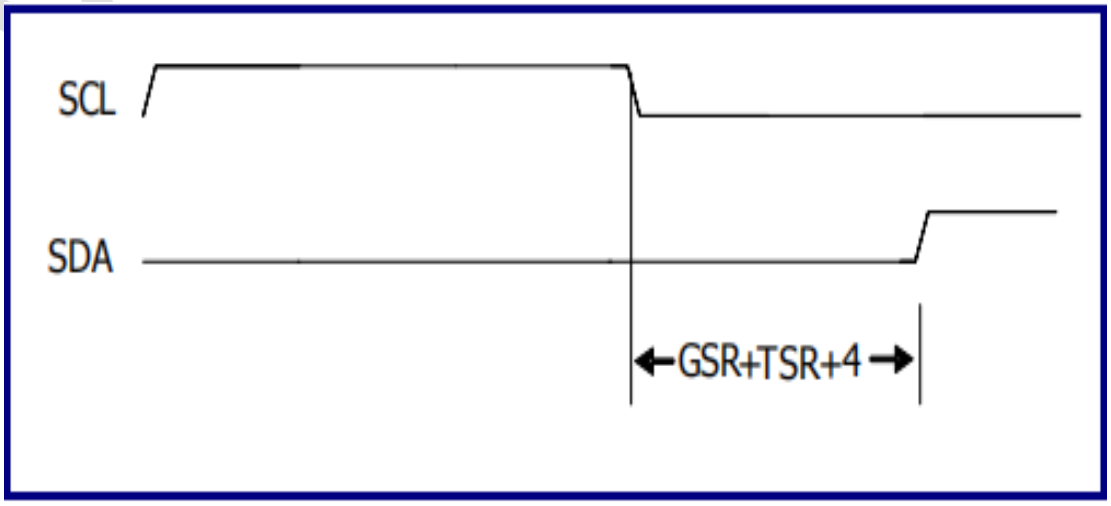


Figure 8-26. Relationship among TSR, SCL, and SDA

8.5.6.7. I2C Bus Monitor Register

Table 8-105. I2C Bus Monitor Register (BMR, Offset: 0x18)

Bit	Name	Type	Description	Default Value
[31:2]	-	Rsvd	Reserved	0
1	SCLin	RO	This bit continuously reflects the value of the SCLin pin.	1
0	SDAin	RO	This bit continuously reflects the value of the SDAin pin.	1

8.5.6.8. I2C Burst Mode Register

Table 8-106. I2C Burst Mode Register (BSTMR, Offset: 0x1C)

Bit	Name	Type	Description	Default Value
[31:19]	-	Rsvd	Reserved	0
[18:16]	BUFHW	RO	Buffer depth 3'h5: 32 bytes	5
[15:14]	-	Rsvd	Reserved	0
[13:8]	BSTTDC	RW	These bits are the total burst data count, which must be less than or equal to BUFHW. Please do not set them to 0.	1
[7:6]	-	Rsvd	Reserved	0
[5:0]	BSTTHOD	RW	These bits must be less than or equal to BSTTDC. Please note that BSTTHOD is only valid in the master RX burst mode. Please do not set them to 0.	1

8.5.6.9. I2C Revision Register

Table 8-107. I2C Revision Register (REVISION, Offset: 0x30)

Bit	Name	Type	Description	Default Value
[31:0]	REVISION	RO	These bits represent the revision number.	0x0001_2100

Vangotech

8.6. UART

8.6.1. Overview

The UART and IrDA controller is a serial communication element that implements the most common infrared communication protocols. In addition to the infrared modes, this device also provides a UART mode of the operation that is backward compatible to 16550 to support existing the communication software.

8.6.2. Features

The UART and IrDA communication controller provides the following features:

- Supports AMBA 2.0 APB bus
- Firmware compatible with high-speed NS 16C550A UART
- IrDA 1.3 SIR with data rates of up to 115.2 kbps
- Programmable SIR pulse width as 1.6 μs or 3/16 of baud-rate pulse width
- Supports IrDA 1.3 FIR/ SIR (UART5/ UART8 only supports IrDA)
- Multi-frame transmission and reception in FIR mode
- Back-to-back infrared frame transmission and reception in FIR mode
- 32-bit IEEE 802 CRC32 hardware CRC generators and checkers for FIR communications
- Hardware supports 16 transmit/receive FIFOs
- Hardware configurable UART to completely remove IrDA related circuits
- Hardware configurable to remove internal diagnostic circuit
- Internal diagnostic capabilities:
  - Break, parity, overrun, and framing error simulations in UART mode
  - CRC error and physical error simulations in FIR mode
- Supports hardware flow control by RTS/CTS

8.6.3. Block Diagram

Figure 8-27 shows the block diagram of the UART and IrDA communication controller.

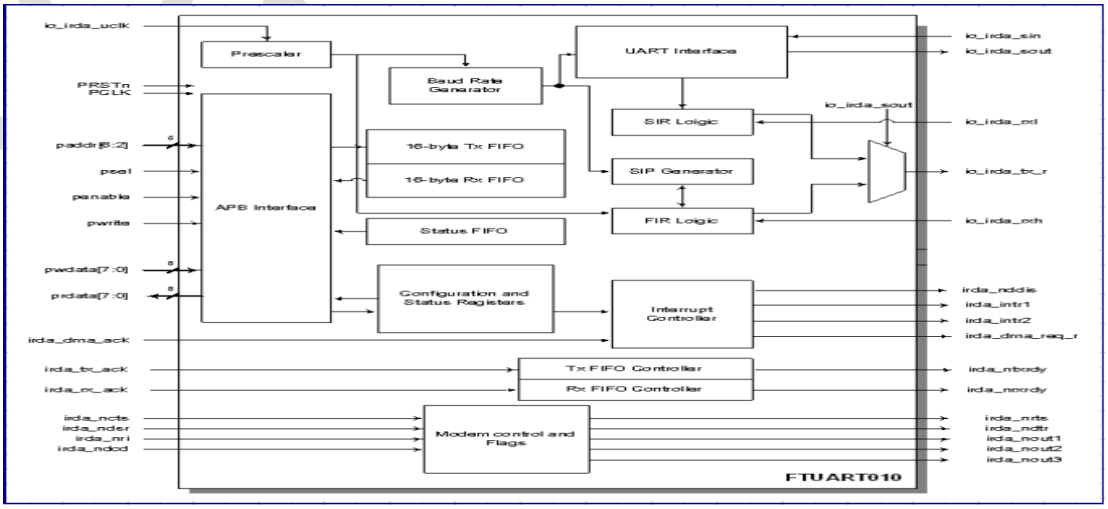


Figure 8-27. Block Diagram of UART and IrDA Communication Controller

The main building blocks of FTUART010 are described below.

8.6.3.1. APB Interface

The APB interface provides the accesses to the controller registers and data FIFOs.

8.6.3.2. Prescaler

The prescaler divides io_irda_uclk by using the divisors of 1 to 25-1 and produces a pre-divided clock,PB16XCLK, to the baud rate generator.

The prescaler also recovers the sampling clock from the received signal in the FIR mode. The clock,PB16XCLK, derived from the prescaler is used to sample the received signal in the FIR mode.

8.6.3.3. Baud Rate Generator

The baud rate generator is capable of dividing the pre-divided clock, PB16XCLK, by using the divisors of 1 to 216-1, and producing a 16 times baud clock, B16XCLK.

8.6.3.4. TX FIFO and RX FIFO

Both TX FIFO and RX FIFO are 8-bit wide, configurable location deep. The CPU data written across the APB interface are stored in TX FIFO until being read out by the transmit logic. The received data are stored in the receive FIFO by the receive logic until being read out by CPU across the APB interface. Both TX FIFO and RX FIFO can be disabled to act as a one-byte holding register.

FIFOs are enabled or disabled by using the FIFO Control Register (FCR). When FIFOs are disabled,the behavior is the same as if they had not been implemented. The UART working without FIFOs is commonly called the 16450 mode, due to the name of the industry standard UART without FIFOs. In the SIR and FIR modes, TX FIFO and RX FIFO should always be enabled.

8.6.3.5. Status FIFO(ST FIFO)

During the data reception, the corresponding error bits, Parity Error, Framing Error, and Break Condition,are stored in the Status FIFO (ST FIFO). Disable TX/RX FIFOs will also disable ST FIFO.

The ST FIFO is also used in the FIR mode. It is always an intended 8-level FIFO to support the back to-back incoming frames in the DMA mode. Each entry of ST FIFO contains both the status information and frame length of a single frame. ST FIFO is flushed when the hardware reset occurs or when RX FIFO is soft reset.The status and length information of the received frames are loaded into ST FIFO whenever the DMA_EN bit in the MDR register is set to '1'.

8.6.3.6. Configuration and Status Registers

The register files store the data written or to be read across the AMBA APB interface.

8.6.3.7. Modem Control and Flags

This block controls the interface with the modem and provides the current state of the control lines from the modem (Or peripheral device) to CPU.

8.6.3.8. SIR Logic

The SIR logic includes the SIR encoder and SIR decoder. The SIR encoder converts the UART zeros to the light pulses and ones to no pulses. The converted optical pulse duration can be 3/16 of a bit duration or 1.6 μ s by programming the ACR[7] bit. The decoder will "stretch" a valid light pulse to a single bit cell of zero (Low) level and no light pulses to ones (High).

8.6.3.9. FIR Logic

The FIR logic includes the FIR encoder/decoder and transmitter/receiver. The FIR encoder uses the Four

Pulse Position Modulation (4 PPM) scheme, in which two bits of data (Symbol) are encoded into a 4-bit cell "chip." Each bit cell or "chip" is 125 ns wide, and each input byte (Symbol) requires 2 μ s to send. Unlike the SIR, a chip with logic 1 will produce a light pulse, and a chip with logic 0 will not produce the light pulse.The decoder takes a valid light pulse as a single chip with logic one and no light pulses to zeros, and de-modulates the encoded 4 PPM input signal to its raw data.

8.6.3.10. SIP Generator

In the FIR operating mode, the SIP generator sends a 1.6 μ s Serial Infrared Interaction Pulse (SIP) according to the setting of MDR[2] bit and ACR[4] bit.

8.6.4. Operations and Functional Descriptions

FTUART010 operates in three modes. For a proper operation, the device must be correctly programmed in specific mode. The operation modes include:

- Universal Asynchronous Receiver Transmitter (UART) mode
- Serial Infrared (IrDA SIR) mode
- Fast Infrared (IrDA FIR) mode

The basic data communication involves at least two devices. In the data transmission mode, the first device transmits the data and the second device receives the transmitted data. In the reception mode, the second device transmits the data and the first device receives the data. When FTUART010 is configured for the UART mode, both data transmission and data reception can occur simultaneously, which is known as a full-duplex operation. In the Infrared (IR) modes, either the data transmission or data reception is possible at any time but not simultaneously, which is known as a half-duplex operation.

8.6.4.1. UART Mode

The UART mode is the default operation after power up and system reset. This mode uses a wired interface for the serial communication with a remote device or a modem. FTUART010 can operate in a full-duplex mode, such as the data transmission and reception can occur simultaneously. FTUART010 in the UART mode works as a regular serial asynchronous communication controller that converts the parallel data received from CPU or the DMA controller into the serial data. It also converts the serial data received on the serial input terminal into parallel data.

Throughout this block data sheet, the term character will appear. This term is commonly used to describe a word of data being transmitted serially. A character can have several possible word lengths,including 5 bits, 6 bits, 7 bits, or 8 bits, managed by FTUART010.

This mode is designed to support the serial data communications with a remote peripheral device or modem by using a wired interface. The controller provides the transmit channel and receive channel that can concurrently operate to handle the full-duplex operation. They perform the parallel-to-serial conversion on the data characters received from CPU or a DMA controller, and the serial-to-parallel conversion on the data characters received from the serial interface. The format of the serial data stream is shown in Figure 8-28.

Vangotech

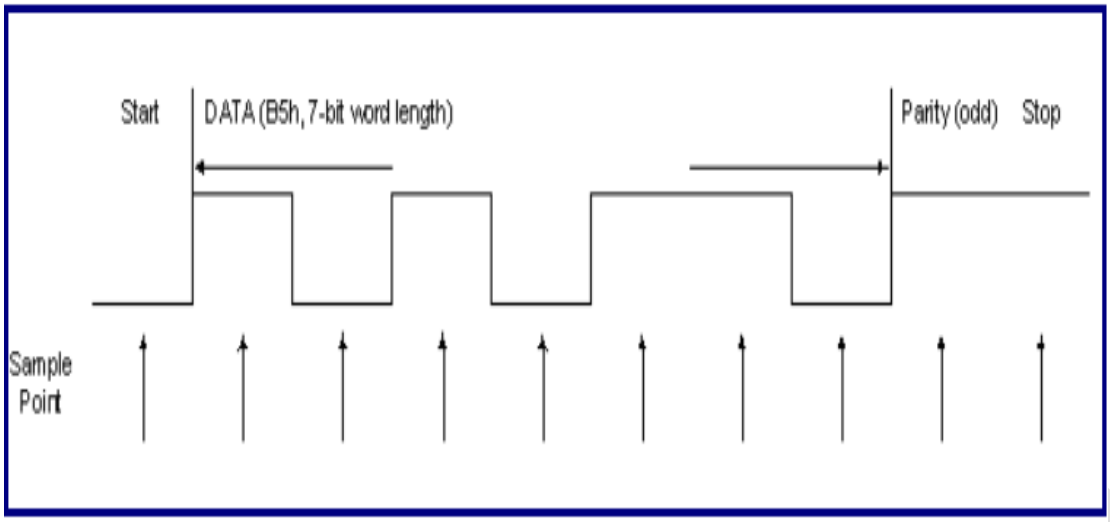


Figure 8-28. UART Data Representation and Sampling

A data character contains five to eight data bits. It is preceded by a start bit and is followed by an optional parity bit and a stop bit. Data is transferred in little-endian order (Least significant bit first).

The UART mode is the default mode of operation after power up or reset. In fact, after reset, the 16450-compatibility mode is selected. In addition to the 16450 and 16550 compatibility modes, IrDA modes are also available. When the FIR mode is selected, the interrupt sources are no longer prioritized.The additional features include the transmitter FIFO threshold, DMA capability, and interrupts on transmitter empty.

The clock for both transmit and receive channels is provided by an internal baud generator that divides the pre-scaled clock by any divisor value from 1 to 216 - 1. The output clock frequency of the baud generator must be programmed to be sixteen times the baud rate value. The baud generator input clock is derived from io_irda_uclk clock through a programmable prescaler. The PSR register determines the prescaler value. Its default value is 0x01.

Both the communications format and baud rate must be properly programmed before operation. The communications format is programmed by setting the LCR register, while the baud rate is selected by programming the baud generator divisor registers (DLL and DLM). The software can read the status of the device at any time during operation. The status information includes state for FIFO, and any other condition detected on the received data stream, like parity error, framing error, data overrun, or break event.

8.6.4.2. IrDA 1.3 SIR Mode

IrDA 1.3 SIR mode is the first operational mode that has been defined by the IrDA committee. This mode supports the bidirectional data communication with a remote device by using the infrared radiation as the transmission medium. IrDA 1.3 SIR mode allows the serial communication at the baud rates of up to 115.2 kbps. The format of the serial data is similar to the UART data format. Each data word is serially sent with a zero value start bit at the beginning, followed by the 8-data bits,and ending with one stop bit with a binary value of one. Sending a single infrared pulse signals a zero. A one is signaled by not sending any pulse. The width of each pulse can be either 1.6 μs or 3/16 of a single bit time (1.6 μs is equal to 3/16 of a bit time at 115.2 kbps). Therefore, each word begins with a pulse for the start bit. The device operation in the IrDA SIR mode is similar to the operation in the UART mode. The main difference is that those data transfer operations are normally performed in the half-duplex fashion. Select the IrDA SIR mode is controlled by the mode select bits in the MDR register.

Each data byte starts with a start bit 0, 1 byte of data, and then ends with at least a stop bit 1. Each serial data bit is encoded before the transmission and decoded after reception. A '1' is decoded with no IR pulse and a '0' is decoded by sending 3/16 of one bit time IR pulse. Similarly, the received serial pulse is decoded as a '0' and the absence of an IR pulse is decoded as a '1' (Please refer to Figure 8-29 for details).

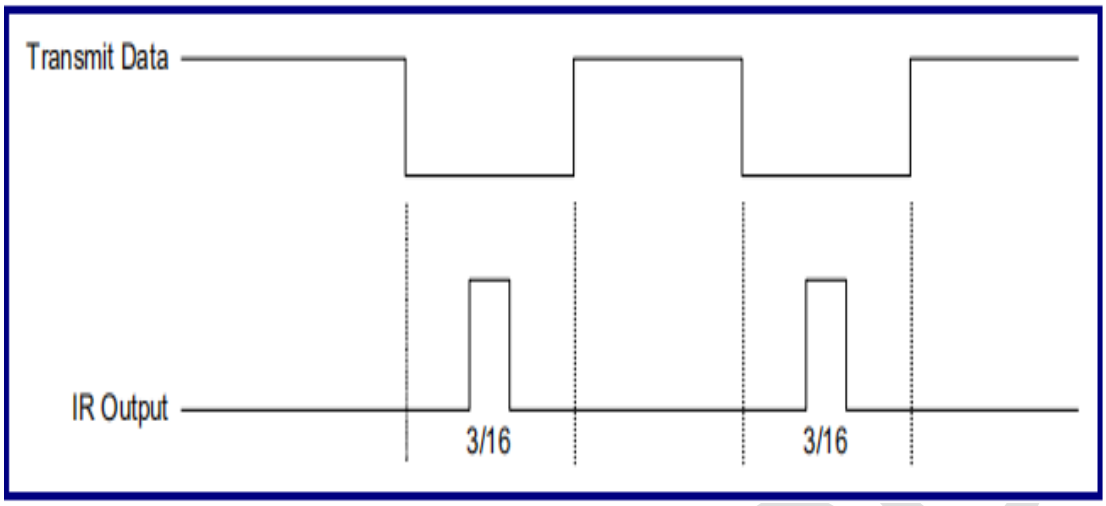


Figure 8-29. SIR Encoding

8.6.4.3. IrDA 1.3 FIR Mode

FTUART010 supports the IrDA 1.3 FIR mode, with the data rates of 4.0 Mbps. Please refer to the appropriate IrDA document for the detailed information of the frame format, encoding schemes, CRC sequences, and so on.

The FIR transmitter front-end section adds the preamble and start and stop flags to each frame and encodes the transmit data into a Four Pulse Position Modulation (4PPM) data stream. The FIR receiver front-end section strips the preamble and flags from the inbound data stream and decodes the 4PPM-encoded data while also checking for the coding violations.

The FIR front-end also automatically appends the CRC sequences to the transmitted frames and checks for the CRC errors on the received frames.

FIR Transmission Closing Method

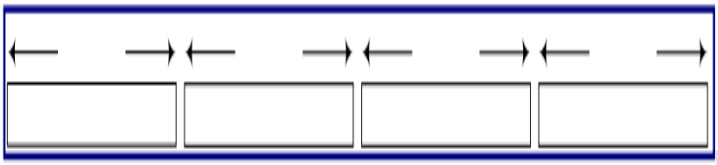
There are two methods that a transmission frame can be properly terminated in the FIR mode:

Frame-length method and set-EOT bit method. The two methods are described as follows:

- Frame-Length Counter method:

  This method can be used when the data transfers are performed in either the PIO mode or the DMA mode. This method is selected when the MDR[3] bit is equals to '0'. CPU writes the frame-length value to the TXLENH and TXLENL registers. The device automatically attaches an ending flag to the frame when the number of bytes transmitted is equal to the value of TXLENH and TXLENL. This method also allows a large data block to be automatically split into the equal-size back-to-back frames and can add an equal-size or different-size frame in the PIO or the DMA mode. The length of the last frame is defined in the registers, LSTFMLENH and LSTFMLENL. The number of the frames to be transmitted is defined in LSTFMLENH[7:5]. For example, when LSTFMLENH[7:5] are programmed as zeros, the data in TX FIFO will always be transmitted as the equal-size frames (The four-byte CRC is not shown in the figure for simplification).

When LSTFMLENH[7:5] are programmed as non-zeros, 3, the data in TX FIFO will always be transmitted as three equal-size frames plus one equal-size frame or different-size frame, please refer to the following figure for details:



Note: When MDR[2] bit is set to '0', SIP will not be issued until all back-to-back frames are transferred, for example: TX FIFO is empty.

- Set-EOT bit method:

  This method is used when data transfers are performed in the PIO mode. This method is selected when the MDR[3] bit is equal to '1'. CPU writes a '1' to the SET_EOT bit of the ACR register before writing the last byte to the TX FIFO. When CPU writes the last byte to the TX FIFO, the device internally sets the tag bit for the particular byte in TX FIFO. As FTUART010 in the transmission mode reads bytes from TX FIFO, the flag-bit information is used to attach an ending flag and properly terminate the frame. Set the TX FIFO trigger level to '1' may often result in TX FIFO underrun in this mode.

FIR Data-Receive Method in PIO Mode

When the data reaches the set threshold level of RX FIFO, the device interrupts CPU. When CPU is interrupted, it reads FMIIR_PIO to identify the source of the interrupt. When the source of the interrupt is the FMIIR_PIO[0] bit and not the FMIIR_PIO[4] bit, CPU goes to the threshold mode as follows.

- Threshold Mode

  In the threshold mode, CPU reads a number of bytes (Determined by the RX FIFO threshold level) from RX FIFO. For example, when the RX threshold value is set to '16', CPU can perform 16 consecutive read operations from RX FIFO. When the source of interrupts is the FMIIR_PIO[4] bit,CPU should read RXFF_CNTR (Offset: 0x5C) to determine the number of data bytes in RX FIFO when an End-Of-Frame (EOF) flag is detected.

Serial Infrared Interaction Pulse

In the FIR mode operation, the transmitter sends a 1.6-μ s Serial Infrared Interaction Pulse (SIP) at least once every 500 ms. The purpose of this special pulse is to inform the slow device (In the SIR mode) that the High-Speed device involved in the data transaction is currently occupied. When the MDR[2] bit is a '0' (The default value), FTUART010 in the transmission mode will always send a 1.6-μ s pulse at the end of a transmission frame. However, when bit MDR[2] is a '1', the transmission of a 1.6-μ s pulse will depend on the value of ACR[4]. CPU keeps a timer and sets ACR[4] at least once in every 500 ms. When MDR[2] is set to '1', FTUART010 in the transmission mode will send a 1.6-μs pulse only if the ACR[4] bit is a '1'. The advantage of this approach over the default value (Always sending a 1.6-μ s pulse at the end of a frame) is that FTUART010 in the transmission mode will not need to send the special 1.6-μ s pulse at the end of every frame. Sending a 1.6-μ s pulse at the end of every frame may increase the overhead.

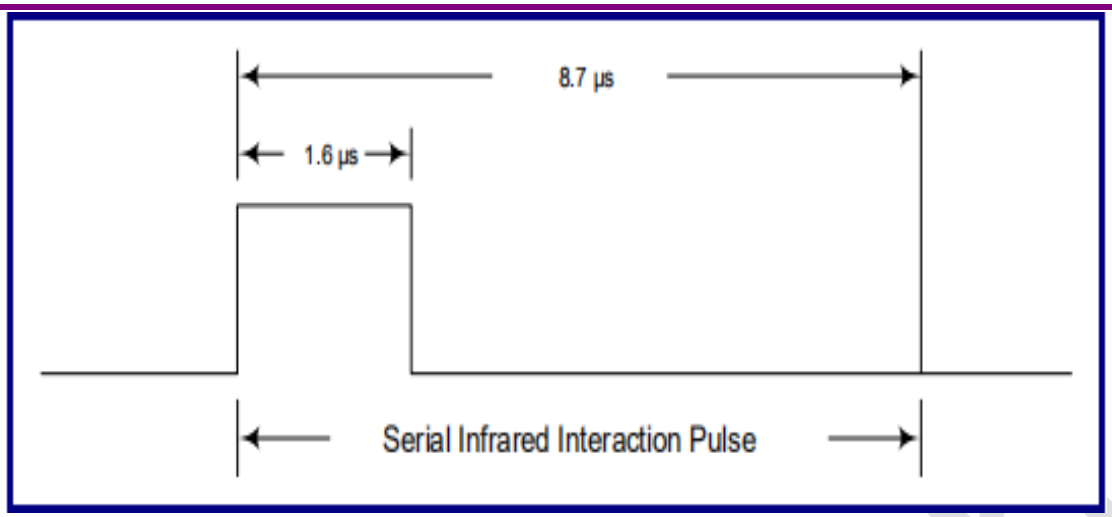


Figure 8-30. Serial Infrared Interaction Pulse

TX FIFO Underrun in FIR Mode

An underrun during the data transmission occurs when CPU fails to supply the data to TX FIFO. TX FIFO becomes empty before the end of the frame is transmitted. When an underrun occurs, FTUART010 closes the frame with an ending flag but attaches an incorrect CRC value. The receiving device detects the CRC error and discards the frame. FTUART010 sets an internal flag and further transmission of data is disabled. CPU must reset TX FIFO and read the RESUME register (Offset: 0x58). This read operation clears the internal flag.

RX FIFO Overrun in FIR Mode

An overrun occurs during data reception if CPU cannot timely read out data from RX FIFO and RX FIFO is overwritten. When an overrun occurs in a PIO mode, FTUART010 interrupts CPU with the FMIIR_PIO[2] bit and discards the remaining portion of the frame. When an overrun occurs,FTUART010 sets an internal flag and the receive operation of the next frame is disabled.Before the next frame can be received, CPU must reset RX FIFO and read the RESUME register. This read operation clears the internal flag.

When an RX FIFO overrun occurs in the DMA mode, the receiver continues to receive further frames. The status of data reception will be reported in the status FIFO.

ST FIFO Overrun in FIR Mode

The ST FIFO overrun occurs during data reception if CPU cannot timely read out data from ST FIFO and St FIFO is overwritten. The ST FIFO overrun occurs only in DMA mode. FTUART010 interrupts CPU with the FMIIR_DMA[2] bit and stops receiving the next frame. Then FTUART010 sets an internal flag and the receive operation of the next frame is disabled. Before the next frame can be received, CPU may read out data from ST FIFO or reset ST FIFO and read the RESUME register. This read operation clears the internal flag.

8.6.4.4. DMA Operation in FIR Mode

The data transfer in a DMA mode is used to achieve faster data transfer when FTUART010 is operating in the FIR mode. In a multi-application environment, where multiple applications are running at the same time, the DMA mode should be used to keep up with the high rate of data transfer without having underrun during data transmission or overrun during data reception. Frames can be transmitted and received back-to-back.

DMA Data-Transmit Mode

In the DMA mode, packets are automatically fragmented into the equal-sized frames (Decided by the

TXLENH and TXLENL register values) and properly transmitted. The last portion of the packet, which may be different from the register values of TXLENH and TXLENL, can be transmitted back-to-back by properly programming the LSTFMLENL and LSTFMLENH registers. If LSTFMLENH[7:5] are programmed as zeros,the interrupt FMIIR_DMA[5] bit is generated to inform CPU whenever a frame is transmitted.

If LSTFMLENH[7:5] are programmed as non-zeros, the interrupt FMIIR_DMA[5] bit will be generated to inform CPU that the data transmission is completed when all frames are transmitted. FTUART010 asserts the irda_dma_req_r signal when the number of bytes in the TX FIFO falls below the set trigger level and de-asserts the irda_dma_req_r when the irda_dma_ack is sampled high.

DMA Data-Receive Mode

The DMA controller controls the data reception of the back-to-back frames. The back-to-back frames are transferred to memory and the status of each received frame is stored in the Status FIFO that can hold up to eight entries. Each entry in the Status FIFO corresponds to one received frame. Each entry stores the length of each received frame and the error-status of that frame. CPU reads the Status FIFO entries to locate the frame boundaries and status of individual frames inside memory.

The Status FIFO has four trigger levels, which are 1, 4, 7, and 8. In a data receive transaction, the receiver expects to receive 1 to 7 frames. When the number of the received frames is equal to or greater than the set threshold value in the Status FIFO, FTUART010 will generate an interrupt FMIIR_DMA[0] bit. After CPU receives the Status FIFO interrupt (FMIIR_DMA[0] bit), it reads the STFF_STS register to determine the error-status of that frame. The Status FIFO Received Frame Length Low (STFF_RXLENL) and Status FIFO Received Frame Length High (STFF_RXLENH) are read to determine the length of the frame. CPU checks the content of the STFF_STS[4] bit, which is set to a '1' when this entry of Status FIFO is valid. If this bit is a '0', CPU should stop reading STFF_RXLENL and STFF_RXLENH.

The Status FIFO timeout interrupt (Bit FMIIR_DMA[1]) is set to 1 ms. This is useful when the number of frames in a data receive transaction is less than the Status FIFO threshold level. When CPU receives a Status FIFO timeout interrupt, it should read the STFIFO_STS, STFF_RXLENL, and STFF_RXLENH registers until bit STFF_STS[4] is '0' (Which indicates the invalid entry).

The Status FIFO Time-out conditions are as follows:

- At least one entry is in Status FIFO.
- More than 1 ms has elapsed since the last byte was loaded into Status FIFO from the receiver logic.
- More than 1 ms has elapsed since CPU reads the last entry from Status FIFO.

FTUART010 asserts the irda_dma_req_r signal when RX FIFO is not empty. The irda_dma_req_r signal is de-asserted when the irda_dma_ack signal is sampled high. An overrun in RX FIFO in the DMA mode is handled differently than in the programmed I/O mode. When an overrun occurs in RX FIFO, during any frame, data reception is terminated and the receiver state machine waits for the next frame. It continues receiving further frames. RXFIFO_ORUN is set to a '1' in the STFF_STS register for the frame in which the RX FIFO overrun has occurred.

UART Hardware Flow Control

The feature of the hardware flow control is fully selectable; enable users to control the serial data flow by using the io_irda_nrst output signal and io_irda_ncts input signal. Figure 8-31 shows how two similar devices can communicate with each other by using the UART hardware flow control.

Vangotech

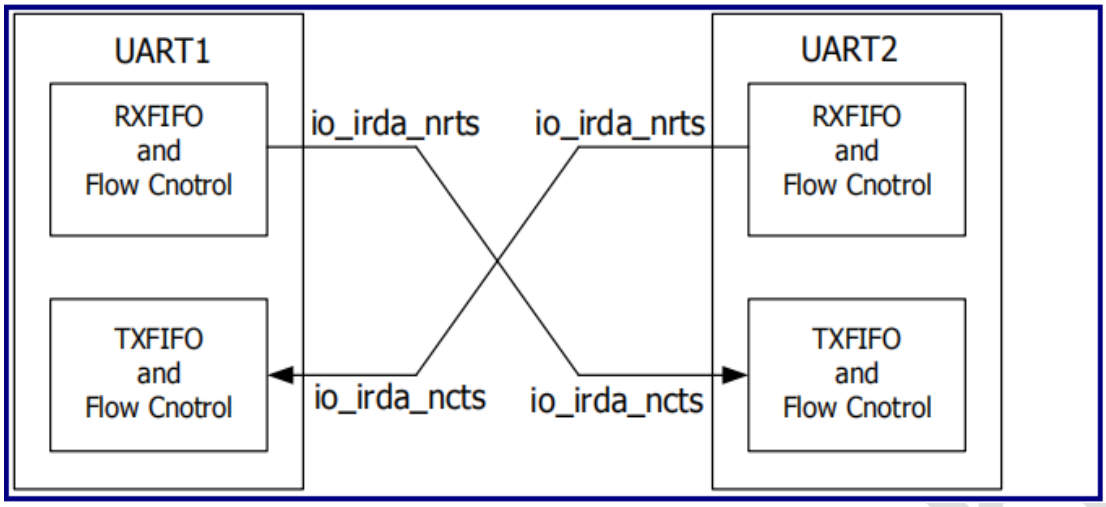


Figure 8-31. Hardware Flow Control between Two Similar Devices

When the RTS flow control is enabled, the io_irda_rts signal is asserted until the receive FIFO is filled up to the programmed trigger level. When the CTS flow control is enabled, the transmitter can only transmit data when the io_irda_cts signal is asserted.

The hardware flow control is selectable by using the RTSEn and CTSEn bits in the Interrupt Enable Register (IER, offset: 0x04). Please refer to Table 8-111 for more details on how to simultaneously and independently set the bits to enable the RTS and CTS flow controls.

RTS Flow Control

The RTS flow control logic is linked to the programmable receive FIFO trigger levels (FCR, offset:0x08).When the RTS flow control is enabled, the io_irda_nrts signal is asserted until the receive FIFO is filled up to the trigger level. When the receive FIFO trigger level is reached, the io_irda_nrts signal will be de-asserted. This indicates that there will be no space to receive more data.The transmission of data will be stopped after the current character is transmitted. The io_irda_nrts signal will be re-asserted when data are read out from the receive FIFO; therefore, it is filled to less than the trigger level.

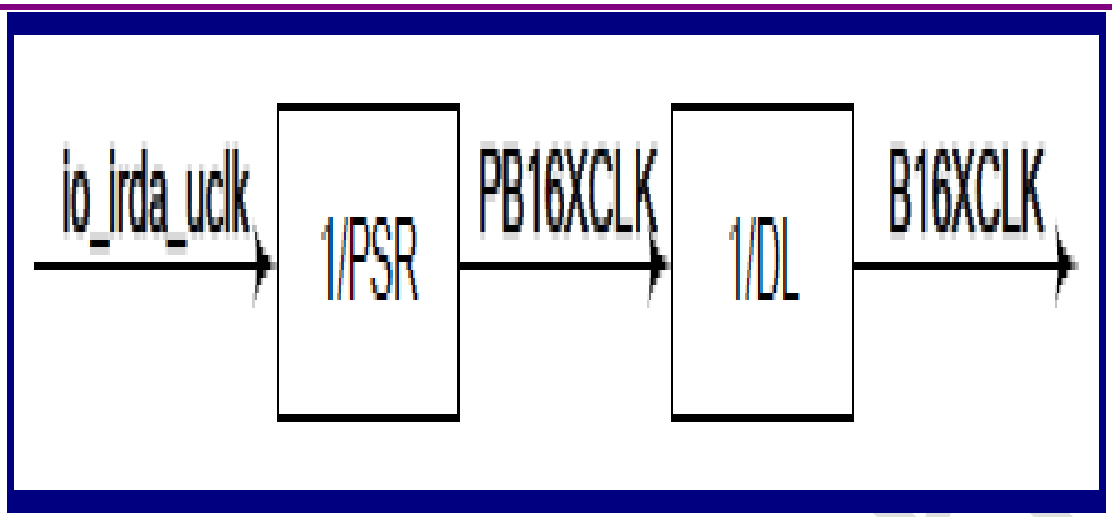
CTS Flow Control

If the CTS flow control is enabled, the transmitter will check the io_irda_ncts signal before transmitting the next byte. If the io_irda_ncts signal is asserted, the byte will be transmitted; otherwise, the transmission will not occur. The data continues transmitting while the io_irda_ncts signal is asserted and the transmit FIFO is not empty. If the transmit FIFO is empty and the io_irda_ncts signal is asserted,no data will be transmitted. If the io_irda_ncts signal is de-asserted and the CTS flow control is enabled, the current character transmission will be completed before stopping.

8.6.5. Programming Guidelines

8.6.5.1. UART Mode

1. Program the prescaler (PSR) register and DL to ensure that the output (B16XCLK) is approximately example: io_irda_uclk=1.843 MHz and the baud rate is 1.843 MHz/16 = 115.2 kbps. The default value of PSR and DL is 0x01.
2. The baud rate is given by:



$$\text{Baud rate} = \frac{f_{io\_irda\_uclk}}{PSR \times DL \times 16}$$

3. Program the Mode Definition Register (MDR) to the UART mode
4. Program the LCR register
   - Set the character
   - Set the number of stop bits or other need settings

4.1.1.3.1 UART Data Transmission Mode

The following programming steps of the register are required for data transmission in the UART mode:

1. Program FCR (Offset:0x08)
2. Write THR (Offset: 0x00) and transmit start
3. Read LSR (Offset: 0x14) to check transmitter empty and THR empty, then transmission is done.

4.1.1.3.2 UART Data Receive Mode

1. Program FCR (Offset:0x08)
2. Program the IER register to enable the data reception related interrupts:
   - Enable the Data Ready interrupt
   - Disable THR empty
3. Wait IIR (Offset: 0x08) received data ready interrupt status
4. Read RBR

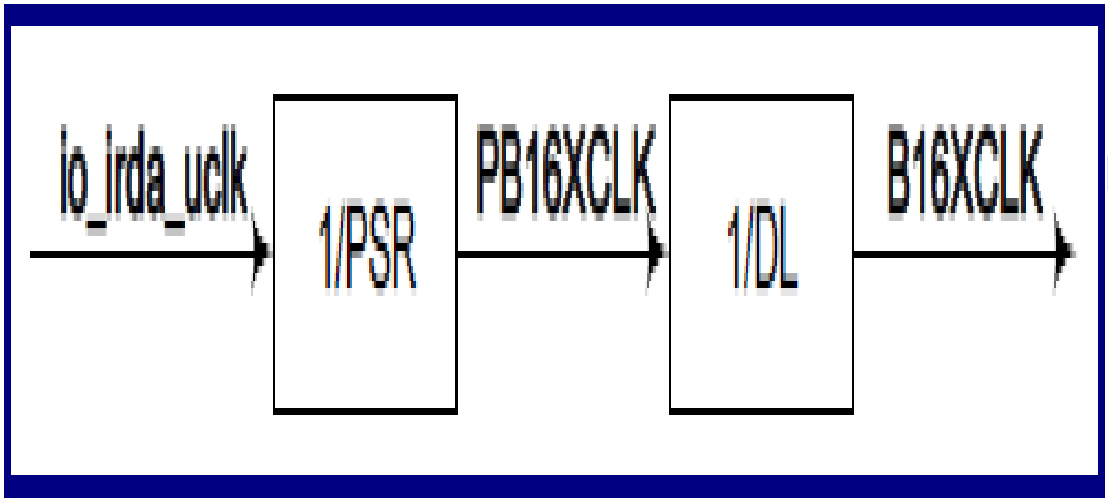
8.6.5.2. SIR Mode

The operation in the SIR mode is similar to the one in the UART mode, except that the data communication occurs in a slow-speed half-duplex manner. The baud rate in the SIR mode is up to 115.2

kbps and the following programming setting is based on the baud rate at 115.2 kbps. The method of data transfer is infrared instead of on a wired path. The Modem Control Register (MCR) and the Modem Status Register (MSR) are not used.

The following programming steps of the register are examples and required to receive or transmit data in the SIR mode:

1. Program the prescaler (PSR) register and DL to ensure that the output (B16XCLK) is approximately 1.843 MHz and the baud rate is 1.843 MHz/16 = 115.2 kbps. The default value of PSR is 0x01.
2. The baud rate is given by:



$$\text{Baud rate} = \frac{f_{io\_irda\_uclk}}{PSR \times DL \times 16}$$

3. Program the Mode Definition Register (MDR) to the SIR mode
4. Program the LCR register
   - Set the character length to 8 bits
   - Set the number of stop bits to '1'
   - Disable the parity

4.1.1.3.3 SIR Data Transmission Mode

The following programming steps of the register are required for data transmission in the SIR mode:

1. Program the FCR register.
   - Reset TX FIFO and Rx FIFO
   - Enable FIFO and set the trigger level for TX FIFO and RX FIFO
2. Program the IER register to enable only the data transmission interrupts.
   - Enable the THR Empty interrupt
   - Disable the Receiver Line Status interrupt

- Disable the Receiver Data Ready interrupt

3. Program the ACR register bit 7 to select the fixed 1.6-μs or 3/16 of the pulse width and bit 0 to enable the data transmission

4.1.1.3.4 SIR Data Receive Mode

1. Program the FCR register:
   - Reset TX FIFO and Rx FIFO
   - Enable FIFO and set the trigger level for TX FIFO and RX FIFO
2. Program the IER register to enable the data reception related interrupts:
   - Enable the Data Ready interrupt
   - Disable the THR Empty
3. Write a '1' to the ACR register bit 1 to enable the data reception

8.6.5.3. FIR Mode

4.1.1.3.5 FIR Data Transmission Mode

The following programming steps of the registers are examples and performed for transmission in the FIR mode:

1. Write a '6' to the PSR register. This divides the 48 MHz input clock by '6' and generates 8 MHz internal clock for the FIR operation. Set DLL and DLM registers to 0x0. In the FIR mode, PB16XCLK must be 8 MHz for the FIR internal logic.
2. Program the Mode Definition register (MDR) to:
   - Select the FIR mode
   - Enable/Disable the software control on the 1.6-μs SIP pulse
   - Select the frame closing method (Frame-length method or set–EOT bit method)
   - Enable/Disable the operation in the DMA mode
3. Program the PLR register to select the number of preambles, PLR[1:0]. The default value is 16.
4. If the frame-length method is selected for frame closing, program the TXLENL and TXLENH registers for the frame length. In the multi-frame transmission, LSTFMLENL and LSTFMLENH should also be properly set.
5. Program the FCR register to:
   - Reset TX FIFO
   - Select the TX FIFO trigger level
6. CPU writes a part of the frame (For a bigger frame) or the whole frame (A small frame such as a supervisory frame) in TX FIFO. This step is optional.
7. Program the FMIIER register to enable only the transmitter related interrupts and disable the remaining interrupts. Enable the following items:

   For programmed I/O mode

   - Transmitter FIFO below threshold level interrupt enable (FMIIER[1])
   - Transmitter underrun interrupt enable (FMIIER[3])
   - Frame has sent the interrupt enable (FMIIER[5]).

   For DMA mode

   - Transmitter underrun interrupt enable (FMIIER[3])
   - Frame has sent the interrupt enable (FMIIER[5]).
8. Program the FMLSIER register to enable only the transmitter-related interrupts and disable the

Vangotech

remaining interrupts.

9. Write '1' to the ACR[0] bit to enable data transmission.

4.1.1.3.6 FIR Data Receive Mode

The following register programming steps are performed for reception in the FIR mode:

1. Program the Mode Definition Register (MDR) to:
   - Select the FIR mode
   - Enable/Disable the DMA operating mode
2. Program the MRXLENL and MRXLENH registers for the maximum receive receive-frame length value. The MRXLENL register stores th e lower eight bits and the MRXLENH register stores the remaining upper bits. When the intended maximum receive receive-frame length is n, program the MRXLENL and MRXLENH registers to be n + 4.
3. Program the FCR register to:
   - Reset RX FIFO
   - Select the RX FIFO trigger level
4. Program the FMIIER register to enable only the receiver related interrupts and disable the remaining interrupts. Enable the following:

   For programmed I/O mode
   - Rx threshold interrupt (FMIIER[0] bit)
   - The Rx FIFO overrun interrupt (FMIIER[2] bit)
   - The received end of the frame interrupt (FMIIER[4] bit)

   For DMA mode
   - Status FIFO threshold interrupt disable (FMIIER[0])
   - Status FIFO timeout interrupt disable (FMIIER[1])
   - Status FIFO overrun interrupt disable (FMIIER[2])
5. Program the FMLSIER register to enable only the receiver-related interrupts and disable the remaining interrupts.
6. Write '1' to the ACR[1] bit to enable data reception

This completes the programming of the registers for the data reception and FTUART010 is ready to receive data. The device decodes the serial data, converts it from the serial data to parallel data and stores the data bytes in RX FIFO. When the stored data in RX FIFO reaches the set threshold level, the device interrupts CPU. When CPU is interrupted, it will read FMIIR to identify the source of the interrupt.

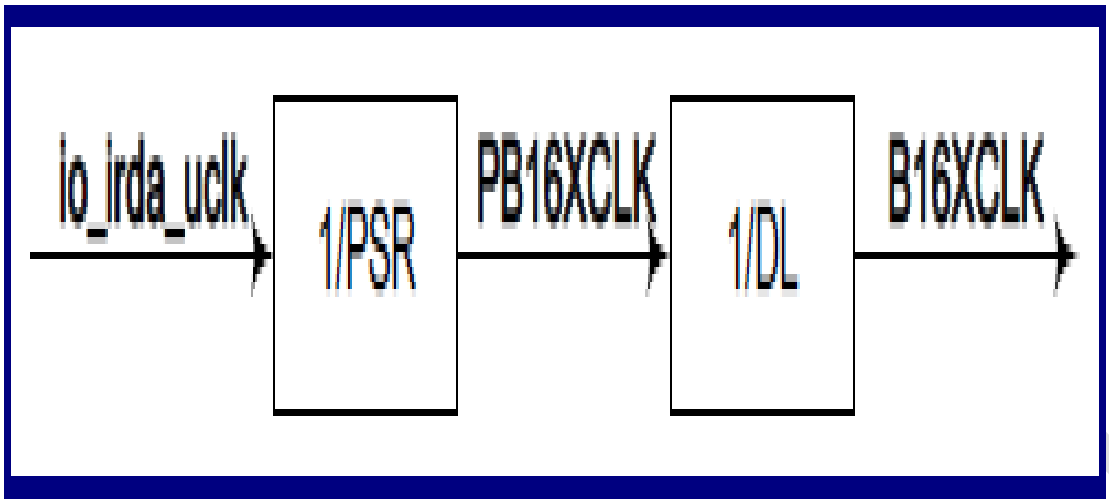
In the PIO mode, when the source of the interrupt is the FMIIR[0] bit and not the FMIIR[4] bit, CPU will go to the threshold mode.

When the source of the interrupt is the FMIIR[4] bit and not the FMIIR[0] bit, CPU will read RXFF_CNTR to determine how many bytes (Including 4-byte CRC32) related to the received frame should be read out.

If the source of the interrupt is the FMIIR[2] bit, there has been an overrun while receiving data and CPU will need to service the overrun.

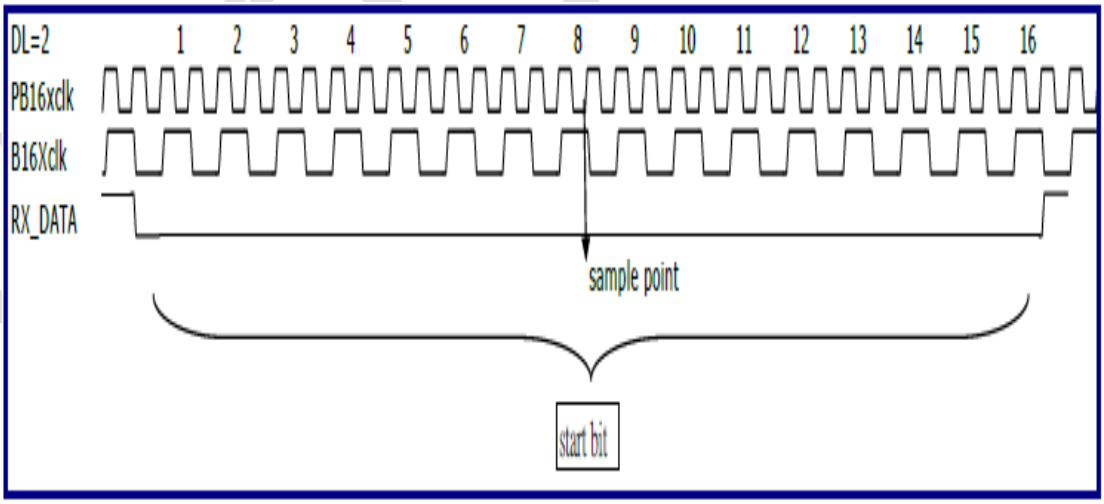
Vangotech

8.6.5.4. Clock Frequency Limitation



$$\text{Baud rate} = \frac{f_{io\_irda\_uclk}}{PSR \times DL \times 16}$$

The following figure shows the signal relationship among for io_irda_uclk, PB16XCLK, and B16XCLK, when PSR is set to '1' and DL is set to '2'. The reference clock for data transfer is B16XCLK at the 8-th rising edge if B16XCLK is the data sample point.



In the UART/SIR mode, there is a constraint on the ratio of clock frequencies for B16XCLK and PCLK. The frequency of B16XCLK must be programmed to be no more than 56 times faster than the frequency of PCLK: FPCLK is the frequency of PCLK and FB16XCLK is the frequency of B16XCLK, and the frequency constraint of PCLK and B16XCLK is shown below:

$56 * F_{PCLK} > F_{B16XCLK}$

B16XCLK is io_irda_uclk/(PSR * DL) and baud rate = B16XCLK/16.

B16XCLK is the internal clock used for the data sampling.

This allows sufficient time to move the read pointer of TX FIFO. The ratio is calculated when the shortest character, 5-bit word with no parity and one stop bit, is transmitted. It indicates that 56 * FPCLK > FB16XCLK is based on the 5-bit word with no parity case (The worst-case condition) calculate result.

Baud rate limitation:

In some applications, the ideal baud rate is not equal to the practical baud rate, like the following application.

The ideal baud rate is the specification definition and the practical baud rate is the real application data rate which is calculated by the DL and PSR registers.

The valid deviation between the ideal and practical condition is within 3.125%. The deviation is calculated by the following formula:

Deviation = [(Practical baud rate/Ideal baud rate) - 1] * 100%, "Deviation < | 3.125% |" is a valid operation.

If the deviation > 3.125%, it will cause an error data transfer.

Note: The measured maximum baud rate is 1.5 Mbps.

In the FIR mode, the criterion is 8 $F_{PCLK} > F_{PB16XCLK}$.

The operating frequency of FIR is 8 MHz; therefore, F_{PCLK} should be faster than 1 MHz.

8.6.6. Register List

There are 10 UART controllers.

Module Name	Base Address
UART_0	0x54E00000
UART_1	0x54F00000
UART_2	0x55000000
UART_3	0x55100000
UART_4	0x55200000
UART_5	0x56900000
UART_6	0x56A00000
UART_7	0x56B00000
UART_8	0X56C00000
UART_9	0X57100000

Table 8-108 Register Summary Table

Base Offset (Hex)	Name	Size (Bit)	Type	Description	Default Value (Hex)
UART/Infrared SIR Mode					
00	RBR	8	RO	Receiver Buffer Register	0x00
	THR	8	WO	Transmitter Holding Register	0x00
04	IER	4	RW	Interrupt Enable	0x00

				Register	
08	IIR	8	RO	Interrupt Identification Register	0x01
	FCR		WO	FIFO Control Register	0x00
0C	LCR	8	RW	Line Control Register	0x00
10	MCR	7	RW	Modem Control Register	0x00
14	LSR	8	RO	Line Status Register	0x60
	TST		WO	Testing Register	0x00
18	MSR	8	RO	Modem Status Register	0x00
1C	SPR	8	RW	Scratch Pad Register	0x00
Registers accessible when DLAB = 1					
00	DLL	8	RW	Baud Rate Divisor Latch Least Significant Byte	0x01
04	DLM	8	RW	Baud Rate Divisor Latch Most Significant Byte	0x00
08	PSR	5	RW	Prescaler Register	0x01
Infrared FIR Mode					
20	MDR	7	RW	Mode Definition Register	0x00
24	ACR	8	RW	Auxiliary Control Register	0x00
28	TXLENL	8	RW	Transmitter Frame Length Low	0x00
2C	TXLENH	5	RW	Transmitter Frame Length High	0x00
30	MRXLENL	8	RW	Maximum Receiver Frame Length Low	0x00
34	MRXLENH	5	RW	Maximum Receiver Frame Length High	0x00
38	PLR	2	RW	FIR Preamble Length Register	0x00
3C	FMIIR_PIO FMIIR_DMA	5	RO	FIR Mode Interrupt Identification Register in the PIO mode	0x00
			RO	FIR Mode Interrupt Identification Register in the DMA mode	0x00
40	FMIIR_PIO FMIIR_DMA	5	RW	FIR Mode Interrupt Enable Register for the PIO mode	0x00
			RW	FIR Mode Interrupt Enable Register in the	0x00

Vangotech

				DMA mode	
44	STFF _STS	5	RO	Status FIFO Line Status Register	0x00
48	STFF_RXLENL	8	RO	Status FIFO Received Frame Length Register Low	0x00
4C	STFF_RXLENH	5	RO	Status FIFO Received Frame Length Register High	0x00
50	FMLSR	8	RO	FIR Mode Link Status Register	0xC3
54	FMLSIER	8	RW	FIR Mode Link Status Interrupt Enable Register	0x00
5C	RXFF_CNTR	5	RO	Rx FIFO Count Register	0x00
60	LSTFMLENL	8	RW	Last Frame Length Register Low	0x00
64	LSTFMLENH	8	RW	Last Frame Length Register High	0x00
68	FEATURE	8	RO	Feature Register	0x11
6C	REVD1	8	RO	Revision Register Digit 1	1
70	REVD2	8	RO	Revision Register Digit 2	0x11
74	REVD3	8	RO	Revision Register Digit 3	0

8.6.7. Register Description

8.6.7.1. Receiver Buffer Register

The user can get the data by reading this read-only location. It is the data read port of RX FIFO.

- If FIFOs are enabled:

  Once a character has been assembled at the receive shift register, the contents will be written to RXFIFO. This RBR refers to RXFIFO location which the current read pointer indicates.
- If FIFOs are not enabled:

  Once a character has been assembled at the receive shift register, the contents will be written to RXFIFO. This RBR refers to RXFIFO location which the read pointer "0" indicates.

Table 8-109 Receiver Buffer Register (RBR, Offset: 0x00 for Read)

Bit	Name	Type	Description	Default Value
[7:0]	RBR	R	Receive Data Port	0

8.6.7.2. Transmit Holding Register

Users can write data to the transmitter holding register. It is the data write port of the TX FIFO.

- If FIFOs are enabled:

  Once the transmit shift register is empty, the contents of THR will be written to the transmit shift

register for the transmitted data shift out. This THR refers to TXFIFO location which the current read pointer indicates.

- If FIFOs are not enabled:

  Once the transmit shift register is empty, the contents of THR will be written to the transmit shift register for the transmitted data shift out. This THR refers to TXFIFO location which the read pointer "0" indicates.

If the transmitted character width is less than eight bits, it must be right-justified. Left bits (i.e. MSB) are "don't care" bits. For example, with a word length of five bits, writing 0xd3 or 0xf3 will result in the transmission of a 13h character. Before writing this register, the user must ensure that FTUART010 is ready to accept data for transmission, for example, checking if the THR Empty flag is set in LSR (Please refer to the description of this register in the following table.

Table 8-110 Transmitter Holding Register (THR, Offset: 0x00 for Write)

Bit	Name	Type	Description	Default Value
[7:0]	THR	WO	Transmit Data Port	0

8.6.7.3. Interrupt Enable Register

This register individually enables each of the possible interrupt sources. A '1' in any of these bits enables the corresponding interrupt, while a '0' disables the corresponding interrupt. For a detailed description of the interrupt sources, please refer to the description of the Interrupt Identification Register (IIR) in the following table.

Table 8-111 Interrupt Enable Register (IER, Offset: 0x04)

Bit	Name	Type	Description	Default Value
7	DSREn	RW	DSR flow control enabled 1: Enable the DSR hardware flow control 0: Disable the DSR hardware flow control When DSREn is set to '1', the loopback mode must not be set to '1' and FIFO must be enabled (FCR[0]).	0
6	DTREn	RW	DTR flow control enabled 1: Enable the DTR hardware flow control 0: Disable the DTR hardware flow control When DTREn is set to '1', the loopback mode must not be set to '1' and FIFO must be enabled (FCR[0]).	0
5	CTSEn	RW	CTS flow control enabled 1: Enable the CTS hardware flow control 0: Disable the CTS hardware flow control When CTSEn is set to '1', the loopback mode must not be set to '1' and FIFO must be enabled (FCR[0]).	0
4	RTSEn	RW	RTS flow control enabled 1: Enable the RTS hardware flow control 0: Disable the RTS hardware flow control	0

Vangotech

			When RTSEn is set to '1', the loopback mode must not be set to '1' and FIFO must be enabled (FCR[0]).	
3	MODEM Status	RW	This bit enables the modem status interrupt when set to logic 1.	0
2	Receiver Line Status	RW	This bit enables the Receiver Line Status Interrupt when set to logic 1.	0
1	THR Empty	RW	This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.	0
0	Receiver Data Available	RW	This bit enables the Received Data Available Interrupt (And the character reception timeout interrupts in the FIFO mode) when set to logic 1.	0

8.6.7.4. Interrupt Identification Register

The main purpose of this register is to identify the interrupt with the highest priority that is currently pending. FTUART010 implements a four-level priority encoder from the highest priority to the lowest priority, as follows:

1. Receive Line Status (Highest priority)
2. Receive Data Ready and Character Reception Timeout (Second priority)
3. Transmitter Holding Register Empty (Third priority)
4. Modem Status (Lowest priority)

The following table describes different interrupt conditions, codes of identification, and reset method.

Table 8-112 Interrupt Identification Register (IIR , Offset: 0x08)

Bit	Name	Type	Description	Default Value
[7:6]	FIFO mode enable	R	These two bits are set when FCR[0] is set to '1'.	0
5	Reserved	R	This bit of the IIR is always '0'.	0
4	Tx FIFO full	R	This bit is set to '1' when TX FIFO is full.	0
3	FIFO mode only	R	In the 16450 mode, this bit is 0. In the FIFO mode, this bit is set along with bit 2 when a timeout interrupt is pending.	0
[2:1]	Interrupt Identification Code	R	These bits identify the highest priority interrupt that is pending. Table 8-113 lists and describes different interrupt conditions, the codes of identification, and the reset method. Please note that for an interrupt source to be considered as pending, the corresponding bit in the IER must be enabled.	0
0	Interrupt Pending	R	This bit can be used in a prioritized interrupt environment to indicate whether an interrupt is pending. 0: An interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service	1

			routine. 1: No interrupt is pending.	

Table 8-113 Interrupt Control Table

Interrupt Identification Register					Interrupt Set and Reset Function		
Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Priority Level**	**Interrupt Type**	**Interrupt Source Description**	**Interrupt Reset Method**
0	0	0	1	-	None	There is no interrupt pending.	None
0	1	1	0	Highest	Receiver Line Status	There is an overrun error, parity error, framing error, or break interrupt indication corresponding to the received data on top of the receive FIFO. Please note that the FIFO error flag in LSR does not influence this interrupt, which is related only to the data on top of Rx FIFO. This is directly related to the presence of a 1 in any of the LSR bits 1 to 4.	Read the Line Status Register (LSR)
0	1	0	0	Second	Received Data Ready	In the non-FIFO mode, there is received data available in the RHR register. In the FIFO mode, the number of characters in the receive FIFO is equal to or greater than the trigger level programmed in FCR. The interrupt signal will stay active while the number of words in FIFO stays higher than that value and will be cleared when the microprocessor reads the necessary words to make the number of words in	Read the Receiver Buffer Register (RBR)

Vangotech

						FIFO less than the trigger level. Please note that this is not directly related to LSR bit 0, which always indicates that there is at least one word ready.	
1	1	0	0	Second	Character Reception Timeout	There is at least one character in the receive FIFO and during a time corresponding to four characters at the selected baud rate, no new character has been received. A FIFO timeout interrupt will occur, if the following conditions exist: 1. At least one character is in FIFO. 2. The most recent serial character received was longer than 4 continuous character times ago (if 2 stop bits are programmed, the second one is included in this time delay). 3. The most recent CPU read of FIFO was longer than four continuous characters time ago.	Read the Receiver Buffer Register (RBR) Alternatively, in the UART mode, if Character Reception Timeout is asserted, reset RX FIFO can clear it.
0	0	1	0	Third	Transmitter Holding Register Empty	In the non-FIFO mode, the 1-byte THR is empty. In FIFO mode, the complete 16-byte transmit FIFO is empty, so 1 to 16 characters can be written to THR. That is to say, THR Empty bit in LSR is one.	Write the Transmitter Holding Register (THR) Alternatively, reading the Interrupt Identification Register (IIR) will also clear the

Vangotech

							interrupt if this is the interrupt type being currently indicated (this will not clear the flag in the LSR).
0	0	0	0	Fourth	Modem Status	A change has been detected in the Clear To Send (CTS), Data Set Ready (DSR) or Carrier Detect (CD) input lines or a trailing edge in the Ring Indicator (RI) input line. That is to say, at least one of MSR bits 0 to 3 is one.	Read the Modem Status Register (MSR)

8.6.7.5. FIFO Control Register

This is a write-only register at the same location as IIR (The IIR is a read-only register). This register is used to enable and clear FIFOs, and set the RX FIFO trigger level. If users reset FIFO, the status of FIFO will not clear. Users must clear these registers by reading the corresponding registers.

Table 8-114 FIFO Control Register (FCR, Offset: 0x08 for Write)

Bit	Name	Type	Description	Default Value
[7:6]	RXFIFO_TRGL	WO	Set the trigger level of the RXFIFO interrupt.Please refer to Table 8-115 for details.	0
[5:4]	TXFIFO_TRGL	WO	Set the trigger level of TXFIFO interrupt.Please refer to Table 8-116 for details.	0
3	DMA Mode	WO	This bit selects the UART DMA mode. The DMA mode affects the way in which the DMA signaling outputs pins (irda_nrxrdy and irda_ntxrdy) behave.	0
2	TX FIFO Reset	WO	Sett this bit to logic 1 clears all bytes in TX FIFO and resets the counter logic to 0. The shift register is not cleared, so any reception active will continue. This bit will automatically return to zero.	0
1	RX FIFO Reset	WO	Set this bit to logic 1 clears all bytes in Rx FIFO and resets the counter logic to 0. The shift register is not cleared, so any reception active will continue. Set this bit also clears the Status FIFO.	0

Vangotech

			This bit will automatically return to zero.	
0	FIFO Enable	WO	Set this bit to logic 1 enables both the transmit and receive FIFOs (As well as the Status FIFO). Change this bit will automatically reset both FIFOs. In the FIR mode, the device driver should always set this bit to '1'.	0

Table 8-115 FIFO Trigger Level of Receiver

FCR Code		**16-byte Receiver FIFO Trigger Level**	**32-byte Receiver FIFO Trigger Level**
Bit 7	**Bit 6**		
0	0	1 character	1 character
0	1	4 characters	8 characters
1	0	8 characters	16 characters
1	1	14 characters	28 characters
FCR Code		**64-byte Receiver FIFO Trigger Level**	**128-byte Receiver FIFO Trigger Level**
Bit 7	**Bit 6**		
0	0	1 character	1 character
0	1	16 characters	32 characters
1	0	32 characters	64 characters
1	1	56 characters	120 characters

Table 8-116 FIFO Trigger Level of Transmitter

FCR Code		**16-byte Transmitter FIFO Trigger Level**	**32-byte Transmitter FIFO Trigger Level**
Bit 5	**Bit 4**		
0	0	1 character	1 character
0	1	3 characters	6 characters
1	0	9 characters	18 characters
1	1	13 characters	26 characters
FCR Code		**64-byte Transmitter FIFO Trigger Level**	**128-byte Transmitter FIFO Trigger Level**
Bit 5	**Bit 4**		
0	0	1 character	1 character
0	1	16 characters	32 characters
1	0	32 characters	64 characters
1	1	56 characters	120 characters

8.6.7.6. Line Control Register

This register controls the way in which the transmitted characters are serialized and received characters are assembled and checked.

Table 8-117 Line Control Register (LCR, Offset: 0x0C)

Bit	**Name**	**Type**	**Description**	**Default Value**

7	DLAB	RW	Divisor Latch Access Bit (DLAB) This bit must be set in order to access the DLL, DLM and PSR registers which program the division constants for the baud rate divider and the prescaler.	0
6	Set Break	RW	This bit causes a break condition to be transmitted to the receiving UART. When it is set to logic 1, the serial output (**io_irda_sout**) is forced to the Spacing (Logic 0) state. The break is disabled by setting bit 6 to 0. The Break Control bit acts only on **io_irda_sout** and has no effect on the transmitter logic, so if several characters are stored in the transmit FIFO, they will be removed from this FIFO and passed sequentially to the Transmitter Shift Register which serializes them, even if Set Break is set. This fact can be useful to establish the break time making use of the THR Empty and Transmitter Empty flags of the LSR. Firmware can follow the sequence below to assure no erroneous or extraneous characters will be transmitted because of the break: Set break when transmitter is idle (LSR bit 6). Write a character with any value to THR. Wait for the transmitter to become idle (LSR bit 6), and clear break when normal transmission has to be restored.	0
5	Stick Parity	RW	If bits[5:3] are logic 1, the Parity bit will be transmitted and checked as '0'. If bit 3 and bit 5 are '1' and bit 4 is '0', then the Parity bit will be transmitted and checked as '1'. If bit 5 is '0', Stick Parity will be disabled. Please refer to Table 8-118 for details.	0
4	Even Parity	RW	This bit is the Even Parity Select bit. If bit 3 is '1' and bit 4 is '0', an odd number of logic 1s will be transmitted or checked in the data word bits and Parity bit. Please refer to Table 8-118 for details.	0
3	Parity Enable	RW	This bit is the Parity Enable bit. If this bit is a '1', a Parity bit will be generated (Transmit data) or checked (Receive data) between the last data word bit and Stop bit of the serial data. If bit 3 is '1' and bit 4 is a '1', an even number of 1s will be transmitted or checked. Please refer to Table 8-118 for details. Note: If a parity error occurs, the parity error will be persistent until a correct parity is received. If the error parity is received continuously in receive pin, and users want to discard the parity error status, users can set FTUART010 in the internal loopback mode (0x10[4] = '1') and send a correct parity bit. After the correct parity is sent and is received via the loopback mode, the parity enable can be disabled. Then, parity error check is disabled, and the parity error status can be cleared via	0

Vangotech

			LSR (0x14) read.	
2	Stop Bits	RW	This bit selects the number of stop bits to be transmitted. If cleared, only one stop bit will be transmitted. If set, two stop bits (1.5 with 5-bit data) will be transmitted before the start bit of the next character. The receiver always checks only one stop bit. Please refer to Table 8-119 for details.	0
1	WL1	RW	This bit along with WL0 defines the word length of the data being transmitted and received. Please refer to Table 8-119 for the possible selections.	0
0	WL0	RW	This bit along with WL1 defines the word length of the data being transmitted and received. Please refer to Table 8-119 for the possible selections.	0

Table 8-118 Parity Setting Table

LCR Code			Parity Bit (Transmitted or Checked)
Bit 5 Stick Parity	**Bit 4 Stick Parity**	**Bit 3 Stick Parity**	
X	X	0	Not transmitted or checked
0	1	1	Even parity
0	0	1	Odd parity
1	0	1	1
1	1	1	0

Table 8-119 Word Length and Stop Bit Setting

LCR Code			Character Length (Bits)	Number of Stop Bits
Bit 2 Stop Bit	**Bit 1 WL1**	**Bit 0 WL0**		
0	0	0	5	1
	0	1	6	
	1	0	7	
	1	1	8	
1	0	0	5	1.5
	0	1	6	2
	1	0	7	
	1	1	8	

8.6.7.7. Modem Control Register

By writing this register, user can set the modem control outputs (io_irda_ndtr and io_irda_nrts). This register also controls the loopback mode, and provides the general purpose outputs.

Table 8-120 Modem Control Register (MCR, Offset: 0x10)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	-	0

Vangotech

6	Out3	RW	This bit controls the general purpose, active low, output io_irda_nout3, in the same way as bit 0 controls io_irda_ndtr. This bit is not found in the standard 16550 UART.	0
5	DMAmode2	RW	This bit selects the UART/SIR DMA mode. The DMA mode2 affects the way in which the DMA signaling output pins, irda_nrxrdy and irda_ntxrdy, behave.	0
4	Loop	RW	Loopback mode control bit Loopback mode is intended to test the UART or SIR communication.	0
3	Out2	RW	This bit controls the general purpose, active low, output io_irda_nout2, in the same way as bit 0 controls io_irda_ndtr.	0
2	Out1	RW	This bit controls the general purpose, active low, output io_irda_nout1, in the same way as bit 0 controls io_irda_ndtr.	0
1	RTS (Request to Send)	RW	This bit controls "request to send", active low output, io_irda_nrts, in the same way as bit 0 controls io_irda_ndtr.	0
0	DTR (Data Terminal Ready)	RW	This bit controls "data terminal ready", active low output, io_irda_ndtr. A 1 in this bit makes io_irda_ndtr output a '0'. When this bit is cleared, io_irda_ndtr outputs a '1'.	0

When this UART is set in loopback mode, the following items occur:

- The serial output is internally connected to the serial input; therefore, every character sent is looped back and received.
- The input pin, io_irda_sin, is not used and the output pin, io_irda_sout, is set to '1' (Inactive state).
- The four modem control inputs are internally connected to the two modem control outputs plus the general purpose outputs. This way, io_irda_ncts is internally controlled by io_irda_nrts, io_irda_ndsr is internally controlled by io_irda_ndtr, io_irda_nri is internally controlled by io_irda_nout1, and io_irda_ndcd is internally controlled by io_irda_nout2. There is a non-ordered correspondence between the four least significant bits of MCR and the four most significant bits of MSR. Please refer to Figure 8-32 for the connections. The modem control output pins are forced to the inactive state (High).
- The four modem control input pins, io_irda_ncts, io_irda_ndsr, io_irda_nri and io_irda_ndcd, are not used. The two modem control output pins, io_irda_ndtr and io_irda_nrts, and the three user outputs, io_irda_nout1, io_irda_nout2, and io_irda_nout3, are set to '1' (Inactive state).

Vangotech

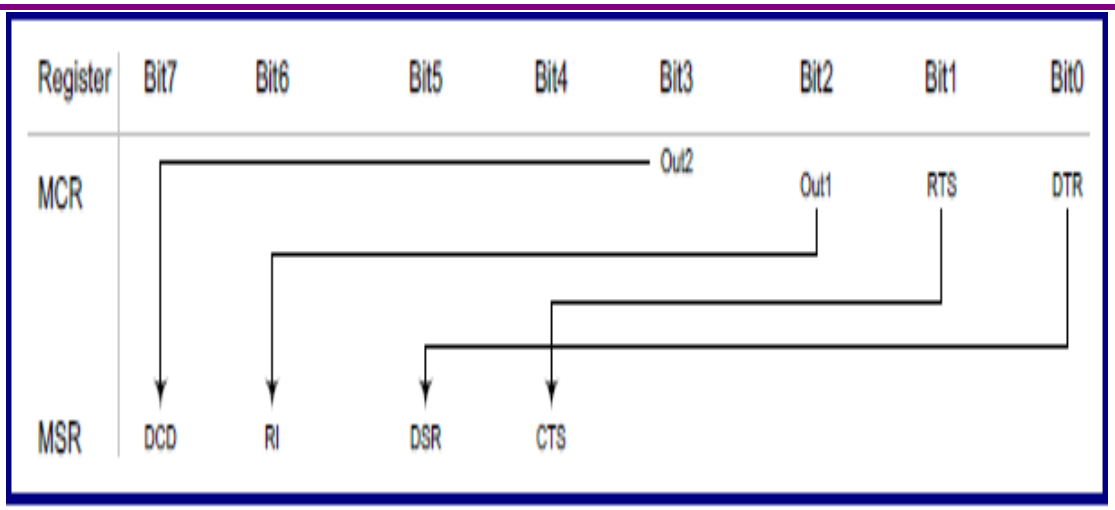


Figure 8-32. Interconnection between MCR and MSR in Loopback Mode

8.6.7.8. Line Status Register

This register informs users of the status of the transmitter and the receiver. In order to acquire the information about a received character, LSR must be read before reading the received character from RBR.

Table 8-121 Line Status Register (LSR, Offset: 0x14 for Read)

Bit	Name	Type	Description	Default Value
7	FIFO Data Error	RO	If FIFO is disabled (16450 mode), this bit will always be zero. If FIFO is active, this bit will be set as soon as any data character in the receive FIFO has parity or framing error or the break indication is active. This bit will be cleared when CPU reads LSR and the rest of the data in the receive FIFO do not have any of these three associated flags on.	0
6	Transmitter Empty	RO	This bit will be '1' when both THR (Or TX FIFO) and TSR (Transmitter Shift Register) are empty. Reading this bit as '1' indicates that no transmission will currently occur in the io_irda_sout output pin, and that the transmission line is idle. As soon as new data is written in THR, this bit will be cleared.	1
5	THR Empty	RO	This bit indicates that UART is ready to accept a new character for transmission. In addition, this bit will cause UART to issue an interrupt to CPU when the Transmit Holding Register Empty Interrupt enable bit (IER [1]) is set high. In the non-FIFO mode, this bit is set whenever the 1-byte THR is empty. If THR holds data to be transmitted, this bit will be immediately set when this data is passed to TSR. In the FIFO mode, this bit is set when the transmitter FIFO is completely empty, being 0 if there is at least one byte in FIFO waiting to be passed to the TSR for transmission.	1
4	Break	RC	This bit is set to '1' if the receiver line input, io_irda_sin, is	0

	Interrupt		held at zero for a complete character time. This indicates that the positions correspond to the start bit, data, parity bit (If any), and (First) stop bit are all detected as zeroes. Please note that a Framing Error flag always accompanies this flag. This bit is queued in the receiver FIFO in the same way as the Parity Error bit. When the break occurs, only one zero character is loaded into FIFO. The next character transfer is enabled after io_irda_sin goes to the marking state and receives the next valid start bit. This bit is cleared as soon as LSR is read.	
3	Framing Error	RC	This bit indicates that the received character did not have a valid stop bit (i.e., a '0' is detected in the (First) stop bit position instead of a '1'). This bit is queued in the receiver FIFO in the same way as the Parity Error bit. When a framing error is detected, the receiver tries to resynchronize; if the next sample is again a zero, it will be taken as the beginning of a possible new start bit. This bit is cleared as soon as LSR is read.	0
2	Parity Error	RC	When this bit is set, it indicates that the parity of the received character is wrong according to the current setting in LCR. This bit is queued in the receive FIFO; therefore, it is associated with the particular character that had the error. LSR must be read before RBR: Each time a character is read from RBR, the next character passes to the top of FIFO and LSR is loaded with the queued error flags corresponding to this top-of-the-FIFO character. This bit is cleared as soon as LSR is read.	0
1	Overrun Error	RC	When this bit is set, a character will be completely assembled in the Receiver Shift Register without having free space to put it in the receive FIFO or buffer register. When an overrun condition appears, the result is different depending on whether the 16-byte FIFO is active or not. If FIFO is inactive, only a 1-character Receiver Buffer Register will be available, and the unread data in this RBR will not be overwritten with the new character just received. If FIFO is active, the character just received in the Receiver Shift Register will be overwritten, but the data already present in FIFO will not be changed. The Overrun Error flag is set as soon as the overrun condition appears. This bit will not be queued in FIFO if it is active. This bit will be cleared as soon as LSR is read.	0
0	Data Ready	RO	This bit is set if one or more characters have been received	0

			and are waiting in the receiver FIFO for users to read. It is cleared to logic 0 by reading all of the data in the Receiver Buffer Register or FIFO.	

8.6.7.9. Testing Register

This register provides internal diagnostic capabilities if the circuit is hardware implemented. The loopback mode is supported only in the UART and SIR modes. For FIR, because there is only one CRC module in FTUART010, the FIR transmit and receive cannot be tested with the loopback mode.

Table 8-122 Testing Register (TST, Offset: 0x14 for Write)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd		0
4	TEST_CRC_ERR	WO	When set, FTUART010 generates incorrect CRC during the FIR transmission.	0
3	TEST_PHY_ERR	WO	When set, FTUART010 generates incorrect 4PPM encoding chips during the FIR transmission.	0
2	TEST_BAUDGEN	WO	This bit is used to improve the baud-rate generated toggle rate.	0
1	TEST_FRM_ERR	WO	When set, FTUART010 generates a logic 0 STOP bit during the UART transmission.	0
0	TEST_PAR_ERR	WO	When set, FTUART010 generates incorrect parity during the UART transmission.	0

8.6.7.10. Modem Status Register

This register provides information about the status of the four modem control input pins. The four most significant bits directly provides the status of the pin, while the four least significant bits give information about the changes in these pins.

The four least significant bits can generate an interrupt (Modem Status interrupt) if enabled by the corresponding bit in IER. The interrupt will be generated as soon as any of them is '1'. They are reset to logic 0 whenever the Modem Status Register is read.

Table 8-123 Modem Status Register (MSR, Offset: 0x18)

Bit	Name	Type	Description	Default Value
7	DCD	RO	Data Carrier Detect (DCD), which is the complement of the io_irda_ndcd input.	0
6	RI	RO	Ring Indicator (RI), which is the complement of the io_irda_nri input.	0
5	DSR	RO	Data Set Ready (DSR), which is the complement of the io_irda_ndsr input.	0
4	CTS	RO	Clear To Send (CTS), which is the complement of the io_irda_ncts input.	0

3	Delta DCD	RO	The delta-DCD flag. If set, it means that the io_irda_ndcd input has changed since the last time the microprocessor read this bit.	0
2	Trailing edge R1	RO	This bit is set when a trailing edge is detected in the io_irda_nri input pin; that is to say, when io_irda_nri changes from 0 to 1.	0
1	Delta DSR	RO	If set, it means that the io_irda_ndsr input has changed since the last time the microprocessor read this bit.	0
0	Delta CTS	RO	If set, it means that the io_irda_ncts input has changed since the last time the microprocessor read this bit.	0

8.6.7.11. Scratch Pad Register

Table 8-124 Scratch Pad Register (SPR, Offset: 0x1C)

Bit	Name	Type	Description	Default Value
[7:0]	User Data	RW	This 8-bit read/write register has no effect on the operation of the Serial Port. It is intended as a scratchpad register to be temporarily used by the programmer to hold data.	0

8.6.7.12. Baud-Rate Divisor Latch

The Divisor Latch is a 16-bit register, whose most significant byte is held in DLM and least significant byte is held in DLL. The division factors can be programmed from 1 to 65535. The access to these two registers, located at addresses 1 and 0, respectively, is conditioned on the value of the DLAB bit in the LCR register. The two registers can be written and read only if this bit is '1'. Otherwise, IER, RBR and THR will be accessed.

These two registers, together with the Prescaler Register (PSR) select the speed at which the communication will occur. This is the baud rate at which characters will be transmitted and the expected baud rate for the characters that will be received. Only one baud rate is defined for both transmission and reception.

The baud rate is defined as the io_irda_uclk frequency divided by 16, divided by the contents of the PSR register, and then divided by the contents of the Divisor Latch. When DLM and DLL are programmed as '0', there is no output clock. It is recommended programing DLL and DLM as '0' for the power-saving in the FIR mode.

Table 8-125 Baud Rate Divisor Latch LSB (DLL, Offset: 0x00 when DLAB = 1)

Bit	Name	Type	Description	Default Value
[7:0]	DLL	RW	Baud Rate Divisor Latch Least Significant Byte	1

Table 8-126 Baud Rate Divisor Latch MSB (DLM, Offset: 0x04 when DLAB = 1)

Bit	Name	Type	Description	Default Value
[7:0]	DLM	RW	Baud Rate Divisor Latch Most Significant Byte	0

8.6.7.13. Prescaler Register

This 5-bit register (PSR[4:0]) adds a second programmable division factor to obtain the desired baud rate (Please refer to the description of the Divisor Latch register above). The division factor is the value hold in this register, so the maximum factor is 31 and the minimum is 0. Bits[7:5] are always zero. This is a non-standard register (i.e., it is not present in the industry standard 16550 UART). The input clock, io_irda_uclk, is divided by integers from 1 to 31. When PSR is a '0', there is no input clock to divisor latch unit. Therefore, programming DLL and DLM is useless when PSR is set as 0. The default value for the PSR register is 1.

This register is only accessible when the DLAB bit in LCR is set. Otherwise, the Line Status Register will be accessed.

Table 8-127 Prescaler Register (PSR, Offset: 0x08 when DLAB = 1)

Bit	Name	Type	Description	Default Value
7:5	-	Rsvd	Reserved	0
4:0	PSR	RW	Prescaler Value	1

8.6.7.14. Mode Definition Register

This register is used to select the operating mode.

Table 8-128 Mode Definition Register (MDR, Offset: 0x20)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	Reserved	0
6	IR_INV_TX	RW	When set, FTUART010 generates the inverted FIR or SIR pulse during transmission.	0
5	FIR_INV_RX	RW	This bit is to support the optical transceivers with the receive signals of the opposite polarity (Active high instead of active low). When set to '1', an inverter is placed on the receiver input signal path. In the SIR mode, this bit is useless.	0
4	DMA_EN	RW	When set to '1', the DMA mode of operation is enabled. When data transfers are performed by a DMA controller transmit and/or receive data, interrupts in the PIO mode should be disabled to avoid the spurious interrupts.	0
3	FMEND_MD	RW	0: Apply the Frame-Length Counter method 1: Apply the Set End of Transmission bit method	0
2	SIP_BYCPU	RW	0: The controller in transmission mode always sends a 1.6-μs pulse at the end of a transmission frame when TX FIFO is empty. 1: The transmission of a 1.6-μs pulse depends on the ACR[4]. CPU should keep a timer and set the ACR[4] bit at least once in every 500 ms.	0

Vangotech

[1:0]	MODE_SEL	RW	**Bit 1**	**Bit 0**	**Operational Mode**	0
			0	0	UART (Default)	
			0	1	SIR	
			1	0	FIR	
			1	1	Reserved	

8.6.7.15. Auxiliary Control Register

Table 8-129 Auxiliary Control Register (ACR, Offset: 0x24)

Bit	Name	Type	Description	Default Value
7	SIR_PW	RW	SIR pulse duration selection This bit determines the selection of either a 1.6-μs pulse duration or 3/16 of the baud-rate pulse width. 0: 3/16 1: 1.6 μs	0
[6:5]	STFF_TRGL	RW	Set the trigger level for the status FIFO interrupt This setting is only effective in the FIR mode. Please refer to Table 8-130 for details.	0
4	SEND_SIP	RW	When MDR bit 2 equals to a '1' and CPU writes a '1' to this bit while FIR is transmitting, an SIR Interaction Pulse (SIP) occurs at the end of a transmission frame when TX FIFO is empty.When CPU writes a 1 to this bit while FIR is idle, an SIP will be emitted immediately. This bit is cleared automatically by the controller at the end of the 1.6-μs IR pulse data transmission.	0
3	FORCE_ABORT	RW	In the FIR mode, CPU can intentionally abort the data transmission of a frame by writing a '1' to this bit. The default value is a '0'. Neither the end flag nor the CRC bits are appended to the frame. CPU must reset TX FIFO and clear this bit to '0' before transmitting next frame.	0
2	SET_EOT(End of thetransmission)	RW	In the FIR mode, CPU writes a '1' to this bit j before writing the last byte to Tx FIFO register. This bit is automatically cleared to '0' in the next CPU write of the last byte to FTUART010. When CPU sets this bit,	0

Vangotech

			TX_ENABLE should also be set to '1' to keep transmitting the last byte. The MDR bit 4 cannot be set to '1' when SET_EOT is logic 1.	
1	RX_ENABLE	RW	This bit is set to '1' to enable the data reception in all the IR modes.	0
0	TX_ENABLE	RW	This bit is set to '1' to enable the data transmission in all the IR modes.	0

Table 8-130 Status FIFO Trigger Level

ACR Code		Status FIFO Trigger Level
Bit 6	Bit 5	
0	0	1
0	1	4
1	0	7
1	1	8

8.6.7.16. Transmit Frame Frame-Length Register

The TXLENL register along with the TXLENH register stores the value for the number of bytes of a data frame to be transmitted. The TXLENL register stores the lower eight bits and the TXLENH register stores the upper bits.

The value of the frame length does not include any appended CRC bytes. The zero value is reserved and must not be used. Values from 1 to 213 - 5 can be used. To properly program the TXLEN register, CPU must always write the lower value into TXLENL first and then the upper value into TXLENH.

The upper three bits of TXLENH are reserved and must be written with 0's.

Table 8-131 Transmit Frame-Length Register Low (TXLENL, Offset: 0x28)

Bit	Name	Type	Description	Default Value
[7:0]	TXLENL	RW	Transmitter Frame Length Low	0

Table 8-132 Transmit Frame-Length Register High (TXLENH, Offset: 0x2C)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd	-	0
4:0][	TXLENH	RW	Transmitter Frame Length High	0

8.6.7.17. Maximum Receiver Frame Frame-Length

While receiving data, the maximum length of a frame is limited to the value written to the MRXLENL and the MRXLENH registers. The MRXLENL register stores the lower 8 bits and the MRXLENH register stores the upper bits. Any frame greater than the set maximum value will be reported as a size error in FMIIR. The maximum frame length value includes the 4-byte CRC field. Values from 5 to 2^{13} - 1 can be used. The values from 0 to 4 are reserved and must not be used.

Table 8-133 Maximum Receiver Frame-Length Register Low (MRXLENL, Offset: 0x30)

Bit	Name	Type	Description	Default Value

[7:0]	MRXLENL	RW	Maximum receiver frame-length at level low	0

Table 8-134 Maximum Receiver Frame-Length Register High (MRXLENH, Offset: 0x34)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd	-	0
[4:0]	MRXLENH	RW	Maximum receiver frame-length at level high	0

8.6.7.18. Preamble Length Register (PLR , Offset: 0x38)

The PLR specifies the number of the preambles for FIR. At 4 Mbps (FIR), the FPL count number of additional PA bytes is inserted at the start of every frame.

Table 8-135 Preamble Length Register

Bit	Name	Type	Description	Default Value
[7:2]	-	Rsvd	-	0
[1:0]	FPL	RW	Number of Preambles for the FIR frames Please refer to Table 8-136 for details.	0

Table 8-136 Number of FIR Preambles

FPL[1:0]		FIR Preamble Times
0	0	16 (Default)
0	1	4
1	0	8
1	1	32

8.6.7.19. Interrupt Identification Register in FIR Mode

This register has two different definitions depending on whether the current data transfer mode is in the PIO or the DMA mode. All status bits reflect the raw status, no matter the corresponding enable bit is set or not.

Table 8-137 Interrupt Identification Register in FIR Mode (FMIIR_PIO, Offset: 0x3C, PIO Mode)

Bit	Name	Type	Description	Default Value
[7:6]	-	Rsvd	-	0
5	FRM_SENT	RC	This bit will be set under the following circumstance: If the Set-EOT bit method is applied: Whenever a frame is transmitted, this bit will be set. If the Frame-Length Counter method is applied: When LSTFMLENH[7:5] are programmed as 0, this bit will be set at the end of each frame. When LSTFMLENH[7:5] are programmed as non-zeros, this bit will be set at the end of the last frame. This bit is cleared to '0' when FMIIR_PIO is read	0
4	EOF_DECTED	RC	Received end-of-frame interrupt pending	0

			Bit 4 is set to a '1' when the receiver detects STO flag of a frame. This bit is cleared to '0' when FMIIR_PIO is read.	
3	TXFIFO_URUN	RC	Transmit underrun interrupt pending Bit 3 is set to '1' when an underrun occurs in the transmit FIFO. This bit is cleared when FMIIR_PIO is read.	0
2	RXFIFO_ORUN	RC	Receiver FIFO overrun interrupt Bit 2 is set to '1' when an overrun occurs in RX FIFO. Overrun occurs when CPU cannot read the data fast enough from RX FIFO and there is no empty space available in the RX FIFO register to store the frame data. If an overrun occurs, the CPU has to service the overrun before the FTUART010 can receive more data. This bit is cleared to 0 when FMIIR_PIO is read.	0
1	TXFIFO_TRIG	RO	Transmitter interrupt pending Bit 1 is set to '1' when TX FIFO level is below the trigger level.	0
0	RXFIFO_TRIG	RO	Receiver interrupt pending Bit 0 is set to '1' when RX FIFO level is equal to or above the trigger level.	0

8.6.7.20. Interrupt Identification Register in FIR Mode

All status bits reflect the raw status, no matter whether the corresponding enable bit is set or not.

Table 8-138 Interrupt Identification Register in FIR Mode (FMIIR_DMA, Offset: 0x3C, DMA Mode)

Bit	Name	Type	Description	Default Value
[7:6]	-	Rsvd	-	0
5	FRM_SENT	RC	This bit will be set under the following circumstance: If the Frame-Length Counter method is applied: When LSTFMLENH[7:5] are programmed as zeros, this bit will be set at the end of each frame. When LSTFMLENH[7:5] are programmed as non-zeros, this bit will be set at the end of the last frame. This bit is cleared to '0' when FMIIR_PIO is read.	0
4	RXFIFO_ORUN	RC	Receiver FIFO overrun interrupt Bit 4 is set to '1' when an overrun occurs in RX FIFO. Overrun occurs when CPU cannot read the data fast enough from RX FIFO and there is no empty space available in the RX FIFO register to store the frame data. If an overrun occurs, CPU has	0

			to service the overrun before FTUART010 can receive more data. This bit is cleared to '0' when FMIIR_PIO is read.	
3	TXFIFO_URUN	RC	Transmit underrun pending Bit 3is set to '1' when an underrun occurs in the data TX FIFO. This bitis cleared to '0' whenFMIIR_DMAis read.	0
2	STFIFO_ORUN	RC	Status FIFO overrun interrupt Bit 2 is set to '1' when an overrun occurs in the StatusFIFO. Overrun occurswhen CPU cannot read the data fast enough from the StatusFIFO and there is no empty space available inthe StatusFIFO register to store the information of the received frames. If an overrun occurs, the CPUhas to service the overrun before FTUART010can receive more data.Thisbitis cleared to '0' whenFMIIR_DMAis read.	0
1	STFIFO_TIME OUT	RO	Status FIFO timeout interrupt pending This bit is cleared to '0' by reading the register, STFF_RXLENH.	0
0	STFIFO_TRIG	RO	Status FIFO threshold interrupt pending This bit is cleared to'0' by reading the Status FIFOuntil thenumber of valid entriesis less than the set trigger level.	0

8.6.7.21. Interrupt Identification Enable Register in FIR Mode

Table 8-139 Interrupt Enable Register in FIR Mode (FMIIER_PIO, Offset: 0x40, PIO Mode)

Bit	Name	Type	Description	Default Value
[7:6]	-	Rsvd	-	0
5	IRIIREN5	RW	Enable the FRM_SENT interrupt	0
4	IRIIREN4	RW	Enable the EOF_DECTED interrupt	0
3	IRIIREN3	RW	Enable the TXFIFO_URUN interrupt	0
2	IRIIREN2	RW	Enable the RXFIFO_ORUN interrupt	0
1	IRIIREN1	RW	Enable the TXFIFO_TRIG interrupt	0
0	IRIIREN0	RW	Enable the RXFIFO_TRIG interrupt	0

8.6.7.22. Interrupt Identification Enable Register in FIR Mode

Table 8-140 Interrupt Enable Register in FIR Mode (FMIIER_DMA, Offset: 0x40, DMA Mode)

Bit	Name	Type	Description	Default Value
[7:6]	-	Rsvd	-	0

Vangotech

5	IRIIREN5	RW	Enable the FRM_SENT interrupt	0
4	IRIIREN4	RW	Enable the RXFIFO_ORUN interrupt-	0
3	IRIIREN3	RW	Enable the TXFIFO_URUN interrupt	0
2	IRIIREN2	RW	Enable the STFIFO_ORUN interrupt	0
1	IRIIREN1	RW	Enable the STFIFO_TRIG interrupt	0
0	IRIIREN0	RW	Enable S the TFIFO_TIME OUT interrupt	0

8.6.7.23. Status FIFO Line Status Register

This register returns the status byte at the bottom of Status FIFO. When bit 4 is '1', bits 0 to '3' indicate that any error condition occurred during reception of the corresponding frame. The error conditions will also affect the error flags in the FMLSR register.

Table 8-141 Status FIFO Line Status Register (STFF_STS, Offset: 0x44)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd	-	0
4	STS_VLD	RO	When set to '1', the bottom Status FIFO entry contains the valid data.	0
3	SIZE_ERR	RO	Set to '1' when a frame exceeding the maximum length has been received.	0
2	PHY_ERR	RO	Set to '1' when an illegal symbol is receivedor abort is detectedduring a reception.	0
1	CRC_ERR	RO	Set to '1' when a mismatch between the received CRC and the receiver-generated CRC is detected.	0
0	RXFIFO_ORUN	RO	This bit is set to '1' when the incoming characters or the entire frames have been discarded whenRX_FIFO is full.	0

8.6.7.24. Status FIFO Received Frame-Length Register Low

This register must be read only when the STS_VLD bit in STFIFO_STS is '1'. Upon reset, all bits are set to '0'.

Table 8-142 Status FIFO Received Frame-Length Register Low (STFF_RXLENL, Offset: 0x48)

Bit	Name	Type	Description	Default Value
[7:0]	RCVLENL	RO	Least significant eight bits of the received frame length	0

8.6.7.25. Status FIFO Received Frame-Length Register High

This register must be read only when the STS_VLD bit in the STFIFO_STS register is '1'. Upon reset, all bits are set to '0'. Reading this register removes the bottom STFIFO entry. Therefore, when the STS_VLD is read as '1', the STFF_RXLENL and STFF_RXLENH must be read sequentially.

Table 8-143 Status FIFO Received Frame-Length Register High (STFF_RXLENH, Offset: 0x4C)

Bit	Name	Type	Description	Default Value
[4:0]	RCVLENH	RO	Five most significant bits of the received frame length	0

8.6.7.26. Link Status Register in FIR Mode

All status bits reflect the raw status, no matter the corresponding enable bit is set or not.

Table 8-144 Link Status Register in FIR Mode (FMLSR, Offset: 0x50)

Bit	Name	Type	Description	Default Value
7	FIR_IDLE	RO	No FIR transaction is in progress.	1
6	TXFIFO_EMPTY	RO	TX FIFO is empty. When TX FIFO becomes empty, bit 6 is a '1', and when TX FIFO is not empty, it is a '0'.	1
5	STFIFO_FULL	RO	This bit is set to '1' when the Status FIFO is full.	0
4	SIZE_ERR	RC	Frame is longer than expected. This bit is set to '1' when a frame exceeding the maximum length (Set by the MRXLENL register and the MRXLENH register) is received. When this error is detected, the current frame reception is terminated. The data received in Rx FIFO is incomplete. The reception is stopped until the next BOF is detected. Bit 4 is cleared to '0' when the FMLSR register is read.	0
3	PHY_ERR	RC	Physical error In the FIR mode, bit 3 is set to '1' when an illegal symbol is received or abort is detected during the reception. This bit is cleared to '0' when the FMLSR register is read.	0
2	CRC_ERR	RC	CRC error When a bad CRC is detected on a data reception, this bit is set to 1. Bit 2 is cleared to '0' when the FMLSR register is read.	0
1	STFIFO_EMPTY	RO	This bit is set to a '1' when the Status FIFO is empty.	1
0	RXFIFO_EMPTY	RO	Rx FIFO is empty. When RX FIFO is empty, this bit is '1', and when RX FIFO is not empty, it is '0'.	1

8.6.7.27. Link Status Interrupt Enable Register in FIR Mode

Table 8-145 shows the bit assignment of the link status interrupt enable register in the FIR mode. The interrupt enable registers determine which status flag generate an interrupt request by setting the corresponding bit to '1'.

Table 8-145 Link Status Interrupt Enable Register in FIR Mode (FMLSIER, Offset: 0x54)

Bit	Name	Type	Description	Default Value
7	FMLSIER7	RW	Enable the FIR_IDLE interrupt	0
6	FMLSIER6	RW	Enable the TX_EMPTY interrupt	0
5	FMLSIER5	RW	Enable the STFIFO_FULL interrupt	0

Vangotech

4	FMLSIER4	RW	Enable the SIZE_ERR interrupt	0
3	FMLSIER3	RW	Enable the PHY_ERR interrupt	0
2	FMLSIER2	RW	Enable the CRC_ERR interrupt	0
1	FMLSIER1	RW	Enable the STFIFO_EMPTY interrupt	0
0	FMLSIER0	RW	Enable the RXFIFO_EMPTY interrupt	0

8.6.7.28. RX FIFO Count Register

The RX FIFO Count register represents the number of data bytes in RX FIFO. This register is useful when the number of remaining bytes of a frame received in RX FIFO is below the set trigger level. When FIFO is full, RXFF_CNTR is 0x10 for FIFO with the size of 16-byte. The 4-byte CRC32 appended to the frame is also received in RX FIFO. The bit-width of this register depends on the FIFO depth configuration. The bit-width shown in Table 8-146 is for the 16-byte FIFO. This register is for the FIFO mode only. When the FIFO mode is disabled, the returned value is unexpected.

Table 8-146. RX FIFO Count Register (RXFF_CNTR, Offset: 0x5C)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd	-	0
[4:0]	RXFF_CNTR	RO	The number of data bytes in RX FIFO.	0

8.6.7.29. Last Frame-Length Register Low

The LSTFMLENL register along with the LSTFMLENH register store the value for the number of bytes of the last data frame to be transmitted. The LSTFMLENL register stores the lower 8 bits and the LSTFMLENH register stores the upper bits.

The value of the frame length does not include any appended CRC bytes. Values from 1 to 213- 5 can be used. To properly program the LSTFMLEN register, CPU must always write the lower value into LSTFMLENL first and then the upper value into LSTFMLENH.

The upper 3 bits of LSTFMLENH define the number of frames to be transmitted. Programming these 3 bits as all 0's indicate only one frame with length TXLEN is intended to be transmitted (When the method Frame-Length Counter is applied or the DMA mode is selected).

Table 8-147. Last Frame-Length Register Low (LSTFMLENL, Offset: 0x60)

Bit	Name	Type	Description	Default Value
[7:0]	LSTFMLENL	RW	Last Transmitter Frame Length Low	0

8.6.7.30. Last Frame Length Register High

Table 8-148 Last Frame-Length Register High (LSTFMLENH, Offset: 0x64)

Bit	Name	Type	Description	Default Value
[7:5]	FRM_NUM	RW	Number of frames to be transmitted	0
[4:0]	LSTFMLENH	RW	Last Transmitter Frame Length High	0

Vangotech

Table 8-149 Frame Number Decoding Table (LSTFMLENH, Offset: 0x64)

LSTFMLENH			Description
Bit 7	Bit 6	Bit 5	
0	0	0	The data in TX FIFO is always transmitted as an equal-size (Length is TXLEN) frames.
0	0	1	The data in TX FIFO is always transmitted as one equal-size frame (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
0	1	0	The data in TX FIFO is always transmitted as two equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
0	1	1	The data in TX FIFO is always transmitted as three equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
1	0	0	The data in TX FIFO is always transmitted as four equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
1	0	1	The data in TX FIFO is always transmitted as five equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
1	1	0	The data in TX FIFO is always transmitted as six equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).
1	1	1	The data in TX FIFO is always transmitted as seven equal-size frames (Length is TXLEN) plus one equal- or different-size frame (Length is LSTFMLEN).

8.6.7.31. Feature Register

The value of the feature register depends on the hardware configuration. Please refer to Table 8-150 for details.

Table 8-150 Feature Register (Feature, Offset: 0x68)

Bit	Name	Type	Description	Default Value
[7:5]	-	Rsvd	-	0
4	IrDA_INSIDE	RO	1: FTUART010 contains the IrDA function. 0: FTUART010 is a pure UART.	1
[3:0]	FIFO_DEPTH	RO	4'b0001: TX/RX FIFOs are 16-byte deep. 4'b0010: TX/RX FIFOs are 32-byte deep. 4'b0100: TX/RX FIFOs are 64-byte deep. 4'b1000: TX/RX FIFOs are 128-byte deep.	1

Vangotech

8.6.7.32. Revision Register Digit 1

Revision Register Digit 1 combined with Digit 2 and Digit 3 are used to indicate the version of RTL. For example, if Revision Register Digit 1 reads as 1, Digit 2 as 4, and Digit 3 as 0, the version of this RTL is version_1_4_0.

Table 8-151. Revision Register Digit 1 (REVD1, Offset: 0x6C)

Bit	Name	Type	Description	Default Value
[7:0]	REVD1	RO	Denote First Digit of Revision Number	1

8.6.7.33. Revision Register Digit 2

Table 8-152. Revision Register Digit 2 (REVD2, Offset: 0x70)

Bit	Name	Type	Description	Default Value
[7:0]	REVD2	RO	Denote Second Digit of Revision Number	0x11

8.6.7.34. Revision Register Digit 3

Table 8-153. Revision Register Digit 3 (REVD3, Offset: 0x74)

Bit	Name	Type	Description	Default Value
[7:0]	REVD3	RO	Denote Third Digit of Revision Number	0

Vangotech

8.7. MAC

8.7.1. Overview

FTGMAC030 is a high-quality Ethernet controller with the DMA function. It includes the AHB/AXI wrapper, DMA engine, on-chip memories (TXFIFO and RXFIFO), MAC, and MII/RMII/RGMII interfaces.

FTGMAC030 is an Ethernet controller that provides the AHB/AXI master capability and is fully compliant with the IEEE 802.3 specification for the 10/100 Mbps Ethernet and the IEEE 802.3z specification for the 1000 Mbps Ethernet. FTGMAC030 supports MII//RMII/RGMII interfaces. The FTGMAC030 Ethernet controller with the DMA function handles all data transfers between the system memory and the on-chip memories. With the DMA engine, this controller reduces the CPU loading, maximizes the performance, and minimizes the FIFO size. FTGMAC030 has the on-chip memories for buffering so that the external local-buffer memory will not be needed. The MII/RMII interfaces support two specific data rates, 10 Mbps and 100 Mbps. The RGMII interface supports three specific data rates, 10 Mbps, 100 Mbps, and 1000 Mbps. The RGMII protocol is based on the HP RGMII Specification, version 2.0.

In order to reduce the processing load of the host CPU, FTGMAC030 implements TCP, UDP, IPv4, and IPv6 checksum generations and validations, and supports VLAN tagging. For the QoS and CoS requirements, FTGMAC030 supports the high-priority queues to reduce the processing load of the host CPU for transmitting packets.

FTGMAC030 provides a Wake-On-LAN function. It supports three wake-up events: Link status change, magic packet, and wake-up frame. This function allows a system containing FTGMAC030 to be woken up from the remote side. The AHB/AXI wrapper allows users to quickly integrate FTGMAC030 into the SoC design.

Terminology

AMBA: Advanced Micro-controller Bus Architecture

AHB: Advanced High-performance Bus

AXI: Advanced eXtensible Interface

MAC: Media Access Control

MII: Media Independent Interface

RMII: Reduced Media Independent Interface

RGMII: Reduced Gigabit Media Independent Interface

EEE: Energy-Efficient Ethernet

LPI: Low Power Idle

PTP: Precision Time Protocol

TSU: Time Stamp Unit

8.7.2. Features:

- Supports AHB or AXI bus master for DMA function
- Supports AHB or APB bus slave for register setting
- DMA engine for transmitting and receiving packets
- Supports TCP, UDP, IPv4, and IPv6 checksum offloads

Vangotech

- Supports IEEE 802.1Q VLAN tag insertion and removal
- Supports high-priority queues for QoS and CoS applications
- Supports Wake-On-LAN function and three wake-up events: Link status change, magic packet, and wake-up frame
- Independent TX/RX FIFOs
- Supports half and full duplex operations
- Supports flow control for full duplex and back pressure for half duplex
- Supports MII/RMII/RGMII interfaces
- Supports jumbo packets (9K bytes)
- Supports IEEE 1588v1 and 1588v2 frame recognition and timestamp generation
- Supports IEEE 802.3az LPI assertion and detection

8.7.3. Block Diagram

The blocks shown in Figure 8-33 are listed and described as follows:

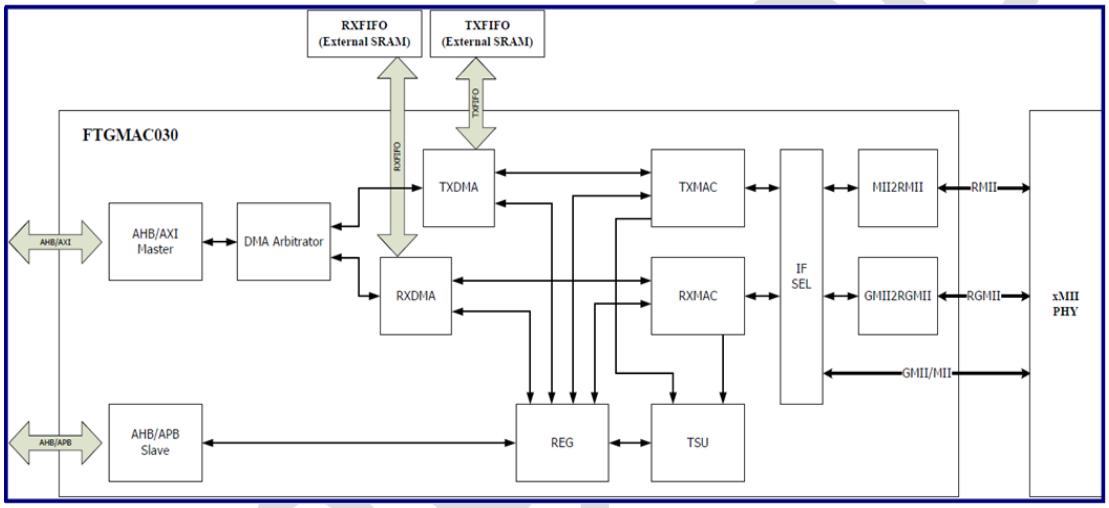


Figure 8-33. Block Diagram

8.7.3.1. AHB/AXI Master

The block implements the AHB/AXI master function of FTGMAC030. When TXDMA moves the transmit packets to TXFIFO, TXDMA will request DMA Arbitrator to use the DMA channel. After DMA Arbitrator gives the DMA channel to TXDMA, it initiates a burst transfer to AHB/AXI Master through DMA Arbitrator and AHB/AXI Master in turn to initiate a read operation to the AHB/AXI bus. Then, AHB/AXI Master passes the transmit packet data to TXDMA. When RXDMA moves the receive packets from RXFIFO, RXDMA will request DMA Arbitrator to use the DMA channel. After DMA Arbitrator gives the DMA channel to RXDMA, it initiates a burst transfer to AHB/AXI Master through DMA Arbitrator and the AHB/AXI Master in turn to initiate a write operation to the AHB/AXI bus. Then, AHB/AXI Master passes the receive packet data from RXDMA.

8.7.3.2. AHB/APB Slave

The block implements the AHB/APB slave function of FTGMAC030. When the bus master writes data to REG in FTGMAC030, it initiates a write operation to the AHB/APB bus, which will prompt AHB/APB Slave to respond to the operation by forwarding the write data to REG. If the bus master reads data from REG in

FTGMAC030, it initiates a read operation to the AHB/APB bus, which will prompt AHB/APB Slave to respond to the operation by obtaining the read data from REG and passes information back to the AHB/APB bus to complete a read operation.

8.7.3.3. DMA Arbitrator

The block acts as the bridge between two sets of control signals from TXDMA and RXDMA, and the control signal to AHB/AXI Master. It acts as an arbiter to decide if TXDMA or RXDMA is entitled to use AHB/AXI Master.

8.7.3.4. TXDMA

The block performs four main functions: (1) Read the transmit descriptor; (2) Move the transmit packet data from the transmit buffer to TXFIFO; (3) Control the read/write actions of TXFIFO; and (4) Do the TCP/UDP/IP checksum calculation.

When TXDMA transmits packet to Ethernet, it will fetch the descriptor information and transmit the buffer base address and size. Then, TXDMA moves the transmit packet data from the corresponding transmit buffer to TXFIFO and requests TXMAC to read the transmit packet data with the help of TXDMA and sends it to the network. When the packet has been transmitted, TXMAC sends the transmit status to TXDMA and writes the transmit status back to the transmit descriptor.

8.7.3.5. RXDMA

RXDMA performs three main functions:

1. Read the receive descriptor and write the receive status to the receive descriptor
2. Move the receive packet data from RXFIFO to the receive buffer
3. Control the read/write actions of RXFIFO

When FTGMAC030 sends a packet, it will save the received packet into the RXFIFO. Then RXDMA moves the received packet from RXFIFO to the receiving buffer and writes the receive status to the receive descriptor.

8.7.3.6. TXMAC

TXMAC transmits packets from TXFIFO to Ethernet with CRC, preamble, jam generator, and transmit state machine included.

When TXMAC transmits a packet, it detects the Ethernet status and halts the transmission until Ethernet is free. Then, TXMAC adds preamble and CRC to this packet and sends the packet to Ethernet. If TXMAC detects the collision during a transmission, it sends the jam to Ethernet and determines whether the collision is excessive. If not, it waits for backing to off-time and transmits the packet again. TXMAC also performs the VLAN tag insertion.

Besides, if TXMAC can transmit no packets, then it will generate LPI pattern when Send_TX_LPI function is turn on. For IEEE 1588 PTP application, TXMAC can recognize the PTP frame to record precise timestamp for users.

8.7.3.7. RXMAC

RXMAC receives packets from Ethernet to RXFIFO, with address recognition circuit, CRC check circuit, and receive state machine included.

When a packet is incoming, RXDMA will pass the received packet data to RXFIFO from RXMAC. RXDMA will save the received packet in RXFIFO if both the CRC result and packet address are correct; otherwise, the packet will be discarded. RXMAC also performs the TCP/UDP/IP checksum verification and VLAN tag

removal.

Besides, RXMAC can recognize LPI pattern from PHY RX channel. For IEEE 1588 PTP application, RXMAC can recognize the PTP frame to record precise timestamp for users.

8.7.3.8. REG

The block stores the registers of FTGMAC030. The bus master reads/writes these registers by using AHB/APB Slave.

8.7.3.9. TSU

TSU control time stamp timer of FTGMAC030. User can program and adjust the time stamp timer. Besides, it records the time stamp timer value when TX PTP frame or RX PTP frame is detected.

8.7.3.10. IFSEL

The block acts as selector that switches the Ethernet interface by user programming. When users select to RMII, it will switch the data path to MII2RMII block. When user select to RGMII, then it switch the data path to GMII2RGMII block. Otherwise, MII interface pass through to IP boundary.

8.7.3.11. MII2RMII

The block is a wrapper that translates the TXMAC/RXMAC MII signals to the RMII signals. With this module, FTGMAC030 will support the RMII interface. The interface type to Ethernet side is decided by GMAC Interface Selection Register (Offset 0xE8). When bit[1:0] is set to 0b01, the RMII interface will be the interface to Ethernet. FTGMAC030 can only work in the 10/100 Mbps modes. When bit[1:0] is set to 0b00, the MII interface is the interface to Ethernet. FTGMAC030 works in the 10/100/1000 Mbps modes.

8.7.3.12. GMII2RGMII

The block is a wrapper that translates the TXMAC/RXMAC GMII signals to the RGMII signals. The RGMII protocol is compatible with the HP RGMII Specification, Version 2.0. With this module, FTGMAC030 will support the RGMII interface. The interface type to the Ethernet side is decided by GMAC Interface Selection Register (Offset 0xE8). When bit[1:0] is set to 0b10, the RGMII interface will be the interface to Ethernet. FTGMAC030 can work in the 10/100/1000 Mbps modes. When bit[1:0] is set to 0b00, the MII interface will be the interface to Ethernet. FTGMAC030 works in the 10/100/1000 Mbps modes

8.7.3.13. TXFIFO and RXFIFO

The two modules are not blocks of FTGMAC030. They are external single-port SRAM and act as buffer to store packet. Users can decide the memory size depending on the maximum packet length and application.

8.7.3.14. xMII PHY

The module is not the block of FTGMAC030. It is external Ethernet PHY. Users need to select a proper interface to connect FTGMAC030 and PHY.

8.7.4. Operations and Functional Descriptions

8.7.4.1. Transmit Descriptors and Data Buffers

FTGMAC030 uses a descriptor ring to manage the transmit buffers. The transmit descriptors and data buffers are all located in the system memory. FTGMAC030 moves the transmit packet data from the transmit buffers in the system memory to TX FIFO inside FTGMAC030 and then transmits the packet to Ethernet. The transmit descriptors that reside in the system memory act as pointers to the transmit buffers.

Vangotech

Each transmit descriptor contains a transmit buffer. A transmit buffer consists of either an entire frame or part of a frame, but not multiple frames. The transmit descriptor contains the transmit buffer status and the transmit buffer that can only contain the transmit data. FTGMAC030 supports two descriptor rings for transmission. These descriptor rings are the normal-priority transmit ring and high-priority transmit ring. The normal-priority transmit ring is for the normal packet transmission; the high-priority transmit ring is for the high priority packet transmission. Higher priority packets can be put into the high-priority transmit ring for quicker transmission.

The transmit descriptor structure is as follows.

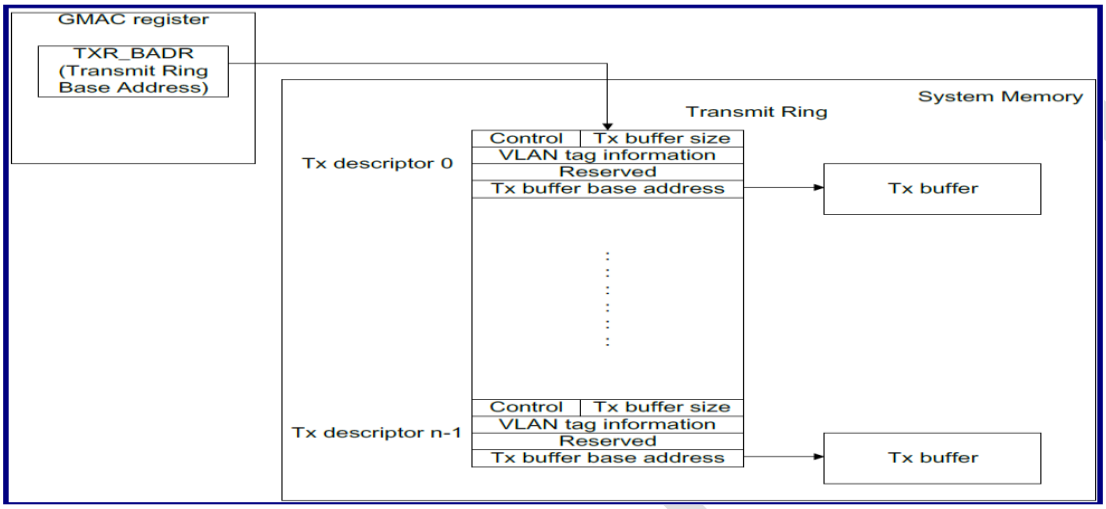


Figure 8-34. Transmit Ring Descriptor Structure

Notes:

- The transmit descriptor ring must have at least two or more descriptors.
- The start address of each transmit descriptor must be 16-byte aligned.
- The maximum transmit packet size including CRC is 9216 bytes (9220 bytes if the VLAN tag is inserted).
- FTGMAC030 supports the IPV4/6 checksum offload. Software must be certain that the transmit packet is an IPV4/6 packet when software requests FTGMAC030 to do checksum offload. Besides, users must be make sure that the IPv4/6 packet is smaller than TX FIFO size when enabling the checksum offload.
- LLC packet is IEEE 802.3/802.2/SNAP format packet.
- FTGMAC030 does not support the following two packets for the checksum offload:

  IEEE 802.3 with IEEE 802.2 packet

  IEEE 802.3 with 802.1Q and 802.2 packets.
- When preparing a transmit packet which contains more than one transmit descriptor, the first transmit descriptor must be the last set descriptor of the transmit packet.
- When setting the transmit descriptor, TXDES0 must be set last. Thus, the setting procedure should be as follows:

  Set TXDES3

  Set TXDES2

  Set TXDES1

  Set TXDES0

TXDES0	TXDMA_OWN	Control1	TX buffer size

TXDES1	VLAN control	VLAN tag control information	
TXDES2	Reserved		
TXDES3	TX buffer base address		

TXDES0 contains the control bits and transmit buffer size and descriptor ownership information.

Bit	Name	Description
31	TXDMA_OWN	TXDMA_OWN - TXDMA ownership bit When set, it indicates that the descriptor is owned by FTGMAC030. When reset, it indicates that the software owns the descriptor. FTGMAC030 clears this bit when it completes the frame transmission.
30	-	Reserved
29	FTS	FTS - First Transmit Segment descriptor When set, it indicates that this is the first descriptor of a TX packet.
28	LTS	LTS - Last Transmit Segment descriptor. When set, it indicates that this is the last descriptor of a TX packet.
27	BUS_ERR	BUS_ERR - BUS response ERROR When set, it indicates that the system bus respond ERROR for TX descriptor and buffer reading. For system bus is AMBA AHB, the ERROR is "ERROR". For system bus is AMBA AXI, the ERROR are "EXOK", "SLVERR", and "DECERR". It is valid only when FTS = 1.
[26:20]	-	Reserved
19	CRC_ERR	CRC_ERR - CRC error When CRC_ERR = 1 and DISCARD_CRCERR, bit 18 of MAC Control Register (Offset: 0x50) = 1, TXDMA will discard the transmit packet and will not send it to Ethernet.
[18:16]	-	Reserved
15	EDOTR	EDOTR - End Descriptor of Transmit Ring When set, it indicates that the descriptor is the last descriptor of the transmit ring.
14	-	Reserved
[13:0]	TXBUF_SIZE	Transmit buffer size in byte The transmit buffer size cannot be zero.

TXDES1 contains the VLAN control bits and VLAN tag control information.

Bit	Name	Description
31	TXIC	TXIC - Transmit Interrupt on Completion When set, FTGMAC030 will assert the transmit interrupt after the present frame has been transmitted. It is valid only when FTS = 1 and bits 8 ~ 14 (TXINT_THR, TXINT_CYC) of Interrupt Timer Control Register = 0.
30	TX2FIC	TX2FIC - Transmit to FIFO Interrupt on Completion

Vangotech

		When set, FTGMAC030 will assert the transmit interrupt after the present frame has been moved into TX FIFO. It is valid only when FTS = 1.
[29:23]	-	Reserved
22	LLC_PKT	The LLC packet is IEEE 802.3/802.2/SNAP format packet. When set, FTGMAC030 will treat the packet as the IEEE 802.3/802.2/SNAP LLC packet. It is valid only when FTS = 1.
[21:20]	PKT_TYPE	Packet Type When set, FTGMAC030 will treat the packet as IPv4 or IPv6 or other packet type. 0b00: IPv4 packet type 0b01: IPv6 packet type 0b10: Other packet type 0b11: Reserved It is valid only when FTS = 1.
19	IPCS_EN	IPCS_EN - IP checksum offload enable When set, FTGMAC030 will offload the IP checksum. It is valid only when FTS = 1.
18	UDPCS_EN	UDPCS_EN - UDP checksum offload enable When set, FTGMAC030 will offload the UDP checksum. It is valid only when FTS = 1.
17	TCPCS_EN	TCPCS_EN - TCP checksum offload enable. When set, FTGMAC030 will offload the TCP checksum. It is valid only when FTS = 1.
16	INS_VLAN	Insert VLAN tag When set, 0x8100 (IEEE 802.1Q VLAN Tag Type) will be inserted after the source address, and two bytes of VLAN_TAGC will be inserted after the IEEE 802.1Q VLAN tag type. When cleared, the packet content will not be changed when transmitting to network. It is valid only when FTS = 1.
[15:0]	VLAN_TAGC	VLAN tag control information The 2-byte VLAN tag control information contains information from the upper layer of user priority, canonical format indicator, and VLAN ID. Please refer to IEEE 802.1Q for more VLAN tag information. Bits[15:13]: User priority Bit 12: CFI (Canonical Format Indicator) Bits[11:0]: VID (VLAN Identifier) It is valid only when FTS = 1.

TXDES2 is reserved.

Bit	Name	Description
[31:0]	-	Reserved

TXDES3 contains the transmit buffer base address.

Vangotech

Bit	Name	Description
[31:0]	TXBUF_BADR	Transmit buffer base address

8.7.4.2. Receive Descriptors and Data Buffers

FTGMAC030 uses a descriptor ring to manage the receive buffers. The receive descriptors and data buffers are all located in the system memory. FTGMAC030 first stores the packet received from the network in the RX FIFO and then moves the received packet data to the receive buffers in system memory. The receive descriptors that reside in the system memory act as pointers to the receive buffers.

There is a descriptor ring for reception. The base address of the receive ring is in the Receive Ring Base Address Register (RXR_BADR, offset: 24h ~ 27h). Each receive descriptor contains a receive buffer. A receive buffer consists of either an entire frame or part of a frame, but not the multiple frames. The receive descriptor contains the receive buffer status and the receive buffer only contains the receive packet data.

The receive descriptor structure is as follows:

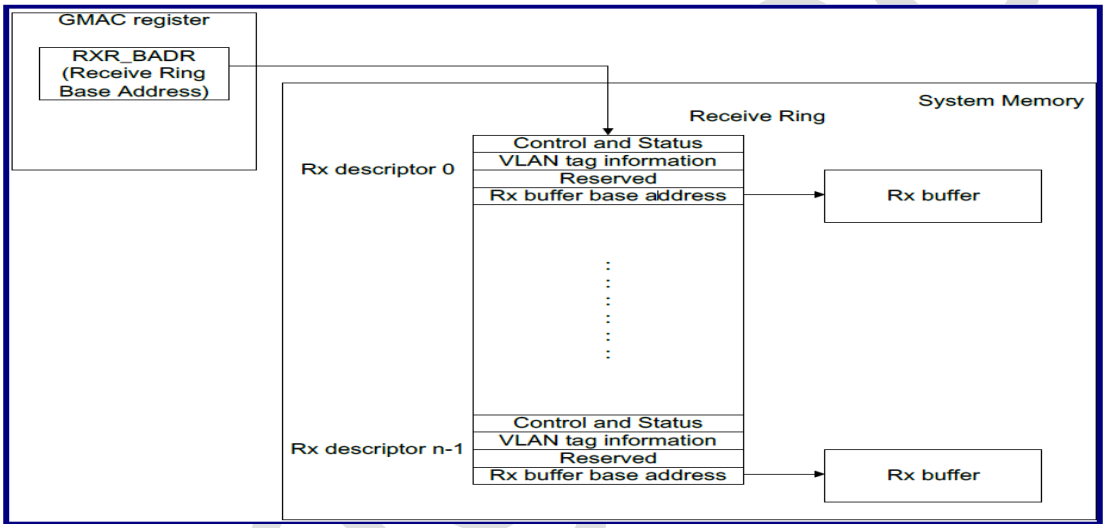


Figure 8-35. Receive Ring Descriptor Structure

Notes:

- The start address of each receive descriptor must be 16-byte aligned.
- The maximum receive packet size is 9216 bytes (9220 bytes for packets with the VLAN tag).
- FTGMAC030 supports the IPV4/6 checksum offload. If the incoming packet is not an IPV4/6 packet, FTGMAC030 will not perform the checksum verification. The IPCS_FAIL, UDPCS_FAIL, and TCPCS_FAIL fields are always zeros.
- LLC packet is IEEE 802.3/802.2/SNAP format packet.
- FTGMAC030 does not support the following two packets for the checksum offload. They are IEEE 802.3 with IEEE 802.2 packet and IEEE 802.3 with 802.1Q and 802.2 packets.

8.7.4.3. Transmitting Packets

When software wants to transmit a packet to Ethernet, it moves the packet data into the transmit buffer first. Then, software writes the packet length and position into the transmit descriptor and triggers FTGMAC030 to send the packet. After the entire packet has been moved into TX FIFO, FTGMAC030 begins to transmit it to Ethernet. When the packet has been transmitted, FTGMAC030 asserts interrupt to notify software that the packet has been transmitted successfully. Higher priority packets can be put into the high priority descriptor for quicker transmission.

8.7.4.4. Receiving Packets

When there is an incoming packet, FTGMAC030 first saves the received packet in RX FIFO if the address check result is correct. After the incoming packet is successfully saved in RX FIFO, FTGMAC030 initiates Direct Memory Access (DMA) function to move the received packet data from RX FIFO to the system memory. Then, FTGMAC030 asserts interrupt to notify software that the packet has been received successfully.

8.7.4.5. Ethernet Address Filtering

FTGMAC030 can be set to recognize any one of the Ethernet receive address groups described in the following table.

- RX_BROADPKT: Bit 11 of MAC Control Register (Offset: 0x50)
- RX_MULTIPKT: Bit 10 of MAC Control Register (Offset: 0x50)
- RX_HT: Bit 9 of MAC Control Register (Offset: 0x50)
- RX_ALLADR: Bit 8 of MAC Control Register (Offset: 0x50)

RX_ ALLADR	RX_ MULTIPKT	RX_ BROADPKT	RX_HT	Group	Received Frame Rule (DA means that destination address of the received frame.)
0	0	0	0	A	DA exactly matches MAC_ADR (Offset: 0x8).
0	0	0	1	B	DA exactly matches MAC_ADR (Offset: 0x8). DA is a multicast address. DA passes the address filtering of the multicast address hash table.
0	0	1	0	C	DA exactly matches MAC_ADR (Offset: 0x8). DA is a broadcast address.
0	0	1	1	D	DA exactly matches MAC_ADR (Offset: 0x8). DA is a multicast address. DA passes the address filtering of the multicast address hash table. DA is a broadcast address.
0	1	X	X	E	DA exactly matches MAC_ADR (Offset: 0x8). DA is a multicast address.
1	X	X	X	F	Reception of all frames on the network regardless of DA

8.7.4.6. DMA Arbitration Scheme

The DMA arbitration scheme is decided by RX_THR_EN (Bit 6 of DMA Burst Length and Arbitration Control

Register, offset: 0x38). When RX_THR_EN = 0, the DMA arbitration scheme does a fair arbitration between TXDMA and RXDMA. If both TXDMA and RXDMA request the DMA channel at the same time, the one last using the DMA channel has lower priority to get the DMA channel.

When RX_THR_EN is set, if the used space in RX FIFO is larger than or equal to RXFIFO_HTHR (Bits 5 ~ 3 of DMA Burst Length and Arbitration Control Register, offset: 0x38), RXDMA has higher priority over TXDMA by using the DMA channel. RXDMA keeps the higher priority until the used space in RX FIFO is less than or equal to RXFIFO_LTHR (Bits 2 ~ 0 of DMA Burst Length and Arbitration Control Register, offset: 0x38). Then, TXDMA gets higher priority over RXDMA. So software must set RXFIFO_HTHR to be larger than RXFIFO_LTHR to keep FTGMAC030 working correctly.

8.7.4.7. Wake-On-LAN

FTGMAC030 supports the Wake-On-LAN function. The Wake-On-LAN function supports three wake-up events: Link status change, magic packet, and wake-up frame.

Link Status Change

Link status change refers to the event where the link state to Ethernet changes. PHY offers a phy_linksts signal. If the link state to Ethernet changes, the state of phy_linksts will also change.

If FTGMAC030 enables the link status change mode, the link status change will be treated as a wake-up event.

Magic Packet

A magic packet contains a specific sequence consisting of 16 duplications of network the adaptor node address without breaks. The specific sequence must be preceded by 6 bytes of 0xFF. The format of a magic packet goes as follows:

DA + SA + … + 6 * (0xFFh) + …+ 16 * (network adaptor's node address) + …

If FTGMAC030 enables the magic packet mode, a magic packet will be treated as a wake-up event.

Wake-up Frame

The purpose of the wake-up frame is to wake up a system when another machine on the network needs to communicate with the system. It does not require the application running on the remote machine to send a special wake-up frame pattern. Instead, when FTGMAC030 is in the wake-up frame mode, it tries to identify certain interesting frames that are sent by the existing network protocols. Some examples are NETBIOS name lookups and ARP requests.

Before putting FTGMAC030 into the wake-up frame mode, the system should pass to the driver a list of wake-up frames that will wake up the system and the corresponding byte masks that correspond to bits will set to 1. Each byte mask defines which bytes of the incoming frames should be compared with the corresponding wake-up frame in order to determine whether or not to accept the incoming frame as a wake-up event.

Besides, the packet length of wake-up frame must be at least 128 bytes.

Table 8-154. Wake-up Frame Format

Wake-up Frame Format								
Byte content	0x00	0x25	0x67	0x90	0x44	0xA3	0x6C	…
Byte mask	0	0	0	1	1	0	1	…

There are two ways to identify if a received packet is a wake-up frame. Signature matching is used in FTGMAC030.

- Exact matching: FTGMAC030 needs many registers to store the entire byte content and byte mask for

each wake-up frame. When a packet arrives from the network, FTGMAC030 checks those bytes of the incoming frame that correspond to bits that set to 1 in the byte mask for each wake-up frame. If the check result is ok for any wake-up frame and if the incoming frame passes the standard CRC check, FTGMAC030 would treat it as a wake-up event.

- Signature matching: FTGMAC030 needs a CRC generation circuit and registers to save the entire byte mask and 4-byte CRC register for each wake-up frame. The driver calculates a CRC value based on those bytes of any wake-up frame which correspond to bits that set to 1 in the byte mask. Besides, the start offset of byte mask field is Destination Address (DA) field. The driver stores the resulting value and corresponding byte mask into FTGMAC030. When the wake-up frame mode is enabled as a frame arriving from the network, each CRC generator calculates a CRC value based on those bytes of the incoming frame which correspond to bits that set to 1 in the CRC generator byte mask. If the calculated value matches the stored value for any wake-up frame and the incoming frame passes the standard CRC check, FTGMAC030 would treat it as a wake-up event.

8.7.4.8. Flow Control

FTGMAC030 implements the flow control function. It supports IEEE802.3x flow control for the full-duplex mode and backpressure for the half-duplex mode.

The IEEE802.3x flow control is used in the full-duplex mode. When A and B are reciprocally transmitting and receiving packets in the full-duplex mode, if RX FIFO in B is nearly full, B sends a pause frame to A in order to avoid loss of the received packet. Then A is inhibited from transmitting packets for a specified period of time. B consumes the received data during the specified period of time. A continues to send packets to B after the pause time has lapsed. Here is a brief account of the features of the flow control in the full-duplex mode:

- Software configures the pause time of the pause frame.
- FTGMAC030 sends the pause frame according to the low/high threshold of RX FIFO.
- Software sends the pause frame by writing the register.

The back pressure mode is used in the half-duplex operation. When A is transmitting and receiving packets in the half-duplex mode, if RX FIFO in A is nearly full, A will send a jam pattern to generate collisions to avoid incoming packets from being saved into RX FIFO. A consumes the received data as soon as possible during the period of time. A does not send a jam pattern to receive packets again when RX FIFO is not nearly full. Here is a brief account of the features of the back pressure mode:

- Software configures the length of the jam.
- FTGMAC030 sends jam according to the low/high threshold of RX FIFO.

8.7.4.9. Supported Ethernet Frame Type for Checksum Offload

Two Ethernet frame formats are supported in order to correctly identify the IP and TCP/UDP headers. FTGMAC030 supports Ethernet Type II format and IEEE 802.3/802.2/SNAP format as shown below. VLAN tagging is also supported on top of these two frame formats.

Ethernet Type II

Destination Address	Source Address	Type	Data	CRC
6-byte	6-byte	2-byte	n-byte	4-byte

IEEE 802.3/802.2/SNAP

Destination Address	Source Address	Length	DSAP	SSAP	Control	OUI	Type	Data	CRC
6-byte	6-byte	2-byte	1-byte	1-byte	1-byte	3-byte	2-byte	n-byte	4-byte
where DSAP = 0xAA, SSAP = 0xAA, Control = 0x03, OUI = 0x000000									

8.7.4.10. IEEE 1588 PTP Frame Reorganization

If users configure the PTP function, then FTGMAC030 supports IEEE 1588 PTP frame recognition.

The IEEE 1588 versions 1 and 2 specify different PTP frame types on Ethernet. FTGMAC030 can recognize the following PTP frames:

For IEEE 1588 version 1:

- PTP over UDP over IPv4 with multicast address

For IEEE 1588 version 2:

- PTP over UDP over IPv4 with multicast address
- PTP over UDP over IPv4 with unicast address
- PTP over UDP over IPv6 with multicast address
- PTP over Ethernet with multicast address

For the detailed format information, please refer to the IEEE 1588-2002 and IEEE 1588-2008 specifications. Meanwhile, the PTP clock cycle value must specify to the PTP Period Increment 0/1 Register (offset 0x13C, 0x140) and PTP Period Offset Increment Register (offset 0x144).

Example 1, the frequency of PTPClk is 125 MHz, period is 8 ns. Then,

- PTP Period Increment 0 Register is 0x8,
- PTP Period Increment 1 Register is 0x0,
- PTP Period Offset Increment Register is 0.

Example 2, the frequency of PTPClk is 160 MHz, period is 6.25 ns. Then,

- PTP Period Increment 0 Register is 0x6,
- PTP Period Increment 1 Register is 0xEE6B280, due to 0.25ns = 250000000nns.
- PTP Period Offset Increment Register is 0x0.

Example 3, the frequency of PTPClk is 133 MHz, period is 7.5187…ns. Due to the period is indivisibility, so the clock has 133 cycles at every 1 microsecond. So the timer with a 133 MHz clock source is constructed by incrementing by 7ns for 132 cycles and then incrementing by 76 (132 x 7 + 76 = 1000). It means that the period is 7 ns and increment is 76 ns at every 133th clock cycle.

- PTP Period Increment 0 Register is 0x7,
- PTP Period Increment 1 Register is 0x0
- PTP Period Offset Increment Register is 0x0084_004C.

8.7.5. Initialization/Application Information

This chapter discusses the procedure for transmitting and receiving frames, as well as the procedure for entering and exiting the power-down mode.

Besides, the Ethernet interface has no "pause" protocol during one packet transmission or receiving. To avoid the packet from being interrupted; FTGMAC030 has system frequency limitation for 10M/100M/1000M Ethernet mode:

Vangotech

	Ethernet 10M bps	Ethernet 100 Mbps	Ethernet 1000 Mbps
System min. frequency requirement	10 MHz	10 MHz	50 MHz

8.7.5.1. Frame Transmitting Procedure

The frame transmitting procedure is as follows:

Initialization:

- Set MAC_SPEED (0x50.bit[25:24]) to proper setting
- Set SW_RST (0x50.bit[31]) = 1 to do software reset
- Read the Feature Register (Offset: 0xF8) to get the real TX/RX FIFO size in hardware
- Allocate the system memory for the transmit descriptor ring and transmit buffer
- Initialize the transmit descriptor ring
- Set the Normal Priority Transmit Ring Base Address Register (Offset: 0x20) to the base address of the normal priority transmit descriptor ring in the system memory
- Set the High Priority Transmit Ring Base Address Register (Offset: 0x2C) to the base address of the high-priority transmit descriptor ring in the system memory if necessary
- Set the Interrupt Enable Register (Offset: 0x4)
- Set the MAC Address Register (Offset: 0x8)
- Set the Multicast Address Hash Table Register (Offset: 0x10)
- Set the TX Interrupt Timer Control Register (Offset: 0x30) to select the manner of the transmit interrupt
- Set the Automatic Polling Timer Control Register (Offset: 0x38) to select the manner of transmit poll
- Set the Transmit Priority Arbitration and FIFO Control Register (Offset: 0x48) to set transmit priority arbitration and proper TX/RX FIFO size in use
- Set the DMA Burst Length and Arbitration Control Register (Offset: 0x3C) to set proper TX/RX descriptor size and DMA burst length
- Set the MAC Control Register (Offset: 0x50) to set valid configuration for FTGMAC030 and to enable the transmit channel

Transmit procedure:

- Software checks if the remained normal-priority transmit descriptors are sufficient for the next packet transmission. If not, software needs to wait until the transmit descriptors are sufficient.
- Prepare the transmit packet data to the transmit buffer
- Set the normal priority transmit descriptor
- Write the Normal Priority Transmit Poll Demand Register (Offset: 0x18) to trigger FTGMAC030 to poll the transmit descriptor if necessary when the packet is put in the normal-priority transmit ring.
- Wait for interrupt
- When interrupt occurs, software checks if it is a transmit interrupt. If ISR[4] = 1, it means that the packet has been successfully transmitted to network. If ISR[7] = 1, it means that the packet has been aborted during transmission due to late collision or excessive collision or under-run.
- Steps 1 through 6 are for the normal packets in the normal-priority transmit ring. If software wants to transmit the high-priority packets, repeat these steps for the high-priority transmit ring.

Notes:

Vangotech

- When setting the transmit descriptor, TXDES0 must be set last. Thus, the setting procedure should be as follows:

  Set TXDES3

  Set TXDES2

  Set TXDES1

  Set TXDES0
- When preparing a transmit packet which contains more than one transmit descriptor, the first transmit descriptor must be the last set descriptor of the transmit packet.

8.7.5.2. Frame Receiving Procedure

The frame receiving procedure is as follows:

Initialization:

- Set MAC_SPEED (0x50.bit[25:24]) to proper setting
- Set SW_RST (0x50.bit[31]) = 1 to perform software reset
- Read the Feature Register (Offset: 0xF8) to get the real TX/RX FIFO size in hardware
- Allocate the system memory for the receive descriptor ring and receive buffer
- Initialize the receive descriptor ring
- Set the Receive Ring Base Address Register (Offset: 0x24) to the base address of the receive descriptor ring in the system memory
- Set the Interrupt Enable Register (Offset: 0x4)
- Set the MAC Address Register (Offset: 0x8)
- Set the Multicast Address Hash Table Register (Offset: 0x10)
- Set the RX Interrupt Timer Control Register (Offset: 0x34) to select the manner of the receive interrupt
- Set the Automatic Polling Timer Control Register (Offset: 0x38) to select the manner of receive poll
- Set the Transmit Priority Arbitration and FIFO Control Register (Offset: 0x48) for the transmit priority arbitration and proper TX/RX FIFO size in use
- Set the DMA Burst Length and Arbitration Control Register (Offset: 0x3C) for the proper TX/RX descriptor size and DMA burst length
- Set the MAC Control Register (Offset: 0x50) to set valid configuration for FTGMAC030 and to enable receive channel
- Write the Receive Poll Demand Register (Offset: 0x1C) to trigger FTGMAC030 for polling the receive descriptor

Receive procedures:

- Wait for interrupt
- When interrupt occurs, software checks if it is a receive interrupt. If ISR [0] = 1, it means that the packet has been successfully moved to the receive buffer. Then, software needs to fetch the receive descriptor to get the receive packet until the owner bit of the next receive descriptor does not belong to software.
- Software releases the receive descriptors to FTGMAC030 after accessing the received packet.
- If the receive automatic poll function is disabled, software needs to write Receive Poll Demand Register (offset: 0x1C) to trigger FTGMAC030 to poll the receive descriptor.

vangotech

8.7.5.3. Procedures of Entering and Exiting Wake-On-LAN

The procedure for entering the Wake-On-LAN mode is as follows:

- Stop TX descriptor preparation. The last descriptor must be end descriptor of packet.
- Poll DMA/FIFO State Register (0x40) to wait for empty TX FIFO
- Set TXDMA_EN (0x50.bit[0]) = 0
- Set TXMAC_EN (0x50.bit[2]) = 0 to stop transmission
- Set RXMAC_EN (0x50.bit[3]) = 0
- Poll the DMA/FIFO State Register (0x40) to wait for empty RX FIFO
- Set RXDMA_EN (0x50.bit[1]) = 0 to stop reception
- Program Wake-up Frame CRC Register (0x78), Wake-up Frame Byte Mask 1st Double-word Register (0x80), Wake-up Frame Byte Mask 2nd Double-word Register (0x84), Wake-up Frame Byte Mask 3rd Double-word Register (0x88), and Wake-up Frame Byte Mask 4th Double-word Register (0x8C), if software wants to support wake-up frame event
- Write 0xFFFF_FFFF to clear the Wake-On-LAN Status Register (0x74)
- Program WOL_TYPE into the Wake-On-LAN Register (0x70) to decide the WOL port type of FTGMAC030
- Program the requested wake-up events into the Wake-On-LAN Register (0x70) to let FTGMAC030 enter the Wake-On-LAN mode
- Set RXMAC_EN (0x50.bit[3]) = 1 to enable the reception

The procedure for exiting the Wake-On-LAN mode is as follows:

- Wait for occurrence of wake-up events. It means that users must monitor the "WOL" port until it asserts to high.
- Set RXMAC_EN (0x50.bit[3]) = 0 to stop the reception
- Read the Wake-On-LAN Status Register (0x74) to check which wake-up event happened
- Program the Wake-On-LAN Register (0x70) to let FTGMAC030 exit the Wake-On-LAN mode
- Set SW_RST (0x50.bit[31]) = 1 to reset FTGMAC030
- Check if SW_RST (0x50.bit[31]) = 0 to make sure that FTGMAC030 has finished reset

Re-initialize FTGMAC030 to transmit and receive packets

8.7.6. Programming Guidelines

8.7.6.1. Ethernet Frame Formats

Table 8-155. FTGMAC030 Frame Structure and Frame Formats

Preamble	SFD	DA	SA	LEN	LLC	PAD	CRC
7-byte	1-byte	6-byte	6-byte	2-byte	0-1500 bytes	0-46 bytes	4-byte

- Preamble - Seven identical bytes. The bits in each byte are 0xAA, transmitted from left to right. The preamble is not required for reception. A sequence of 56 bits alternating between 1 and 0 is used for synchronization. The preamble gives components in the network time to detect the presence of a signal, and read the signal before the frame data arrives.
- Start of Frame Delimiter (SFD) - One byte. This bit is 0xAB, transmitted from left to right. SFD is required for reception.
- Destination Address (DA) - Six bytes. DA may be an individual, a multicast, or a broadcast address.

The destination MAC address identifies the station or stations to receive the frame. The source MAC address identifies the station that originated the frame. The 802.3 standard permits these address fields to be either 2-byte or 6-byte in length, but virtually all Ethernet implementations use 6-byte addresses. A destination address may specify either an "individual address" destined for a single station, or a "multicast address" destined for a group of stations. A destination address of all 1-bit refers to all stations on the LAN and is called a "broadcast address".

- Source Address (SA) - Six bytes. The source MAC address identifies the station that originated the frame.
- Length (LEN) - Two bytes. The length value indicates the number of Logical Link Control (LLC) data bytes in the data field. If the value of this field is less than or equal to 1500, then the Length/Type field indicates the number of bytes in the subsequent MAC data field. If the value of this field is greater than or equal to 1536, then the Length/Type field indicates the nature of the MAC protocol (Protocol type).
- Logical Link Control (LLC) Data - The user data
- PAD - Pad to 46 bytes. If the user data from DA is less than 46 bytes, PAD is needed.
- Cyclic Redundancy Check (CRC) - Four bytes. CRC is a value computed as a function of all fields except the preamble, SFD, and CRC. This CRC field contains a 4-byte CRC value used for error checking. When a source station assembles a MAC frame, it performs a CRC calculation on all bits in the frame from destination MAC address through the Pad fields (All fields except the preamble, start of frame delimiter, and frame check sequence). The source station stores the value in this field and transmits it as part of the frame. When the destination station receives the frame, it performs an identical check. If the calculated value does not match the value in this field, the destination station assumes that an error has occurred during transmission and discards the frame.
- Inter-frame Gap - Ethernet devices must allow a minimum idle period between transmission of frames known as the Inter-frame Gap (IFG) or Inter-Packet Gap (IPG). IFG or IPG provides a brief recovery time between frames to allow devices to prepare for reception of the next frame. The minimum inter-frame gap is 96-bit times, which is 9.6 microseconds for 10 Mbps Ethernet, 0.96 microseconds for 100 Mbps Ethernet, and 0.096 microseconds for 1 Gbps Ethernet.

The transmission order in the MII interface is Preamble → SFD → DA → SA → LEN → LLC → PAD (If necessary) → CRC. The most significant byte of each item is transmitted first. The most significant bit of each byte is transmitted first.

8.7.6.2. MII Management Interface

FTGMAC030 contains an MII Management Interface for an MII compliant PHY device. This allows control and status parameters to be passed between FTGMAC030 and PHY by MDIO and MDC; thereby, reducing the number of control pins required for the PHY mode control. The protocol consists of the bit stream that is sampled at the rising edge of MDC; the bit stream format is described below.

Besides, the IEEE 802.3 specifies the minimum period is 400 ns of MDC. Users need to set the MDC_CYCTHR field of PHY control register (offset 0x60) to meet the requirement.

Table 8-156. PHY Bit Stream Format for 802.3 clause 22.

	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE

READ	1…1	01	10	AAAAA	RRRRR	Z0	D..D (16)	Z
WRITE	1…1	01	01	AAAAA	RRRRR	10	D..D (16)	Z

- PRE (Preamble) - At the beginning of each transaction, FTGMAC030 sends a sequence of 32 contiguous logic bits on MDIO with 32 corresponding cycles on MDC to establish synchronization.
- ST (Start of Frame) - A start of frame with 01
- OP (Operation Code) - 10 denotes read; 01 denotes write.
- PHYAD (PHY Address) - A 5-bit address of PHY device. The first bit transmitted is MSB
- REGAD (Register Address) - A 5-bit address of PHY register. The first bit transmitted is MSB
- TA (Turn Around) - The turn-around is a 2-bit time to avoid contention during a transaction
- DATA (Data) - The data field is 16-bit.
- IDLE (Idle) - The condition on MDIO is high impedance state.

Beside, IEEE 802.3 provides another extended MII management format at clause 45. PHY device can use more register space.

8.7.7. Register List

Module Name	Base Address
MAC_0(RGMII)	0x54100000
MAC_1(RMII)	0x54200000
MAC_2(MII)	0x54300000

Table 8-157. Register Summary Table

Base Offset (Hex)	Name	Size (Byte)	Type	Description	Default Value (Hex)
000	ISR	4	RW1C	Interrupt Status Register	00000000
004	IME	4	RW	Interrupt Enable Register	00000000
008	MAC_MADR	4	RW	MAC Most Significant Address Register	00000000
00C	MAC_LADR	4	RW	MAC Least Significant Address Register	00000000
010	MAHT0	4	RW	Multicast Address Hash Table 0 Register	00000000
014	MAHT1	4	RW	Multicast Address Hash Table 1 Register	00000000
018	NPTXPD	4	WO	Normal Priority Transmit Poll Demand Register	00000000
01C	RXPD	4	WO	Receive Poll Demand Register	00000000
020	NPTXR_BADR	4	RW	Normal Priority Transmit Ring Base Address Register	00000000
024	RXR_BADR	4	RW	Receive Ring Base Address	00000000

Vangotech

028	HPTXPD	4	WO	High Priority Transmit Poll Demand Register	00000000
02C	HPTXR_BADR	4	RW	High Priority Transmit Ring Base Address Register	00000000
030	TXITC	4	RW	TX Interrupt Timer Control Register	00000000
034	RXITC	4	RW	RX Interrupt Timer Control Register	00000000
038	APTC	4	RW	Automatic Polling Timer Control Register	00000000
03C	DBLAC	4	RW	DMA Burst Length and Arbitration Control Register	00000F00
040	DMAFIFOS	4	RO	DMA/FIFO State Register	0C000000
048	TPAFCR	4	RW	Transmit Priority Arbitration and FIFO Control Register	000000F1
04C	RBSR	4	RW	Receive Buffer Size Register	00000640
050	MACCR	4	RW	MAC Control Register	00000000
054	MACSR	4	RW1C	MAC Status Register	00000000
058	TM	4	RW	Test Mode Register	00000000
060	PHYCR	4	RW	PHY Control Register	00000034
064	PHYDATA	4	RO,RW	PHY Data Register	00000000
068	FCR	4	RW, RW1C	Flow Control Register	00000400
06C	BPR	4	RW	Back Pressure Register	00000200
070	WOLCR	4	RW	Wake-On-LAN Control Register	00000000
074	WOLSR	4	RW1C	Wake-On-LAN Status Register	00000000
078	WFCRC	4	RW	Wake-up Frame CRC Register	00000000
080	WFBM1	4	RW	Wake-up Frame Byte Mask 1st Double-word Register	00000000
084	WFBM2	4	RW	Wake-up Frame Byte Mask 2nd Double-word Register	00000000
088	WFBM3	4	RW	Wake-up Frame Byte Mask 3rd Double-word Register	00000000
08C	WFBM4	4	RW	Wake-up Frame Byte Mask 4th Double-word Register	00000000
090	NPTXR_PTR	4	RO	Normal Priority Transmit Ring Pointer Register	00000000
094	HPTXR_PTR	4	RO	High Priority Transmit Ring Pointer Register	00000000
098	RXR_PTR	4	RO	Receive Ring Pointer Register	00000000
0A0	TX_CNT1	4	RO	TPKT_CNT Counter Register	00000000
0A4	TX_CNT2	4	RO	TXMCOL_CNT and TXSCOL_CNT Counter Register	00000000

Vangotech

0A8	TX_CNT3	4	RO	TXECOL_CNT and TXFAIL_CNT Counter Register	00000000
0AC	TX_CNT4	4	RO	TXLCOL_CNT and TXUNDERUN_CNT Counter Register	00000000
0B0	RX_CNT1	4	RO	RPKT_CNT Counter Register	00000000
0B4	RX_CNT2	4	RO	BROPKT_CNT Counter Register	00000000
0B8	RX_CNT3	4	RO	MULPKT_CNT Counter Register	00000000
0BC	RX_CNT4	4	RO	RPF_CNT and AEP_CNT Counter Register	00000000
0C0	RX_CNT5	4	RO	RUNT_CNT Counter Register	00000000
0C4	RX_CNT6	4	RO	CRCER_CNT and FTL_CNT Counter Register	00000000
0C8	RX_CNT7	4	RO	RCOL_CNT and RLOST_CNT Counter Register	00000000
0CC	BISTR	4	RO,RW	BIST Pattern Control Register	00000000
0D0	BM_THR_CTRL	4	RW	Broadcast and Multicast packet threshold control	00000000
0E0	ERCR	4	RW	Error Response Control Register	00000000
0E4	ACIR	4	RO,RW	AXI Additional Control Information Register	00000000
0E8	GISR	4	RW	MAC Interface Selection Register	0/1/2
0EC	SCCR	4	RW	SW Reset Cycle Count Register	000000f1
0F0	ECT	4	RW	EEE Control Register	00000810
0F4	REVR	4	RO	Revision Register	00010400
0F8	FEAR	4	RO	Feature Register	See Table 8-213
100	PTP_RX_UnDA	4	RW	PTP RX Unicast IP Destination Address Register	00000000
104	PTP_TX_UnDA	4	RW	PTP TX Unicast IP Destination Address Register	00000000
110	PTP_TX_PSEC	4	RO	PTP TX Event Frame Register for seconds	00000000
114	PTP_TX_PNSEC	4	RO	PTP TX Event Frame Register for nanoseconds	00000000
118	PTP_RX_PSEC	4	RO	PTP RX Event Frame Register for seconds	00000000
11C	PTP_RX_PNSEC	4	RO	PTP RX Event Frame Register for nanoseconds	00000000
120	PTP_TX_P_PSEC	4	RO	PTP TX Peer Frame Register for seconds	00000000
124	PTP_TX_P_	4	RO	PTP TX Peer Frame Register for	00000000

Vangotech

	PNSEC			nanoseconds	
128	PTP_RX_P_PSEC	4	RO	PTP RX Peer Frame Register for seconds	00000000
12C	PTP_RX_P_PNSEC	4	RO	PTP RX Peer Frame Register for nanoseconds	00000000
130	PTP_TMR0	4	RO	PTP Timer Register for nanoseconds	00000000
134	PTP_TMR1	4	RO	PTP Timer Register for nanoseconds	00000000
138	PTP_TMR2	4	RO	PTP Timer Register for seconds	00000000
13C	PTP_PER0	4	RW	PTP Period Increment 0 Register	00000008
140	PTP_PER1	4	RW	PTP Period Increment 1 Register	00000000
144	PTP_OFF	4	RW	PTP Period Offset Increment Register	00000000
148	PTP_ADJ	4	RW	PTP Timer Adjustment Register	00000000

8.7.8. Register Definition

8.7.8.1. Interrupt Status Register

Table 8-158. Interrupt Status Register, (ISR, Offset: 0x0)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
24	PDELAY_RESP_OUT	RW1C	PTP pdelay_resp frame transmitted	0
23	PDELAY_RESP_IN	RW1C	PTP pdelay_resp frame received	0
22	PDELAY_REQ_OUT	RW1C	PTP pdelay_req frame transmitted	0
21	PDELAY_REQ_IN	RW1C	PTP pdelay_req frame received	0
20	DELAY_REQ_OUT	RW1C	PTP delay_req frame transmitted	0
19	DELAY_REQ_IN	RW1C	PTP delay_req frame received	0
18	SYNC_OUT	RW1C	PTP sync frame transmitted	0
17	SYNC_IN	RW1C	PTP sync frame received	0
16	TSU_SEC_INC	RW1C	PTP Tsu second register indication	0
[15:13]	-	Rsvd	Reserved	0
12	RX_LPI_IN	RW1C	RX LPI mode enter	0
11	RX_LPI_EXIT	RW1C	RX LPI mode exit	0
10	HPTXBUF_UNAVA	RW1C	High-priority transmit buffer unavailable	0
9	PHYSTS_CHG	RW1C	PHY link status changing	0/1[1]
8	BUS_ERR	RW1C	Bus error indication When Master receives the ERROR response at AHB or the SLVERR or DECERR response at AXI.	0
7	TPKT_LOST	RW1C	Packets transmitted to Ethernet are lost due to late collision, excessive collision, or under-run.	0
6	NPTXBUF_UNAVA	RW1C	Normal priority transmit buffer unavailable	0
5	TPKT2F	RW1C	TXDMA has moved data into the TX FIFO.	0

Vangotech

4	TPKT2E	RW1C	Packets transmitted to Ethernet successfully	0
3	RPKT_LOST	RW1C	Received packet is lost due to RX FIFO full.	0
2	RXBUF_UNAVA	RW1C	Receiving buffer unavailable	0
1	RPKT2F	RW1C	Packets received into RX FIFO successfully	0
0	RPKT2B	RW1C	RXDMA has received packets to the RX buffer successfully.	0

[1]Due to the status reflect reflects the external PHY link status changing, so the default value may be 0 or 1.

8.7.8.2. Interrupt Enable Register

Table 8-159. Interrupt Enable Register (IME, Offset: 0x4)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
24	PDELAY_RESP_OUT_EN	RW	Interrupt enable of ISR [24]	0
23	PDELAY_RESP_IN_EN	RW	Interrupt enable of ISR [23]	0
22	PDELAY_REQ_OUT_EN	RW	Interrupt enable of ISR [22]	0
21	PDELAY_REQ_IN_EN	RW	Interrupt enable of ISR [21]	0
20	DELAY_REQ_OUT_EN	RW	Interrupt enable of ISR [20]	0
19	DELAY_REQ_IN_EN	RW	Interrupt enable of ISR [19]	0
18	SYNC_OUT_EN	RW	Interrupt enable of ISR [18]	0
17	SYNC_IN_EN	RW	Interrupt enable of ISR [17]	0
16	TSU_SEC_INC_EN	RW	Interrupt enable of ISR [16]	0
[15:13]	-	Rsvd	Reserved	0
12	RX_LPI_IN_EN	RW	Interrupt enable of ISR [12]	0
11	RX_LPI_EXIT_EN	RW	Interrupt enable of ISR [11]	0
10	HPTXBUF_UNAVA_EN	RW	Interrupt enable of ISR [10]	0
9	PHYSTS_CHG_EN	RW	Interrupt enable of ISR [9]	0
8	BUS_ERR_EN	RW	Interrupt enable of ISR [8]	0
7	TPKT_LOST_EN	RW	Interrupt enable of ISR [7]	0
6	NPTXBUF_UNAVA_EN	RW	Interrupt enable of ISR [6]	0
5	TPKT2F_EN	RW	Interrupt enable of ISR [5]	0
4	TPKT2E_EN	RW	Interrupt enable of ISR [4]	0
3	RPKT_LOST_EN	RW	Interrupt enable of ISR [3]	0
2	RXBUF_UNAVA_EN	RW	Interrupt enable of ISR [2]	0
1	RPKT2F_EN	RW	Interrupt enable of ISR [1]	0
0	RPKT2B_EN	RW	Interrupt enable of ISR [0]	0

8.7.8.3. Most Significant Address Register

Table 8-160. Most Significant Address Register (MAC_MADR, Offset: 0x8)

Bit	Name	Type	Description	Default Value

Vangotech

[31:16]	-	Rsvd	Reserved	0
[15:0]	MAC_MADR	RW	The most significant two bytes of the MAC address	0

8.7.8.4. Least Significant Address Register

Table 8-161. Least Significant Address Register (MAC_LADR, Offset: 0xC)

Bit	Name	Type	Description	Default Value
[31:0]	MAC_LADR	RW	The least significant four bytes of the MAC address	0

8.7.8.5. Multicast Address Hash Table 0 Register

Table 8-162. Multicast Address Hash Table 0 Register (MAHT0, Offset: 0x10)

Bit	Name	Type	Description	Default Value
[31:0]	MAHT0	RW	Multicast address hash table bytes 3 ~ 0 (Hash table 31:0)	0

8.7.8.6. Multicast Address Hash Table 1 Register

Table 8-163. Multicast Address Hash Table 1 Register (MAHT1, Offset: 0x14)

Bit	Name	Type	Description	Default Value
[31:0]	MAHT1	RW	Multicast address hash table bytes 3 ~ 0 (Hash table 63:32)	0

8.7.8.7. Normal Priority Transmit Poll Demand Register

Table 8-164. Normal Priority Transmit Poll Demand Register (NPTXPD, Offset: 0x18)

Bit	Name	Type	Description	Default Value
[31:0]	TXPD	WO	When writing a value to this register, FTGMAC030 reads the normal-priority transmit descriptor and checks the txdma_own bit. If txdma_own = 1, it will move the transmit buffer data to TX FIFO.	0

8.7.8.8. Receive Poll Demand Register

Table 8-165. Receive Poll Demand Register (RXPD, Offset: 0x1C)

Bit	Name	Type	Description	Default Value
[31:0]	RXPD	WO	When writing a value to this register, FTGMAC030 reads the receive descriptor and checks the rxdma_own bit. If rxdma_own = 1, it will move the receive packet data from the RX FIFO to the receiving buffer in the system memory.	0

8.7.8.9. Normal Priority Transmit Ring Base Address Register

Table 8-166. Normal Priority Transmit Ring Base Address Register (NPTXR_BADR, Offset: 0x20)

Bit	Name	Type	Description	Default

				Value
[31:0]	TXR_BADR	RW	Base address of the normal priority transmit ring The base address must be 16-byte aligned. FTGMAC030 treats the base address bits 3 ~ 0 as 0 when reading descriptors if bits 3 ~ 0 are not zero.	0

8.7.8.10. Receive Ring Base Address Register

Table 8-167. Receive Ring Base Address Register (RXR_BADR, Offset: 0x24)

Bit	Name	Type	Description	Default Value
[31:0]	RXR_BADR	RW	Base address of the receive ring The base address must be 16-byte aligned. FTGMAC030 treats the base address bits 3 ~ 0 as 0 when reading descriptors if bits 3 ~ 0 are not zero.	0

8.7.8.11. High Priority Transmit Poll Demand Register

Table 8-168. High Priority Transmit Poll Demand Register (HPTXPD, Offset: 0x28)

Bit	Name	Type	Description	Default Value
[31:0]	HPTXPD	WO	When writing a value to the register, FTGMAC030 reads the high-priority transmit descriptor process and checks the txdma_own bit. If txdma_own = 1, it will move the transmit buffer data into TX FIFO.	0

8.7.8.12. High Priority Transmit Ring Base Address Register

Table 8-169. High Priority Transmit Ring Base Address Register (HPTXR_BADR, Offset: 0x2C)

Bit	Name	Type	Description	Default Value
[31:0]	HPTXR_BADR	RW	Base address of the high-priority transmit ring The base address must be 16-byte aligned. FTGMAC030 treats the base address bits 3 ~ 0 as 0 when reading descriptors if bits 3 ~ 0 are not zero.	0

8.7.8.13. TX Interrupt Timer Control Register

Table 8-170. TX Interrupt Timer Control Register (TXITC, Offset: 0x30)

Bit	Name	Type	Description	Default Value
[31:17]	-	Rsvd	Reserved	0

16	TXINT_TIME_SEL	RW	This field defines the period of TX cycle time. When set, the TX cycle times are: 1000 Mbps mode 16.384 μs 100 Mbps mode 81.92 μs 10 Mbps mode 819.2 μs When cleared, the TX cycle times are: 1000 Mbps mode 1.024 μs 100 Mbps mode 5.12 μs 10 Mbps mode 51.2 μs	0
[15:8]	TXINT_CYC	RW	This field defines the maximum wait time to issue the transmit interrupt after a packet has been transmitted by FTGMAC030. The time unit is 1 TX cycle time. When TXINT_CYC = 0, the function will be disabled. When TXINT_THR = 0 and TXINT_CYC = 0, FTGMAC030 will issue a transmit interrupt or will not depend on TXIC in TXDES1.	0
7	-	Rsvd	Reserved	0
[6:4]	TXINT_THR	RW	This field defines the maximum number of transmit interrupts that are pending before an interrupt is generated. When TXINT_THR is not equal to 0, FTGMAC030 will issue a transmit interrupt if the transmit packet number transmitted by FTGMAC030 reaches TXINT_THR. When TXINT_THR = 0 and TXINT_CYC = 0, FTGMAC030 will issue a transmit interrupt or will not depend on TXIC in TXDES1.	0
[3:2]	-	Rsvd	Reserved	0
[1:0]	TXINT_THR_UNIT	RW	This field defines the unit of TXINT_THR. 0b00 Unit is 1 packet. 0b01 Unit is 4 packets. 0b10 Unit is 16 packets. 0b11 Unit is 64 packets.	0

The TX Interrupt Timer Control Register allows the software driver to reduce the number of transmit interrupts (ISR[4]) by setting the register. This lowers the CPU utilization for handling a large number of interrupts.

The register defines the threshold values for the transmit packet number, and one associated timer. The threshold value defines the maximum number of the transmit interrupts that can be pending before an interrupt is generated. The timer defines the maximum wait time to issue the transmit interrupt after a packet has been transmitted by FTGMAC030. The threshold value and timer combination allow batching of several packets into a single interrupt with a limit for how long it will be pending. The combination prevents throughput from being impeded in the heavy traffic, and the time limit prevents resources from

Vangotech

being held for too long in the low traffic.

The mitigation mechanism is similar for both the receive and transmit interrupts. There is a "transmitted packet counter" in FTGMAC030 to count the packets transmitted by FTGMAC030. When the counter reaches TXINT_THR, and TXINT_THR is not equal to 0, FTGMAC030 will issue the transmit interrupt. And the counter will be cleared when the transmit interrupt is issued.

Below lists the condition for FTGMAC030 to issue a transmit interrupt.

TXINT_THR	TXINT_CYC	FTGMAC030 Action
Zero	Zero	·The transmit interrupt will be issued after a packet is transmitted and TXIC of the packet is set. ·Clear the transmitted packet counter
Zero	Non-zero	·The transmit interrupt will be issued after a packet is transmitted and timer reaches the value of TXINT_CYC. ·Clear the transmitted packet counter
Non-zero	Zero	·The transmit interrupt will be issued if TXPKT_CNT = TXINT_THR. ·Clear the transmitted packet counter
Non-zero	Non-zero	·The transmit interrupt will be issued if the following condition holds: TXPKT_CNT = TXINT_THR TXPKT_CNT = 1 and timer reaches the value of TXINT_CYC. ·Clear the transmitted packet counter

8.7.8.14. RX Interrupt Timer Control Register

Table 8-171. RX Interrupt Timer Control Register (RXITC, Offset: 0x34)

Bit	Name	Type	Description	Default Value
[31:28]	-	Rsvd	Reserved	0
[27:20]	RXINT_RST	RW	This field defines the refresh time. The reset counter will be cleared once one packet is received by FTGMAC030. The time unit is 1 RX cycle time. When RXINT_CYC = 0, the function will be disabled. If RXINT_THR = 0 and RXINT_CYC = 0, a receive interrupt will be issued when a packet is received by FTGMAC030.	0
[19:17]	-	Rsvd	Reserved	0
16	RXINT_TIME_SEL	RW	This field defines the period of RX cycle time. When set, the RX cycle times are: 1000 Mbps mode -> 16.384 μs 100 Mbps mode -> 81.92 μs 10 Mbps mode -> 819.2 μs When cleared, the RX cycle times are: 1000 Mbps mode -> 1.024 μs 100 Mbps mode -> 5.12 μs	0

Vangotech

			10 Mbps mode -> 51.2 μs	
[15:8]	RXINT_CYC	RW	RXINT_CYC This field defines the maximum wait time to issue a receive interrupt after a packet has been received by FTGMAC030. The time unit is 1 RX cycle time. When RXINT_CYC = 0, the function is disabled. If RXINT_THR = 0 and RXINT_CYC = 0, a receive interrupt will be issued when a packet is received by FTGMAC030.	0
7	-	Rsvd	Reserved	0
[6:4]	RXINT_THR	RW	This field defines the maximum number of receive interrupts that are pending before an interrupt is generated. When RXINT_THR is not equal to 0, FTGMAC030 will issue a receive interrupt if the receive packet number received by FTGMAC030 reaches RXINT_THR. If RXINT_THR = 0 and RXINT_CYC = 0, a receive interrupt will be issued when FTGMAC030 finishes receiving a receive packet.	0
[3:2]	-	Rsvd	Reserved	0
[1:0]	RXINT_THR_UNIT	RW	This field defines the unit of RXINT_THR. 0b00 Unit is 1 packet. 0b01 Unit is 4 packets. 0b10 Unit is 16 packets. 0b11 Unit is 64 packets.	0

The RX Interrupt Timer Control Register allows the software driver to reduce the number of the receive interrupts (ISR[0]) by setting the register. This lowers the CPU utilization for handling a large number of interrupts.

The register define threshold values for the receive packet number, and one associated timer. The threshold value defines the maximum number of receive interrupts that can be pending before an interrupt is generated. The timer defines the maximum wait time to issue the receive interrupt after a packet has been received by FTGMAC030. The threshold value and timer combination allow batching of several packets into a single interrupt with a limit for how long it will be pending. The combination prevents throughput from being impeded in the heavy traffic, and the time limit prevents resources from being held for too long in the low traffic.

There is a "Received packet counter" in FTGMAC030 to count the packets received by FTGMAC030. When the counter reaches RXINT_THR, and RXINT_THR is not equal to 0, FTGMAC030 will issue a receive interrupt. And the counter will be cleared when the receive interrupt is issued.

Below lists the condition for FTGMAC030 to issue a receive interrupt.

RXINT_THR	RXINT_CYC	FTGMAC030 Action
Zero	Zero	·The receive interrupt will be issued after a packet is received by FTGMAC030. ·Clear the received packet counter
Zero	Non-zero	·The receive interrupt will be issued after a packet is received by FTGMAC030 and timer reaches the value of RXINT_CYC. ·Clear the received packet counter
Non-zero	Zero	·The receive interrupt will be issued if the received packet counter = RXINT_THR. ·Clear the received packet counter
Non-zero	Non-zero	·The receive interrupt will be issued if the following conditions: The received packet counter = RXINT_THR The received packet counter = 1 and timer reaches the value of RXINT_CYC ·Clear the received packet counter

8.7.8.15. Automatic Polling Timer Control Register

This register allows FTGMAC030 to automatically poll the descriptors. This lowers the CPU utilization. If the transmit automatic poll function is enabled, FTGMAC030 will automatically poll the transmit descriptor each time the transmit automatic poll timer expires. If the function is disabled, software will write the Transmit Poll Demand register (Offset: 0x18) to trigger FTGMAC030 for reading the transmit descriptors after software has prepared the transmit packets in the transmit buffers.

If the receive automatic poll function is enabled, FTGMAC030 will automatically poll the receive descriptor each time the receive automatic poll timer expires. If the function is disabled, software will write the Receive Poll Demand register (Offset: 0x1C) to trigger FTGMAC030 for reading the receive descriptors after software has released the receive descriptors to FTGMAC030.

Table 8-172. Automatic Polling Timer Control Register (APTC, Offset: 0x38)

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	TXPOLL_TIME_SEL	RW	This field defines the period of TX poll time. When set, the TX poll times are: 1000 Mbps mode 16.384 μs 100 Mbps mode 81.92 μs 10 Mbps mode 819.2 μs When cleared, the TX poll times are: 1000 Mbps mode 1.024 μs 100 Mbps mode 5.12 μs 10 Mbps mode 51.2 μs	0

[11:8]	TXPOLL_CNT	RW	This field defines the period of transmit automatic poll time. The unit is 1 TX poll time. When TXPOLL_CNT is not equal to 0, FTGMAC030 will automatically poll the transmit descriptor. If TXPOLL_CNT = 0, FTGMAC030 will not automatically poll the transmit descriptor.	0
[7:5]	-	Rsvd	Reserved	0
4	RXPOLL_TIME_SEL	RW	This field defines the period of RX poll time. When set, the RX poll times are: 1000 Mbps mode 16.384 μs 100 Mbps mode 81.92 μs 10 Mbps mode 819.2 μs When cleared, the RX poll times are: 1000 Mbps mode 1.024 μs 100 Mbps mode 5.12 μs 10 Mbps mode 51.2 μs	0
[3:0]	RXPOLL_CNT	RW	This field defines the period of receive automatic poll time. The unit is 1 RX poll time. When RXPOLL_CNT is not equal to 0, FTGMAC030 will automatically poll the receive descriptor. If RXPOLL_CNT = 0, FTGMAC030 will not automatically poll the receive descriptor.	0

8.7.8.16. DMA Burst Length and Arbitration Control Register

Table 8-173. DMA Burst Length and Arbitration Control Register (DBLAC, Offset: 0x3C)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
23	IFG_INC	RW	IFG (Inter-Frame Gap) increase The field defines the increase or decrease of IFG in Ethernet. When IFG_INC =1, the IFG will increase. When IFG_INC =0, the IFG will decrease.	0
[22:20]	IFG_CNT	RW	IFG (Inter-Frame Gap) count The field defines the increased or decreased number of IFG in Ethernet. When IFG_INC = 1, IFG will increase. When IFG_INC = 0, IFG will decrease. The unit is 1 transmit clock in Ethernet (8 ns in 1000Mbps mode, 40 ns in 100Mbps mode, and 400 ns in 10Mbps mode). When in the 1000Mbps mode and IFG_INC = 0, the value in the field will not be set to be more than 2. For example: When IFG_CNT = 1 and IFG_INC = 1 in 1000Mbps mode, IFG = 96 + 1x8 = 104 ns.	0

			When IFG_CNT = 1 and IFG_INC = 0 in the 1000Mbps mode, IFG = 96 -1x8 = 88 ns.	
[19:12]	-	Rsvd	Reserved	0
[11:10]	TXBST_SIZE	RW	TXDMA maximum burst size per TXDMA burst This field sets the maximum size of the TXDMA burst. The burst sizes are as follows: 0b00: 64 bytes 0b01: 128 bytes 0b10: 256 bytes 0b11: 512 bytes	3
[9:8]	RXBST_SIZE	RW	RXDMA maximum burst size per RXDMA burst This field sets the maximum size of RXDMA burst. The burst sizes are as follows: 0b00: 64 bytes 0b01: 128 bytes 0b10: 256 bytes 0b11: 512 bytes	3
7	-	Rsvd	Reserved	0
6	RX_THR_EN	RW	Enable the RX FIFO threshold arbitration	0
[5:3]	RXFIFO_HTHR	RW	RX FIFO high threshold value for arbitration When the used space in RX FIFO is larger than or equal to the RX FIFO high threshold value, RXDMA will be higher priority over TXDMA by using the DMA channel. RXDMA keeps the higher priority until the used space in RX FIFO is less than or equal to the RX FIFO low threshold value. Then, TXDMA gets higher priority over RXDMA. Consequently, software must set RXFIFO_HTHR to be larger than RXFIFO_LTHR to keep FTGMAC030 work correctly. 0: Threshold = 0 1: Threshold = 1/8 space of RX FIFO 2: Threshold = 2/8 space of RX FIFO 3: Threshold = 3/8 space of RX FIFO 4: Threshold = 4/8 space of RX FIFO 5: Threshold = 5/8 space of RX FIFO 6: Threshold = 6/8 space of RX FIFO 7: Threshold = 7/8 space of RX FIFO	0

Vangotech

[2:0]	RXFIFO_LTHR	RW	RX FIFO low threshold value for arbitration When the used space in RX FIFO is less than or equal to the RX FIFO low threshold value, TXDMA will have higher priority over RXDMA by using the DMA channel. 0: Threshold = 0 1: Threshold = 1/8 space of RX FIFO 2: Threshold = 2/8 space of RX FIFO 3: Threshold = 3/8 space of RX FIFO 4: Threshold = 4/8 space of RX FIFO 5: Threshold = 5/8 space of RX FIFO 6: Threshold = 6/8 space of RX FIFO 7: Threshold = 7/8 space of RX FIFO	0

8.7.8.17. DMA/FIFO State Register, DMAFIFOS

Table 8-174. DMA/FIFO State Register (DMAFIFOS, Offset: 0x40)

Bit	Name	Type	Description	Default Value
31	TXD_REQ	RO	TXDMA request	0
30	RXD_REQ	RO	RXDMA request	0
29	DARB_TXGNT	RO	TXDMA grant	0
28	DARB_RXGNT	RO	RXDMA grant	0
27	TXFIFO_EMPTY	RO	TX FIFO is empty.	1
26	RXFIFO_EMPTY	RO	RX FIFO is empty.	1
[25:22]	-	Rsvd	Reserved	0
[21:18]	TXDMA3_SM	RO	TXDMA 3 state machine The state machine is in charge of the read data flow from TX FIFO to TX pre-buffer.	0
[17:16]	TXDMA2_SM	RO	TXDMA 2 state machine The state machine is in charge of the burst read/write of transmit descriptor and buffer.	0
[15:12]	TXDMA1_SM	RO	TXDMA 1 state machine The state machine is in charge of the read/write of transmit descriptor and buffer.	0
11	-	Rsvd	Reserved	0
[10:8]	RXDMA3_SM	RO	RXDMA 3 state machine The state machine is in charge of RXDMA PVCI interface read/write.	0
[7:4]	RXDMA2_SM	RO	RXDMA 2 state machine The state machine is in charge of the burst read/write of receive descriptor and buffer.	0
[3:0]	RXDMA1_SM	RO	RXDMA 1 state machine	0

Vangotech

			The state machine is in charge of the read/write of receive descriptor and buffer.	

8.7.8.18. Transmit Priority Arbitration and FIFO Control Register

Table 8-175. Transmit Priority Arbitration and FIFO Control Register (TPAFCR, Offset: 0x48)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:27]	TFIFO_SIZE	RW	TX FIFO size Software can program this field to decide the TX FIFO size. Before programming this field, software must read the actual TX FIFO size in the FEATURE register. It is not allowed to program a value larger than the actual TX FIFO size. The FIFO sizes are as follows: 0: 2K 1: 4K 2: 8K 3: 16K 4: 32K 5: 64K 6 ~ 7: Reserved	0
[26:24]	RFIFO_SIZE	RW	RX FIFO size Software can program this field to decide the RX FIFO size. Before programming this field, software must read the actual RX FIFO size in the FEATURE register. It is not allowed to program a value of larger than the actual RX FIFO size. The FIFO sizes are as follows: 0: 2K 1: 4K 2: 8K 3: 16K 4: 32K 5: 64K 6 ~ 7: Reserved	0
[23:8]	-	Rsvd	Reserved	0
[7:4]	HPKT_THR	RW	High Priority Transmit Packet Threshold The hardware behavior is the same whether writing 0 or 1 to this field. Please refer to the following table for priority rule.	0xF
[3:0]	NPKT_THR	RW	Normal Priority Transmit Packet Threshold The hardware behavior is the same whether writing 0 or 1 to	1

Vangotech

			this field. Please refer to the following table for priority rule.	

The following table shows the arbitration between high priority and normal priority TX descriptor

High Priority TX Status	Normal Priority TX Status	Priority
HPKT_CNT < HPKT_THR	NPKT_CNT < NPKT_THR	High Priority TX descriptor is the highest priority.
HPKT_CNT < HPKT_THR	NPKT_CNT NPKT_THR	High Priority TX descriptor is the highest priority.
HPKT_CNT HPKT_THR	NPKT_CNT < NPKT_THR	Normal Priority TX descriptor is the highest priority.
HPKT_CNT HPKT_THR	NPKT_CNT NPKT_THR	High Priority TX descriptor is the highest priority.

8.7.8.19. Receive Buffer Size Register

Table 8-176. Receive Buffer Size Register (RBSR, Offset: 0x4C)

Bit	Name	Type	Description	Default Value
[31:14]	-	Rsvd	Reserved	0
[13:0]	RXBUF_SIZE	RW	Receive buffer size The unit is 1 byte. The receive buffer size must be 8-byte alignment.	0x640

8.7.8.20. MAC Control Register

Table 8-177. MAC Control Register (MACCR, Offset: 0x50)

Bit	Name	Type	Description	Default Value
31	SW_RST	RW	Software reset Writing 1 to this bit enables software reset. The software reset will last 560 system bus clocks, and then be auto-cleared.	0
[30:27]	-	Rsvd	Reserved	0
26	FULLDUP	RW	Full duplex If FULLDUP = 1, FTGMAC030 will be in the full-duplex mode; otherwise, FTGMAC030 will be in the half-duplex mode.	0
[25:24]	MAC_SPEED	RW	MAC operation mode 0b00: 10 Mbps 0b01: 100 Mbps 0b10: 1000 Mbps	0

Vango Tech

			0b11: Reserved	
23	-	Rsvd	Reserved	0
22	HPTXR_EN	RW	High priority transmit ring enable If HPTXR_EN = 1, software will use the high-priority transmit ring; otherwise, software will not use the high-priority transmit ring.	0
21	LOOP_EN	RW	Internal loopback enable When set, FTGMAC030 cannot transmit the packet to Ethernet and packet loopback to the RX path. This field cannot be software reset.	0
20	PTP_EN	RW	IEEE 1588 PTP TX/RX frame detection enable Besides, FTGMAC030 records the timestamp value when TX/RX PTP frame is detected. Note: The multicast PTP frame will be detected whatever the RX_MULTIPKT_EN (offset 0x50, bit[10]) value is.	0
19	-	Rsvd	Reserved	0
18	REMOVE_VLAN	RW	Remove the VLAN tag from the packets received with the VLAN tag.	0
17	CRC_APD	RW	Append CRC to transmitted packets	0
16	DISCARD_CRCERR	RW	Discard the CRC error packet if there is CRC error status in the transmit packet	0
15	-	Rsvd	Reserved	0
14	ENRX_IN_HALFTX	RW	Enable packet reception when transmitting packets in the half-duplex mode	0
13	JUMBO_LF	RW	Jumbo Long Frame When set, packets with length of more than 9216 (9220 for packets with VLAN tag) are treated as long frames. When cleared, packets with length of more than 1518 (1522 for packets with VLAN tag) are treated as long frames.	0
12	RX_RUNT	RW	Receive the incoming packet even if its length is less than 64 bytes The incoming packet length must be longer than or equal to 10 bytes.	0
11	RX_BROADPKT_EN	RW	Receives broadcast packets	0
10	RX_MULTIPKT_EN	RW	Receives all multicast packets	0
9	RX_HT_EN	RW	Enables storing incoming packet if the packet passes hash table address filtering and is a multicast packet	0
8	RX_ALLADR	RW	Destination address of the incoming packet is not checked.	0

Vangotech

7	DIS_IPV6_PKT_REC	RW	Disable IPV6 packet reorganization at RXMAC	0
[6:4]	-	Rsvd	Reserved	0
3	RXMAC_EN	RW	RXMAC enable When set, enable RXMAC to receive packets.	0
2	TXMAC_EN	RW	TXMAC enable When set, enable TXMAC to transmit packets.	0
1	RXDMA_EN	RW	Enable receive DMA channel If this bit is zero, reception will be stopped immediately.	0
0	TXDMA_EN	RW	Enable transmit DMA channel If this bit is zero, transmission will be stopped immediately.	0

8.7.8.21. MAC Status Register

Table 8-178. MAC Status Register (MACSR, Offset: 0x54)

Bit	Name	Type	Description	Default Value
[31:13]	-	Rsvd	Reserved	0
12	AFIFO_UNDERUN	RW1C	TX Asynchronous FIFO is under-run When the system frequency does not meet the clock limitation, then TX Asynchronous FIFO will cause the under-run condition.	0
11	COL_EXCEED	RW1C	Collision amount exceeds 16.	0
10	LATE_COL	RW1C	Transmitter detects late collision.	0
9	XPKT_LOST	RW1C	Packets transmitted to Ethernet are lost due to late collision or excessive collision.	0
8	XPKT_OK	RW1C	Packets transmitted to Ethernet successfully	0
7	RUNT	RW1C	Receiver detects a runt packet.	0
6	FTL	RW1C	Receiver detects a frame that is too long.	0
5	CRC_ERR	RW1C	The CRC check result of the incoming packet is invalid, unless the CRC_DIS bit is set.	0
4	RPKT_LOST	RW1C	Received packets are lost due to RX FIFO full.	0
3	RPKT_SAVE	RW1C	Packets received to RX FIFO successfully	0
2	COL	RW1C	Incoming packet dropped due to collision	0
1	BROADCAST	RW1C	Incoming packet for broadcast address	0
0	MULTICAST	RW1C	Incoming packet for multicast address	0

8.7.8.22. Test Mode Register

Table 8-179. Test Mode Register (TM, Offset: 0x58)

Bit	Name	Type	Description	Default Value

Vangotech

[31:21]	-	Rsvd	Reserved	0
20	PTIMER_TEST	RW	Automatic polling timer test mode When set to 1, TX and RX POLL TIME unit (offset 0x38) will increase the step, 1000Mbps mode 262.144 μs 100Mbps mode 1.3107 ms 10Mbps mode 13.107 ms	0
19	ITIMER_TEST	RW	Interrupt timer test mode When set to 1, TX and RX INT TIME unit (offset 0x30 and 0x34) will increase the step, 1000Mbps mode 262.144 μs 100Mbps mode 1.3107 ms 10Mbps mode 13.107 ms	0
[18:16]	-	Rsvd	Reserved	0
15	TEST_COL	RW	Transmit collision test mode When set to 1, the collision upper limit can be specified by TEST_EXSTHR. Besides, the back-off value can be specified by TEST_BKOFF.	0
[14:5]	TEST_BKOFF	RW	Back-off value in the transmission collision test mode When TEST_COL is set to 1, then the value must not set to 0.	0
[4:0]	TEST_EXSTHR	RW	Retry upper limit in the transmit collision test mode When TEST_COL is set to 1, then the value must not set to 0.	0

8.7.8.23. PHY Control register

Table 8-180. PHY Control Register (PHYCR, Offset: 0x60)

Bit	Name	Type	Description	Default Value
[31:28]	-	Rsvd	Reserved	0
27	PHYWR	RW	Setting this bit to 1 initializes a write sequence to PHY This bit will be automatically cleared after the write operation is finished.	0
26	PHYRD	RW	Setting this bit to 1 initializes a read sequence to PHY This bit will be automatically cleared after the read operation is finished.	0
[25:21]	REG_DEV_AD	RW	Register address for 802.3 Clause 22 Device address for 802.3 Clause 45	0
[20:16]	PHY_PRT_AD	RW	PHY address for 802.3 Clause 22 Port address for 802.3 Clause 45	0
[15:14]	OP	RW	Operation code	0
[13:12]	ST	RW	Start of frame	0
[11:8]	-	Rsvd	Reserved	0

Vangotech

[7:0]	MDC_CYCTHR	RW	MDC cycle threshold This field defines the period of MDC. The MDC period = MDC_CYCTHR x system clock period. Users must set the correct value before using MDC. Note: IEEE 802.3 specifies minimum cycle is 400 ns of MDC.	0x34

8.7.8.24. PHY Data Register

Table 8-181. PHY Data Register (PHYDATA, Offset: 0x64)

Bit	Name	Type	Description	Default Value
[31:16]	RDATA	RO	Read data from PHY	0
[15:0]	WDATA	RW	Write data to PHY	0

8.7.8.25. Flow Control Register

Table 8-182. Flow Control Register (FCR, Offset: 0x68)

Bit	Name	Type	Description	Default Value
[31:16]	PAUSE_TIME	RW	Pause time in the pause frame The unit is 1 slot time.	0
[15:9]	FC_HIGH/ FC_LOW	RW	RX FIFO free space high threshold A pause frame is sent with pause time = 0 when the RX FIFO free space is larger than the high threshold. The unit is 256 bytes, and the default value is 5. RX FIFO free space low threshold A pause frame is sent with the pause time set in bits 31 ~ 16 when the RX FIFO free space is less than the low threshold. The unit is 256 bytes, and the default value is 2. When FC_HTHR_SEL = 1, the RX FIFO free space high threshold is selected. When FC_HTHR_SEL = 0, RX FIFO free space low threshold is selected. The value that software programs in this field should be less than the RX FIFO size; besides, the value must not set to zero when the flow control is enabling.	2
8	FC_HTHR_SEL	RW	RX FIFO free space high threshold select When set, the RX FIFO free space high threshold is selected. When cleared, the RX FIFO free space low threshold is selected.	0
[7:5]	-	Rsvd	Reserved	0
4	RX_PAUSE	RW1C	Receive pause frame	0
3	TXPAUSED	RO	Packet transmission paused due to the receive pause frame	0
2	FCTHR_EN	RW	Enable flow control threshold mode	0

Vangotech

			This bit enables the transmit pause frame for high/low threshold. The function is valid only when FC_EN set to 1.	
1	TX_PAUSE	RW	Transmit pause frame Software can set this bit to send the pause frames. This bit is auto-cleared after the pause frame has been transmitted. The function is valid only when FC_EN set to 1.	0
0	FC_EN	RW	Flow control mode enable	0

8.7.8.26. Back Pressure Register

Table 8-183. Back Pressure Register (BPR, Offset: 0x6C)

Bit	Name	Type	Description	Default Value
[31:15]	-	Rsvd	Reserved	0
[14:8]	BK_LOW	RW	RX FIFO free space low threshold MAC generates a jam pattern if RX FIFO free space is less than the low threshold when packets are incoming. The unit is 256 bytes, and the default value is 2.	2
[7:4]	BKJAM_LEN	RW	Back pressure jam length 0: 4 bytes 6: 256 bytes 1: 8 bytes 7: 512 bytes 2: 16 bytes 8: 1024 bytes 3: 32 bytes 9: 1518 bytes 4: 64 bytes 10: 2048 bytes 5: 128 bytes Others: 4 bytes	0
[3:2]	-	Rsvd	Reserved	0
1	BKADR_MODE	RW	Back pressure address mode 1: Generate jam pattern when packet address matches 0: Generate jam pattern when any packet is incoming	0
0	BK_EN	RW	Back pressure mode enable	0

8.7.8.27. Wake-On-LAN Control Register

Table 8-184. Wake-On-LAN Control Register (WOLCR, Offset: 0x70)

Bit	Name	Type	Description	Default Value
[31:26]	-	Rsvd	Reserved	0
[25:24]	WOL_TYPE	RW	WOL output signal type 0b00: Active high 0b01: Active low 0b10: Positive pulse	0

Vangotech

			0b11: Negative pulse For the "Active high" and "Active low" usages, the level signal will be de-asserted by writing one WOLSR register. For the "Positive pulse" and "Negative pulse" usages, the pulse signal will assert 25 cycles of the RX clock domain and then be de-asserted by itself.	
[23:19]	-	Rsvd	Reserved	0
18	SW_PDNPHY	RW	Software power down PHY It will only inform PHY into power down and will not affect FTGMAC030.	0
[17:16]	WAKEUP_SEL	RW	Wake-up frame select This field is used to select the wake-up frame when the driver accesses the wake-up frame CRC or wake-up frame byte mask register. 0b00: Select wake-up frame 1 0b01: Select wake-up frame 2 0b10: Select wake-up frame 3 0b11: Select wake-up frame 4	0
[15:7]	-	Rsvd	Reserved	0
6	WAKEUP4_EN	RW	Enable wake-up frame 4 event	0
5	WAKEUP3_EN	RW	Enable wake-up frame 3 event	0
4	WAKEUP2_EN	RW	Enable wake-up frame 2 event	0
3	WAKEUP1_EN	RW	Enable wake-up frame 1 event	0
2	MAGICPKT_EN	RW	Enable magic packet event	0
1	LINKCHG1_EN	RW	Enable link change to 1 event	0
0	LINKCHG0_EN	RW	Enable link change to 0 event	0

8.7.8.28. Wake-On-LAN Status Register

Table 8-185. Wake-On-LAN Status Register (WOLSR, Offset: 0x74)

Bit	Name	Type	Description	Default Value
[31:7]	-	Rsvd	Reserved	0
6	WAKEUP4_STS	RW1C	Wake-up frame 4 event status	0
5	WAKEUP3_STS	RW1C	Wake-up frame 3 event status	0
4	WAKEUP2_STS	RW1C	Wake-up frame 2 event status	0
3	WAKEUP1_STS	RW1C	Wake-up frame 1 event status	0
2	MAGICPKT_STS	RW1C	Magic packet event status	0
1	LINKCHG1_STS	RW1C	Link change to 1 event status	0
0	LINKCHG0_STS	RW1C	Link change to 0 event status	0

vangotech

8.7.8.29. Wake-up Frame CRC Register

Table 8-186. Wake-up Frame CRC Register (WFCRC, Offset: 0x78)

Bit	Name	Type	Description	Default Value
[31:0]	WFCRC	RW	Wake-up frame CRC value When WAKEUP_SEL = 0b00, the write value is to be stored in the wake-up frame 1 CRC register. The read value is from the wake-up frame 1 CRC register. When WAKEUP_SEL = 0b01, the write value is to be stored in the wake-up frame 2 CRC register. The read value is from the wake-up frame 2 CRC register. When WAKEUP_SEL = 0b10, the write value is to be stored in the wake-up frame 3 CRC register. The read value is from the wake-up frame 3 CRC register. When WAKEUP_SEL = 0b11, the write value is to be stored in the wake-up frame 4 CRC register. The read value is from the wake-up frame 4 CRC register.	0

8.7.8.30. Wake-up Frame Byte Mask 1st Double-word Register

Table 8-187. Wake-up Frame Byte Mask 1st Double-word Register (WFBM1, Offset: 0x80)

Bit	Name	Type	Description	Default Value
[31:0]	WFBM1	RW	Wake-up frame byte mask 1st double-word This field stores the byte mask of the wake-up frame from byte 1 to byte 32. Bit 0 is the byte 1 mask and bit 31 is the byte 32 mask. When WAKEUP_SEL = 0b00, the write value is to be stored in the wake-up frame 1 byte mask 1st double-word register. The read value is from wake-up frame 1 byte mask 1st double-word register. When WAKEUP_SEL = 0b01, the write value is to be stored in the wake-up frame 2 byte mask 1st double-word register. The read value is from byte mask 1st double-word register of wake-up frame 2. When WAKEUP_SEL = 0b10, the write value is to be stored in the wake-up frame 3 byte mask 1st double-word register. The read value is from byte mask 1st double-word register of wake-up frame 3. When WAKEUP_SEL = 0b11, the write value is to be stored in the wake-up frame 4 byte mask 1st double-word register. The read value is from byte mask 1st double-word register of wake-up frame 4.	0

Vangotech

8.7.8.31. Wake-up Frame Byte Mask 2nd Double-word Register

Table 8-188. Wake-up Frame Byte Mask 2nd Double-word Register (WFBM2, Offset: 0x84)

Bit	Name	Type	Description	Default Value
[31:0]	WFBM2	RW	Wake-up frame byte mask 2nd double-word This field stores the byte mask of the wake-up frame from byte 33 to byte 64. Bit 0 is the byte 33 mask and bit 31 is the byte 64 mask. When WAKEUP_SEL = 0b00, the write value is to be stored in the wake-up frame 1 byte mask 2nd double-word register. The read value is from byte mask 2nd double-word register of wake-up frame 1. When WAKEUP_SEL = 0b01, the write value is to be stored in the wake-up frame 2 byte mask 2nd double-word register. The read value is from byte mask 2nd double-word register of wake-up frame 2. When WAKEUP_SEL = 0b10, the write value is to be stored in the wake-up frame 3 byte mask 2nd double-word register. The read value is from byte mask 2nd double-word register of wake-up frame 3. When WAKEUP_SEL = 0b11, the write value is to be stored in the wake-up frame 4 byte mask 2nd double-word register. The read value is from byte mask 2nd double-word register of wake-up frame 4.	0

8.7.8.32. Wake-up Frame Byte Mask 3rd Double-word Register

Table 8-189. Wake-up Frame Byte Mask 3rd Double-word Register (WFBM3, Offset: 0x88)

Bit	Name	Type	Description	Default Value
[31:0]	WFBM3	RW	Wake-up frame byte mask 3rd double-word This field stores the byte mask of the wake-up frame from byte 65 to byte 96. Bit 0 is the byte 65 mask and bit 31 is the byte 96 mask. When WAKEUP_SEL = 0b00, the write value is to be stored in the wake-up frame 1 byte mask 3rd double-word register. The read value is from byte mask 3rd double-word register of wake-up frame 1. When WAKEUP_SEL = 0b01, the write value is to be stored in the wake-up frame 2 byte mask 3rd double-word register. The read value is from byte mask 3rd double-word register of wake-up frame 2. When WAKEUP_SEL = 0b10, the write value is to be stored in the wake-up frame 3 byte mask 3rd double-word register. The read	0

Vangotech

			value is from byte mask 3rd double-word register of wake-up frame 3. When WAKEUP_SEL = 0b11, the write value is to be stored in the wake-up frame 4 byte mask 3rd double-word register. The read value is from byte mask 3rd double-word register of wake-up frame 4.	

8.7.8.33. Wake-up Frame Byte Mask 4th Double-word Register

Table 8-190. Wake-up Frame Byte Mask 4th Double-word Register (WFBM4, Offset: 0x8C)

Bit	Name	Type	Description	Default Value
[31:0]	WFBM4	RW	Wake-up frame byte mask 4th double-word This field stores the byte mask of the wake-up frame from byte 97 to byte 128. Bit 0 is the byte 97 mask and bit 31 is the byte 128 mask. When WAKEUP_SEL = 0b00, the write value is to be stored in the wake-up frame 1 byte mask 4th double-word register. The read value is from byte mask 4th double-word register of wake-up frame 1. When WAKEUP_SEL = 0b01, the write value is to be stored in the wake-up frame 2 byte mask 4th double-word register. The read value is from byte mask 4th double-word register of wake-up frame 2. When WAKEUP_SEL = 0b10, the write value is to be stored in the wake-up frame 3 byte mask 4th double-word register. The read value is from byte mask 4th double-word register of wake-up frame 3. When WAKEUP_SEL = 0b11, the write value is to be stored in the wake-up frame 4 byte mask 4th double-word register. The read value is from byte mask 4th double-word register of wake-up frame 4.	0

8.7.8.34. Normal Priority Transmit Ring Pointer Register

Table 8-191. Normal Priority Transmit Ring Pointer Register (NPTXR_PTR, Offset: 0x90)

Bit	Name	Type	Description	Default Value
[31:0]	NPTXR_PTR	RO	Normal Priority Transmit Ring Pointer Register This register indicates the current value of the transmit descriptor pointer for the normal priority transmit ring pointer register. When all descriptors of NPTXR are invalid (All TXDMA_OWN bits are zeros), then NPTXR_PTR can decrease one TX descriptor	0

			size.	

8.7.8.35. High Priority Transmit Ring Pointer Register

Table 8-192. High Priority Transmit Ring Pointer Register (HPTXR_PTR, Offset: 0x94)

Bit	Name	Type	Description	Default Value
[31:0]	HPTXR_PTR	RO	High Priority Transmit Ring Pointer Register This register indicates the current value of the transmit descriptor pointer for the high priority transmit ring pointer register. When all descriptors of HPTXR are invalid (All TXDMA_OWN bits are zeros), then HPTXR_PTR can decrease one TX descriptor size.	0

8.7.8.36. Receive Ring Pointer Register

Table 8-193. Receive Ring Pointer Register (RXR_PTR, Offset: 0x98)

Bit	Name	Type	Description	Default Value
[31:0]	RXR_PTR	RO	Receive Ring Pointer Register This register indicates the current value of the transmit descriptor pointer for the receive ring pointer register. When all descriptors of RXR are invalid (All RXPKT_RDY bit are ones), then RXR_PTR can decrease one RX descriptor size.	0

8.7.8.37. TPKT_CNT Counter Register

Table 8-194. TPKT_CNT Counter Register (TX_CNT1, Offset: 0xA0)

Bit	Name	Type	Description	Default Value
[31:0]	TPKT_CNT	RO	Counter for counting packets transmitted successfully	0

8.7.8.38. TXMCOL_CNT and TXSCOL_CNT Counter Register, TX_CNT2 (Offset: 0xA4)

Table 8-195. TXMCOL_CNT and TXSCOL_CNT Counter Register (TX_CNT2, Offset: 0xA4)

Bit	Name	Type	Description	Default Value
[31:16]	TXMCOL_CNT	RO	Counter for counting packets transmitted OK with 2 ~ 15 collisions	0
[15:0]	TXSCOL_CNT	RO	Counter for counting packets transmitted OK with single collision	0

8.7.8.39. TXECOL_CNT and TXFAIL_CNT Counter Register

Table 8-196. TXECOL_CNT and TXFAIL_CNT Counter Register (TX_CNT3, Offset: 0xA8)

Bit	Name	Type	Description	Default Value
[31:16]	TXFAIL_CNT	RO	Counter for counting packets failed in transmission (Due to the late collision or collision count ≥ 16 or transmit underrun)	0
[15:0]	TXECOL_CNT	RO	Counter for counting packets failed in transmission (Due to the collision count ≥ 16)	0

8.7.8.40. TXLCOL_CNT and TXUNDERUN_CNT Counter Register

Table 8-197. TXLCOL_CNT and TXUNDERUN_CNT Counter Register (TX_CNT4, Offset: 0xAC)

Bit	Name	Type	Description	Default Value
[31:16]	TXUNDERUN_CNT	RO	Counter for counting the packets failed in transmission (Due to transmit under-run)	0
[15:0]	TXLCOL_CNT	RO	Counter for counting the packets failed in transmission (Due to late collision)	0

8.7.8.41. RPKT_CNT Counter Register

Table 8-198. RPKT_CNT Counter Register (RX_CNT1, Offset: 0xB0)

Bit	Name	Type	Description	Default Value
[31:0]	RPKT_CNT	RO	Counter for counting the packets received successfully	0

8.7.8.42. BROPKT_CNT Counter Register

Table 8-199. BROPKT_CNT Counter Register (RX_CNT2, Offset: 0xB4)

Bit	Name	Type	Description	Default Value
[31:0]	BROPKT_CNT	RO	Counter for counting the received broadcast packets	0

8.7.8.43. MULPKT_CNT Counter Register

Table 8-200. MULPKT_CNT Counter Register (RX_CNT3, Offset: 0xB8)

Bit	Name	Type	Description	Default Value
[31:0]	MULPKT_CNT	RO	Counter for counting the received multicast packets	0

8.7.8.44. RPF_CNT and AEP_CNT Counter Register

Table 8-201. RPF_CNT and AEP_CNT Counter Register (RX_CNT4, Offset: 0xBC)

Bit	Name	Type	Description	Default

Vangotech

				Value
[31:16]	RPF_CNT	RO	Receive pause frame counter	0
[15:0]	AEP_CNT	RO	Counter for counting the packets with alignment error The counter is to count packets with CRC error and no octet-boundary discarded by FTGMAC030.	0

8.7.8.45. RUNT_CNT Counter Register

Table 8-202. RUNT_CNT Counter Register

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
[15:0]	RUNT_CNT	RO	Counter for counting the received runt packets	0

8.7.8.46. CRCER_CNT and FTL_CNT Counter Register

Table 8-203. CRCER_CNT and FTL_CNT Counter Register (RX_CNT6, Offset: 0xC4h)

Bit	Name	Type	Description	Default Value
[31:16]	CRCER_CNT	RO	Received CRC error packet counter The counter counts the number of the following received packets: 1. The octet-boundary frames discarded due to the CRC error. 2. The frames truncated due to RX FIFO full and it leads to CRC error.	0
[15:0]	FTL_CNT	RO	Counter for counting received FTL packets	0

8.7.8.47. RCOL_CNT and RLOST_CNT Counter Register

Table 8-204. RCOL_CNT and RLOST_CNT Counter Register (RX_CNT7, Offset: 0xC8)

Bit	Name	Type	Description	Default Value
[31:16]	RLOST_CNT	RO	Counter for counting the loss of the received packets (Due to RX FIFO full)	0
[15:0]	RCOL_CNT	RO	Receive collision counter	0

8.7.8.48. BIST Pattern Control Register

The BIST function can send one specific broadcast pattern from TX and compare the pattern from RX without DMA operation. When the BIST function is enabled, FTGMAC030 will force to the full-duplex mode.

The BIST function running step:

Step 1. Users must set to the correct MAC speed (bit [25:24], offset 0x50) to synchronize PHY and disable TX/RX DMA and MAC (bit [3:0], offset 0x50).

Step 2. Users set the BIST pattern control register for the test purpose.

Table 8-205. BIST Pattern Control Register (BISTR, Offset: 0xCC)

Bit	Name	Type	Description	Default Value
[31:10]	-	Rsvd	Reserved	0
9	RX_CRC_FAIL	RO	RX CRC compare fail It will only be valid when RX_COMPARE clears to 0.	0
8	RX_COMPARE	RW	BIST pattern comparison from RX side Once the comparison is complete then automatically clears to 0. Users must wait the bit to clear to 0, and then users can update the bit again. When RX_COMPARE is set to 1, then TX and RX sides must have the loopback path either the internal loopback or external loopback. The internal loopback is LOOP_EN (bit [21], offset 0x50); and the external loopback is PHY loopback mode.	0
[7:1]	-	Rsvd	Reserved	0
0	TX_SEND	RW	BIST Pattern sending enable When TX_SEND is set to 1, then FTGMAC030 sends the BIST pattern. Once the sending is complete then automatically clears to 0. Users must wait the bit to clear to 0, and then users can update the bit again.	0

8.7.8.49. Broadcast and Multicast Receiving Control Register

The register is used to control the broadcast and the multicast packet receiving function.

Table 8-206. Broadcast and Multicast Receiving Control Register (BMRCR, Offset: 0xD0)

Bit	Name	Type	Description	Default Value
[31:25]	-	Rsvd	Reserved	0
24	TIME_STEP	RW	Timer step setting When set to 0, then the time step shows as follows: 10M: 419.43 ms 100M: 41.94 ms 1G: 8.39 ms When set to 1, then the time step shows as follows: 10M: 409.6 us 100M: 40.96 us 1G: 8.19 us	0
[23:16]	TIME_THR_NUM	RW	Timer threshold number If PKT_THR_VAL is not zero, then FTGMAC030 will monitor the increasing number per TIME_THR_NUM; if the increasing number is larger than PKT_THR_VAL, then FTGMA030 automatically disables the broadcast and	0

Vangotech

			multicast packet receiving. The unit is the time step and depends on the speed mode.	
[15:5]	-	Rsvd	Reserved	0
[4:0]	PKT_T HR_VAL	RW	Multicast and broadcast packets threshold value When the value is zero, then it will not affect the broadcast and multicast packet receiving function. The unit is 256 packets.	0

8.7.8.50. Error Response Control Register

Table 8-207. Error Response Control Register (ERCR, Offset: 0xE0)

Bit	Name	Type	Description	Default Value
[31:2]	-	Rsvd	Reserved	0
1	RONLY_ERR_EN	RW	Read-only Error enable When users write the read-only register, then FTGMAC030 will respond "ERROR" if the bit is set to 1.	0
0	DEC_ERR_EN	RW	Decode Error enable When user write does not exist in the register, then FTGMAC030 will respond "ERROR" if the bit set to 1. When user read does not exist in the register, then FTGMAC030 responds "OKAY" no matter the value is 0 or 1.	0

8.7.8.51. AXI Additional Control Information Register

Table 8-208. AXI Additional Control Information Register (ACIR, Offset: 0xE4)

Bit	Name	Type	Description	Default Value
[31:28]	BResp_WAIT	RO	AXI BR channel waiting counter When the value is not zero, it indicates that the BR channel is still waiting write response.	0
[27:24]	WData_WAIT	RO	AXI WD channel waiting counter When the value is not zero, it indicates that the WD channel is still transferring write data.	0
[23:20]	RData_WAIT	RO	AXI RD channel waiting counter When the value is not zero, it indicates that the RD channel is still waiting read data.	0
[19:17]	-	Rsvd	Reserved	0
16	AXI_ACI_en	RW	When set to 1, then AXI ACI value is decided by users; otherwise, the AXI ACI value is decided by FTGMAC030.	0
[15:12]	AWCACHE	RW	User programmable CACHE port of the AW channel	0
11	-	Rsvd	Reserved	0
[10:8]	AWPROT	RW	User programmable PROT port of the AW channel	0

Vangotech

[7:4]	ARCACHE	RW	User programmable CACHE port of the AR channel	0
3	-	Rsvd	Reserved	0
[2:0]	ARPROT	RW	User programmable PROT port of the AR channel	0

8.7.8.52. MAC Interface Selection Register

Table 8-209. MAC Interface Selection Register (GISR, Offset: 0xE8)

Bit	Name	Type	Description	Default Value
[31:2]	-	Rsvd	Reserved	0
[1:0]	MACIF_SEL	RW	MAC interface selection The value records the GMAC interface selection. 0b00: MII 0b01: RMII 0b10: RGMII 0b11: Reserved	0:for MAC_2; 1:for MAC_1; 2:for MAC_0;

8.7.8.53. SW Reset Cycle Count Register

Table 8-210. SW Reset Cycle Count Register (SCCR, Offset: 0xEC)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
[15:0]	SW_CYC	RW	SW reset cycle count For SW reset synchronization to Ethernet clock domain The value must be non-zero.	0xf1

8.7.8.54. EEE Control Register

Table 8-211. EEE Control Register (ECT, Offset: 0xF0)

Bit	Name	Type	Description	Default Value
[31:17]	-	Rsvd	Reserved	0
16	Send_TX_LPI	RW	Send TX LPI pattern	0
[15:0]	TX_Wake_cnt	RW	TX EEE wake up counter For 100M mode, the min. value is 30 us. For 1000M mode, the min. value is 16.5 us.	0x810

8.7.8.55. Revision Register

Table 8-212. Revision Register (REVR, Offset: 0xF4)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:16]	REV_B1	RO	First digit of the revision tag	1

Vango-tech

			For example, if the revision tag is version_1_2_3, this filed will be 1.	
[15:8]	REV_B2	RO	Second digit of the revision tag For example, if the revision tag is version_1_2_3, this filed will be 2.	4
[7:0]	REV_B3	RO	Third digit of the revision tag For example, if the revision tag is version_1_2_3, this filed will be 3.	0

8.7.8.56. Feature Register

Table 8-213. Feature Register (FEAR, Offset: 0xF8)

Bit	Name	Type	Description	Default Value
[31:28]	-	Rsvd	Reserved	0
27	RGMII_IF	RO	RGMII Interface support	1：for MAC_0； 0：for others;
26	RMII_IF	RO	RMII Interface support	1：for MAC_1； 0：for others;
25	GMII_IF	RO	GMII Interface support	0
24	MII_IF	RO	MII Interface support	1: for MAC_2 0: for others
[23:19]	-	Rsvd	Reserved	0
18	IEEE_1588_PTP	RO	PTP function support	1
17	IP_COE	RO	TCP/IP checksum offload engine support	1
16	HOST_IF_SEL	RO	The host interface selection 0: AHB 32-bit data bus 1: AXI 64-bit data bus	1
[15:13]	-	Rsvd	Reserved	0
[12:8]	AXI_ID_WIDTH	RO	The width of AXI ID filed Please ignore it if the host interface is AHB.	8
7	-	Rsvd	Reserved	0
[6:4]	TFIFO_RSIZE	RO	The actual TX FIFO size The FIFO sizes are as follows: 0: 2K 1: 4K 2: 8K 3: 16K 4: 32K 5: 64K 6 ~ 7: Reserved	1: for MAC_1 and MAC_2; 4: for MAC_0;

Vangotech

3	-	Rsvd	Reserved	0
[2:0]	RFIFO_RSIZE	RO	The actual RX FIFO size The FIFO sizes are as follows: 0: 2K 1: 4K 2: 8K 3: 16K 4: 32K 5: 64K 6 ~ 7: Reserved	1: for MAC_1 and MAC_2; 4: for MAC_0;

8.7.8.57. PTP RX Unicast IP Destination Address Register

Table 8-214. PTP RX Unicast IP Destination Address Register (PTP_RX_UnDA, Offset: 0x100)

Bit	Name	Type	Description	Default Value
[31:0]	RDA	RW	Unicast IP destination address used for detection of PTP frames on the receive path	0

8.7.8.58. PTP TX Unicast IP Destination Address Register

Table 8-215. PTP TX Unicast IP Destination Address Register (PTP_TX_UnDA, Offset: 0x104)

Bit	Name	Type	Description	Default Value
[31:0]	TDA	RW	Unicast IP destination address used for detection of PTP frames on the transmit path	0

8.7.8.59. PTP TX Event Frame Register

FTGMAC030 detects the transmitted PTP frame and records the timestamp at the PTP TX Event Frame Register. The following PTP frames will be detected:

Sync Frame

Delay_Req Frame

Table 8-216. PTP TX Event Frame Register (PTP_TX, Offset: 0x110)

Bit	Name	Type	Description	Default Value
[31:0]	PSEC	RO	PTP TX frame timestamp for seconds	0

Table 8-217. PTP TX Event Frame Register (PTP_TX Offset: 0x114)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:0]	PNSEC	RO	PTP TX frame timestamp for nanoseconds	0

Vangotech

8.7.8.60. PTP RX Event Frame Register

FTGMAC030 detects the received PTP frame and records the timestamp at the PTP RX Event Frame Register. The following PTP frames will be detected:

Sync Frame

Delay_Req Frame

Table 8-218. PTP RX Event Frame Register, PTP_RX (Offset: 0x118)

Bit	Name	Type	Description	Default Value
[31:0]	PSEC	RO	PTP RX frame timestamp for seconds	0

Table 8-219. PTP RX Event Frame Register, PTP_RX (Offset: 0x11C)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:0]	PNSEC	RO	PTP RX frame timestamp for nanoseconds	0

8.7.8.61. PTP TX Peer Frame Register

FTGMAC030 detects the transmitted PTP peer frame and record the timestamp at the PTP TX Peer Frame Register. The following PTP peer frames will be detected:

Pdelay_Req Frame

Pdelay_Resp Frame

Table 8-220. PTP TX Peer Frame Register (PTP_TX_P, Offset: 0x120)

Bit	Name	Type	Description	Default Value
[31:0]	PSEC	RO	PTP TX peer frame timestamp for seconds	0

Table 8-221. PTP TX Peer Frame Register (PTP_TX_P, Offset: 0x124)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:0]	PNSEC	RO	PTP TX peer frame timestamp for nanoseconds	0

8.7.8.62. PTP RX Peer Frame Register

FTGMAC030 detects the received PTP Peer frame and records the timestamp at the PTP RX Peer Frame Register. The following PTP peer frames will be detected:

Pdelay_Req Frame

Pdelay_Resp Frame

Table 8-222. PTP RX Peer Frame Register (PTP_RX_P, Offset: 0x128)

Bit	Name	Type	Description	Default Value
[31:0]	PSEC	RO	PTP RX peer frame timestamp for seconds	0

Table 8-223. PTP RX Peer Frame Register (PTP_RX_P, Offset: 0x12C)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:0]	PNSEC	RO	PTP RX peer frame timestamp for seconds	0

8.7.8.63. PTP Timer Register, PTP_TMR

Table 8-224. PTP Timer Register (PTP_TMR, Offset: 0x130)

Bit	Name	Type	Description	Default Value
[31:0]	NNSEC	RW	Nano-Nanosecond timer Unit is 2-9 ns. One 32-bit timer When counting to 999999999 and then it returns to 0.	0x0

Table 8-225. PTP Timer Register (PTP_TMR, Offset: 0x134)

Bit	Name	Type	Description	Default Value
[31:30]	-	Rsvd	Reserved	0
[29:0]	NSEC	RW	Nanosecond timer One 30-bit timer When counting to 999999999 and then it returns to 0.	0x0

Table 8-226. PTP Timer Register (PTP_TMR, Offset: 0x138)

Bit	Name	Type	Description	Default Value
[31:0]	SEC	RW	Second timer Once the register is written, then update the timer value to the PTP timer.	0x0

8.7.8.64. PTP Period Increment 0 Register

Table 8-227. PTP Period Increment 0 Register (PTP_PER0, Offset: 0x13C)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	NS_PER	RW	Period The unit is nanosecond. (Max. is 255 ns.)	0x8

Vangotech

8.7.8.65. PTP Period Increment 1 Register

Table 8-228. PTP Period Increment 1 Register (PTP_PER1, Offset: 0x140)

Bit	Name	Type	Description	Default Value
[31:0]	NNS_PER	RW	Period The unit is 10^{-9} ns. (Max. is 0.999999999 ns.)	0x0

8.7.8.66. PTP Period Offset Increment Register

Table 8-229. PTP Period Offset Increment Register (PTP_OFF, Offset: 0x144)

Bit	Name	Type	Description	Default Value
[31:16]	INC_CYC	RW	Cycle number Set to 0 then disable the function	0x0
[15:0]	NS_PER	RW	Offset The unit is nanosecond.	0x0

8.7.8.67. PTP Timer Adjustment Register

Table 8-230. PTP Timer Adjustment Register (PTP_ADJ, Offset: 0x148)

Bit	Name	Type	Description	Default Value
[31]	INC	RW	Set to 0 indicates subtract Set to 1 indicates add.	0
[30]	-	Rsvd	Reserved	0
[29:0]	ADJ_VAL	RW	30-bit adjust value The unit is nanosecond. Set to 0 then disable the function	0

Vangotech

8.8. GPIO

8.8.1. Overview

The FTGPIO010 is a user-programmable general-purpose input/output controller. It is used to input/output data from the system and device. Each GPIO can be programmed as an input or output. GPIO can also be an interrupt input supporting at the rising edge, falling edge, both edges, and high/low level interrupt sense types.

GPIO provides up to 64 programmable I/O ports. Each port can be independently programmed.

8.8.2. Block Diagram

Figure 8-36 depicts the block diagram of the GPIO controller.

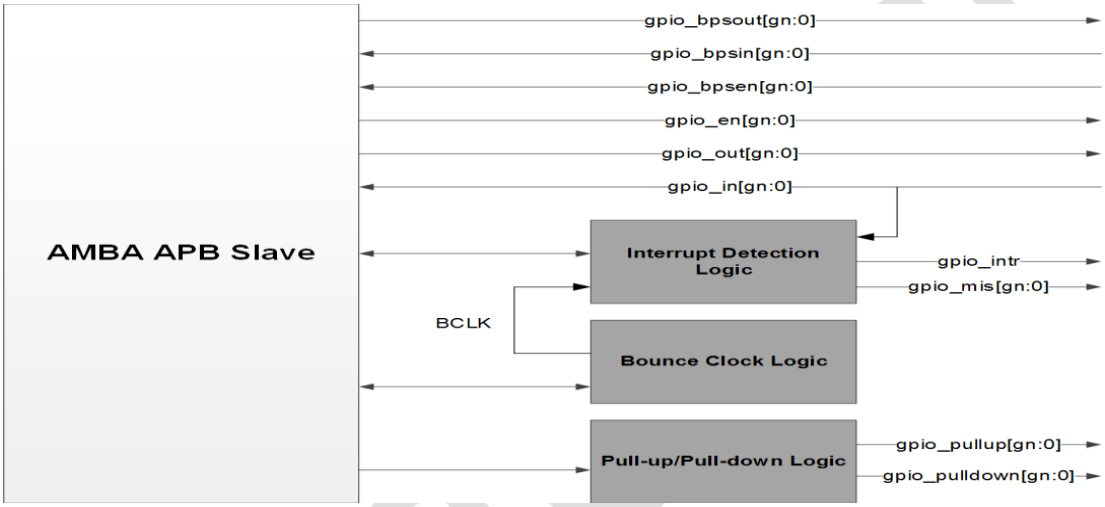


Figure 8-36. Block Diagram

The GPIO supports the data input/output, data bit set/clear, data bypass mode, variable interrupt detection, "Pull" function, and bounce clock. The gray blocks shown in Figure 8-36 can be removed by the hardware configurations. The pin number of the GPIO is 32.(gn=31)

8.8.2.1. Bypass Architecture Diagram

Figure 8-37 depicts the data bypass logic.

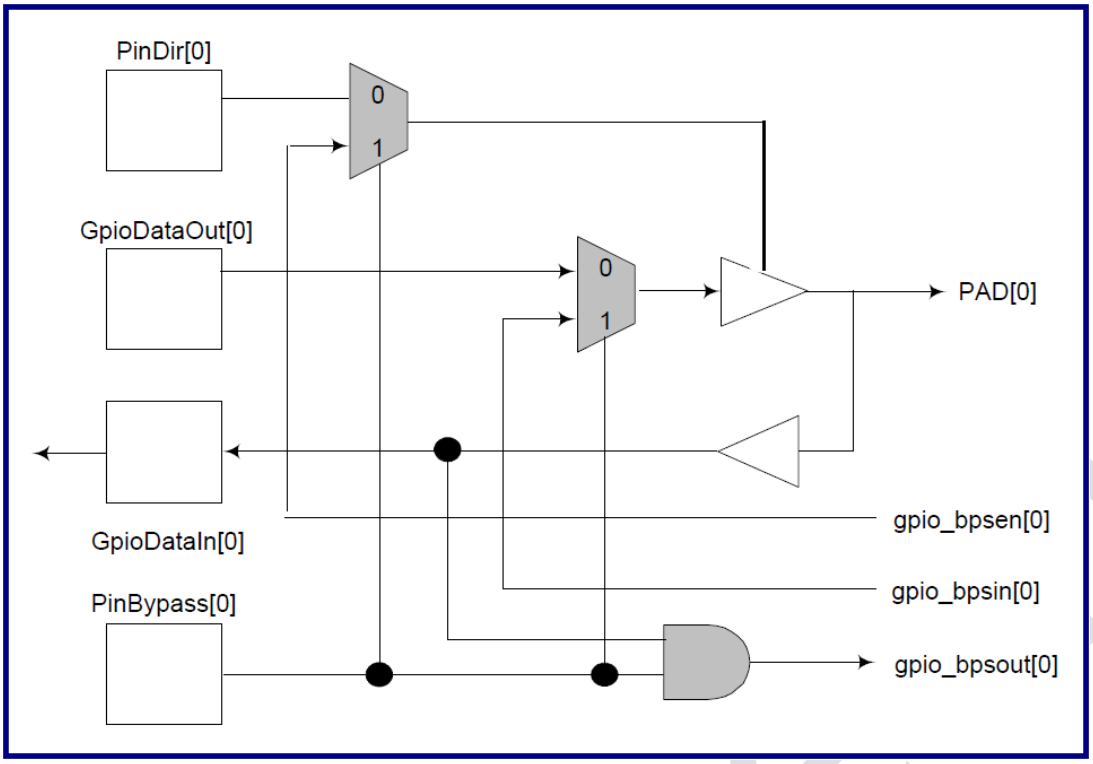


Figure 8-37. Data Bypass Logic

To support the bypass mode, two MUXs and one AND gate need to be added. The PinBypass register controls the bypass mode. If PinBypass[0] is set, it means that pin[0] will be in the bypass mode and all the inputs/outputs of PAD[0] will be controlled by gpio_bpsin[0], gpio_bpsen[0], and gpio_bpsout[0]. Please refer to Figure 8-37 for more detailed information. When the PinBypass[0] register is set, gpio_bpsen[0] will be muxed to the PAD[0] output enable, which means that gpio_bpsen[0] can directly determine the pin direction. Similarly, in the bypass mode, gpio_bpsin[0] is directly muxed to output of PAD[0]. The PAD[0] input is directly connected to gpio_bpsout[0].

8.8.2.2. Connection Diagram

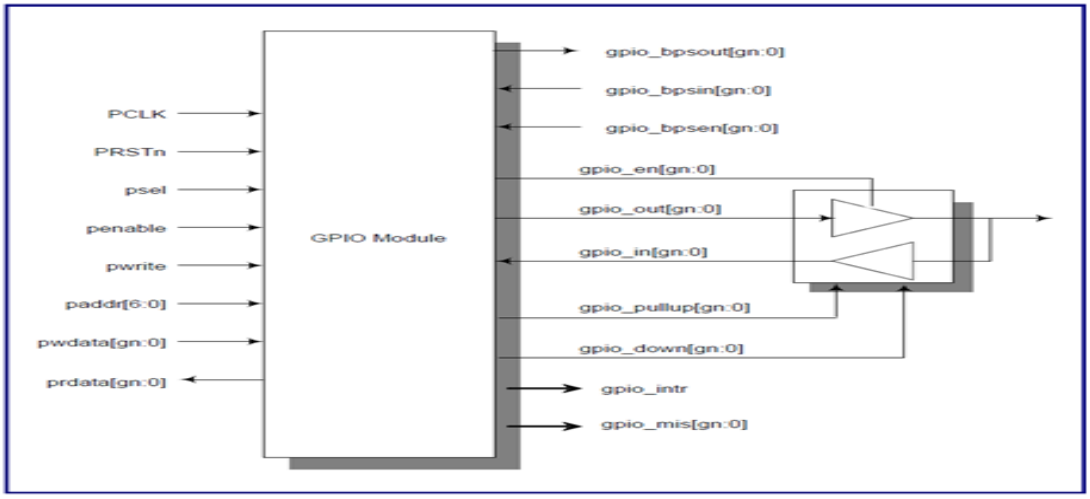


Figure 8-38. Connection Diagram

8.8.3. Programming Guidelines

By default, the GPIO is set to the data input. Programmers can read the GpioDataIn register after the direct reset. When using GPIO to output data, programmers only need to set PinDir to '1'. The GpioDataOut register will then be sent to pad.

If GPIO is used to detect the interrupt, programmers need to set PinDir and the detection type first. PinDir

needs to be set to '0' to indicate that the pin is defined as an input. The detection types include the rising edge, falling edge, both edges, high level, and low level. Please refer to the descriptions of the IntrTrigger, IntrBoth, and IntrRiseNeg register for more detailed information.

The programming steps are as follows:

1. Set PinDir
2. Set IntrTrigger, IntrBoth, and IntrRiseNeg
3. Enable IntrEnable

In some cases, the interrupt detection requires the disbounce function; for examples, using GPIO to control the keypad or switch.

The programming steps for activating the disbounce function are as follows:

1. Set PinDir
2. Set IntrTrigger, IntrBoth, and IntrRiseNeg
3. Set BouncePreScale
4. Enable BounceEnable
5. Enable IntrEnable

8.8.4. Register List

Module Name	Base Address
GPIO_0	0x54000000
GPIO_1	0x56000000

Table 8-231 Register Summary Table

Base Offset (Hex)	Name	Size (Byte)	Type	Description	Default Value (Hex)
00	GpioDataOut	4	RW	GPIO data output register	0000_0000
04	GpioDataIn	4	RO	GPIO data input register	0000_0000
08	PinDir	4	RW	GPIO direction register 0: Input 1: Output	0
0C	PinBypass	4	RW	GPIO bypass register 0: No bypass 1: Bypass	0000_0000
10	GpioDataSet	4	RW1S	GPIO data bit set register When writing to this register, the corresponding bits in the data register will be set to '1' and the other bits will remain unchanged.	0000_0000
14	GpioDataClear	4	RW1C	GPIO data bit clear register When writing to this register, the corresponding bits in the data register will be cleared to '0' and the	0000_0000

				other bits will remain unchanged.	
18	PinPullEnable	4	RW	GPIO pull-up register 0: Pin is not pulled. 1: Pin is pulled.	0000_0000
1C	PinPullType	4	RW	GPIO pull-high/pull-low register 0: Pin is pulled low. 1: Pin is pulled high.	0000_0000
20	IntrEnable	4	RW	GPIO interrupt enable register 0: Pin interrupt is disabled. 1: Pin interrupt is enabled.	0000_0000
24	IntrRawState	4	RO	GPIO interrupt raw status register 0: Interrupt is not detected. 1: Interrupt is detected.	0000_0000
28	IntrMaskedSta te	4	RO	GPIO interrupt masked status register 0: Interrupt is not detected or masked. 1: Interrupt is detected and not masked.	0000_0000
2C	IntrMask	4	RW	GPIO interrupt mask register 0: Mask is disabled. 1: Mask is enabled.	0000_0000
30	IntrClear	4	RW1C	GPIO interrupt clear 0: No effect 1: Clear interrupt	0000_0000
34	IntrTrigger	4	RW	GPIO interrupt trigger method register 0: Edge trigger 1: Level trigger	0000_0000
38	IntrBoth	4	RW	GPIO edge-trigger interrupt by single edge or both edges 0: Single edge 1: Both edges	0000_0000
3C	IntrRiseNeg	4	RW	GPIO interrupt triggered at the rising edge or falling edge 0: Rising edge 1: Falling edge GPIO interrupt triggered by high level or low level 0: High level 1: Low level	0000_0000

Vangotech

40	BounceEnable	4	RW	GPIO pre-scale clock enable When this signal is enabled, PCLK will be divided by the BouncePreScale clocks. This signal is used to extend the clock cycle of detecting interrupt. 0: Disable 1: Enable	0000_0000
44	BouncePreScale	4	RW	GPIO pre-scale This signal is used to adjust different PCLK frequencies. The allowable range is from 0x1 to 0xFFFFFF.	0000_07D0
7C	RevisionNum	4	RO	GPIO revision number	0001_0900

8.8.5. Register Description

The following subsections provide the details of the general-purpose I/O registers.

8.8.5.1. GPIO Data-out Register

The GpioDataOut register is the GPIO data out register. When PinDir indicates the pin is an output, the GpioDataOut register will be connected to io_out. When PinDir indicates the pin is an input, the GpioDataOut register can hold the data.

Table 8-232 GPIO Data-out Register (GpioDataOut, Offset = 0x00)

Bit	Name	Type	Description	Default Value
[31:0]	GpioDataOut	RW	None	0

Note: GpioDataOut[I] is dedicated for io_out[I]. I = '0' to gn.

8.8.5.2. GPIO Data-in Register

The GpioDataIn register is the GPIO data in register. When PinDir indicates the pin is an input, the GpioDataIn register will latch gpio_in at the PCLK rising edge. When PinDir indicates the pin is an output, the GpioDataIn register will be a "don't care" register.

Table 8-233 GPIO Data-in Register (GpioDataIn, Offset = 0x04)

Bit	Name	Type	Description	Default Value
[31:0]	GpioDataIn	RO	None	0

Note: GpioDataIn[I] is dedicated for io_out[I]. I = '0' to gn.

8.8.5.3. Pin Direction Register

The PinDir register controls gpio_en. When PinDir indicates the pin is an output, the related gpio_en will be set to '1'. Otherwise, the related gpio_en will be set to '0'.

Table 8-234 Pin Direction Register (PinDir, Offset = 0x08)

Bit	Name	Type	Description	Default Value
[31:0]	PinBypass	RW	None	0

Note: PinBypass[I] is dedicated for pin[I]. I = '0' to gn.

Vangotech

8.8.5.4. Pin Bypass Mode Register

The PinBypass register controls the bypass mode. When PinBypass indicates the pin is in the bypass mode, gpio_en will be connected to gpio_bps_en, gpio_in will be connected to gpio_bps_out, and gpio_bps_in will be connected to gpio_out.

Table 8-235 Pin Bypass Mode Register (PinBypass, Offset = 0x0C)

Bit	Name	Type	Description	Default Value
[31:0]	PinBypass	RW	None	0

Note: PinBypass[I] is dedicated for pin[I]. I = '0' to gn.

8.8.5.5. GPIO Data Bit Set Register

The GpioDataSet register is the bit operation logic. When writing to this address, if some bits ofGpioDataSet are '1', the related bits of GpioDataOut will be set. For example, if GpioDataOut[7:0] = 0x23 and pwdata[7:0] from APB is 0x47, when writing to the GpioDataSet address, the result of GpioDataOut[7:0] will be 0x67.

Table 8-236 GPIO Data Bit Set Register (GpioDataSet, Offset = 0x10)

Bit	Name	Type	Description	Default Value
[31:0]	GpioDataSet	RW1S	None	0

8.8.5.6. GPIO Data Bit Clear Register

The GpioDataClear register is the bit operation logic. When writing to this address, if some bits of GpioDataClear are '1s', the related bits of GpioDataOut will be cleared. For example, if GpioDataOut[7:0] = 0x23 and pwdata[7:0] from APB is 0x47, when writing to the GpioDataClear address, the result of GpioDataOut[7:0] will be 0x20.

Table 8-237 GPIO Data Bit Clear Register (GpioDataClear, Offset = 0x14)

Bit	Name	Type	Description	Default Value
[31:0]	GpioDataClear	RW1C	None	0

8.8.5.7. Pin Pull Enable Register

The PinPullEnable register controls gpio_pullup and gpio_down. If PinPullEnable indicates that the pin is disabled to "Pull", gpio_pullup and gpio_down will be masked to '0'.

Table 8-238 Pin Pull Enable Register (PinPullEnable, Offset = 0x18)

Bit	Name	Type	Description	Default Value
[31:0]	PinPullEnable	RW	Require the "Pull" option	0

Note: PinPullEnable[I] is dedicated for PAD[I]. I = '0' to gn.

8.8.5.8. Pin Pull Type Register

The PinPullType register controls gpio_pullup and gpio_down. If PinPullEnable is enabled and PinPullType is set to '0', gpio_pullup will be set to '0', and gpio_down will be set to '1'. Please refer to the pull truth table in Table 8-239.

Table 8-239 Pin Pull Type Register (PinPullType, Offset = 0x1C)

Bit	Name	Type	Description	Default Value
[31:0]	PinPullType	RW	Require the "Pull" option	0

Notes:

1. PinPullType[I] is dedicated for PAD[I]. I = '0' to gn.

Vangotech

2. The pull function requires that the I/O pad to be pulled.

Table 8-240 Pull Truth Table

PinDir	PinPullEnable	PinPullType	gpio_pullup	gpio_down
1	X	X	0	0
0	0	X	0	0
0	1	0	0	1
0	1	1	1	0

8.8.5.9. Interrupt Enable Register

The IntrEnable register controls the enable or disable interrupt detection logics. It is a mask of the interrupt detection logic. When the pin direction is the input and the interrupt detection is enabled, the pin can accept the interrupt from pad. The sensed state is stored in the IntrMaskedState register (Masked by IntrEnable). Before turning on IntrEnable, programmers can clear the masked state by writing a '1' to IntrClear to ensure the initial state.

Table 8-241 Interrupt Enable Register (IntrEnable, Offset = 0x20)

Bit	Name	Type	Description	Default Value
[31:0]	IntrEnable	RW	Require the INTR option	0

Note: IntrEnable[I] is dedicated for pin[I]. I = '0' to gn.

8.8.5.10. Interrupt Raw State Register

The IntrRawState register is the raw result of the interrupt detection. If IntrEnable is enabled, the IntrRawState register will reflect the interrupt detection status. Programmers can poll this register to detect an interrupt.

Table 8-242 Interrupt Raw State Register (IntrRawState, Offset = 0x24)

Bit	Name	Type	Description	Default Value
[31:0]	IntrRawState	RO	Require the INTR option	0

Note: IntrRawState[I] is dedicated for pin[I]. I = '0' to gn.

8.8.5.11. Interrupt Masked State Register

The IntrMaskedState register is the masked result of the interrupt detection. The IntrMaskedState register is controlled by the IntrEnable, IntrRawState, and IntrMask registers.

Table 8-243 Interrupt Masked State Register (IntrMaskedState, Offset = 0x28)

Bit	Name	Type	Description	Default Value
[31:0]	IntrMaskedState	RO	Require the INTR option	0

Note: IntrMaskedState[I] is dedicated for pin[I]. I = '0' to gn.

8.8.5.12. Interrupt Mask Register

The IntrMask register is the mask register of the interrupt detection. It masks the IntrRawState register. For example, if IntrEnable[0] = '1', IntrRawState[0] = '1', and IntrMask[0] = '1', IntrMaskedState[0] will never be changed to '1'.

Table 8-244 Interrupt Mask Register (IntrMask, Offset = 0x2C)

Bit	Name	Type	Description	Default Value
[31:0]	IntrMask	RW	Require the INTR option	0

Note: IntrMask[I] is dedicated for pin[I]. I = '0' to 31.

8.8.5.13. Interrupt Clear Register

The IntrClear register is the bit operation logic. If some bits of pwdata are set, when writing to the IntrClear address, the related bits of IntrMaskedState and IntrRawState will be cleared.

Table 8-245 Interrupt Clear Register (IntrClear, Offset = 0x30)

Bit	Name	Type	Description	Default Value
[31:0]	IntrClear	RW1C	Require the INTR option	0

8.8.5.14. IntrTrigger Register

The IntrTrigger register indicates the interrupt trigger method of each pin. If IntrTrigger is '0', the interrupt will be triggered at the edge; otherwise, the interrupt will be triggered at the level.

Table 8-246 IntrTrigger Register (IntrTrigger, Offset = 0x34)

Bit	Name	Type	Description	Default Value
[31:0]	IntrTrigger	RW	Require the INTR option	0

Note: IntrTrigger[I] is dedicated for pin[I]. I = '0' to 31.

8.8.5.15. Interrupt Both Edge Trigger Register

The IntrBoth register indicates that the edge trigger can be single edge or both edges. If IntrTrigger indicates the edge trigger and IntrBoth is '0', the interrupt edge trigger will be done by the single edge. If IntrTrigger indicates the edge trigger and IntrBoth is '1', the interrupt edge trigger will be done by both edges.

Table 8-247 Interrupt Both Edge Trigger Register (IntrBoth, Offset = 0x38)

Bit	Name	Type	Description	Default Value
[31:0]	IntrBoth	RW	Require the INTR option	0

Note: IntrBoth[I] is dedicated for pin[I]. I = '0' to 31.

8.8.5.16. Interrupt Rise or Neg Edge Trigger Register

The IntrRiseNeg register indicates whether the edge trigger is at the rising edge or falling edge. If the IntrTrigger register is the edge trigger, IntrBoth will be the single edge, IntrRiseNeg will be '0', and the interrupt edge trigger will be done at the rising edge. If IntrTrigger is the edge trigger, IntrBoth will be the single edge, IntrRiseNeg will be '1', and the interrupt edge trigger will be done at the falling edge.

Table 8-248 Interrupt Rise or Neg Edge Trigger Register (IntrRiseNeg, Offset = 0x3C)

Bit	Name	Type	Description	Default Value
[31:0]	IntrRiseNeg	RW	Require the INTR option	0

Note: IntrRiseNeg[I] is dedicated for pin[I]. I = '0' to 31.

8.8.5.17. Bounce Enable Register

The BounceEnable register controls the bounce function. If BounceEnable is on, the interrupt detection will be sampled by the extended clock. The extension number will be controlled by the BouncePreScale

register.

Table 8-249 Bounce Enable Register (BounceEnable, Offset = 0x40)

Bit	Name	Type	Description	Default Value
[31:0]	BounceEnable	RW	Require the INTR option	0

Note: BounceEnable[I] is dedicated for pin[I]. I = '0' to 31.

8.8.5.18. Bounce Clock Pre-scale Register

The BouncePreScale register is an automatic register to indicate for the bounce timer. It can extend PCLK to the BouncePreScale cycles, which can be used to adjust the interrupt sample clock period in different machines. The reset value is 0x7D0, which means that if the APB clock frequency is 66 MHz, the disbounce clock will be divided by (0x7D0 + 1) to 32.98 KHz. Programmers can adjust this register to fit different systems.

Table 8-250 Bounce Clock Pre-scale Register (BouncePreScale, Offset = 0x44)

Bit	Name	Type	Description	Default Value
[23:0]	BouncePreScale	RW	Require the INTR option	0x7D0

Vangotech

8.9. CAN

8.9.1. Overview

The CAN controller is an I2C/APB compliant controller to act as a transmitter or receiver in the CAN bus. Data are transmitted to and received from the CAN bus via a buffered interface. There are three transmit buffers and two receive buffers.

8.9.2. Features

- Compliant with BOSCH CAN 2.0A/B and CAN FD 1.0 protocol
- Arbitration bit rate up to 1 Mbps
- Data bit rate up to 8 Mbps
- Supports Sleep mode, Silence Mode, and Loopback Mode
- Three transmit buffers with prioritization
- Two receive buffers, receive buffer0 and receive buffer1
- Four ID filters for receive buffer0 and one data filter for receive buffer0's first two bytes
- Two ID filters for receive buffer1 and one data filter for receive buffer1's first two bytes
- One interrupt output pin

8.9.3. Block Diagram

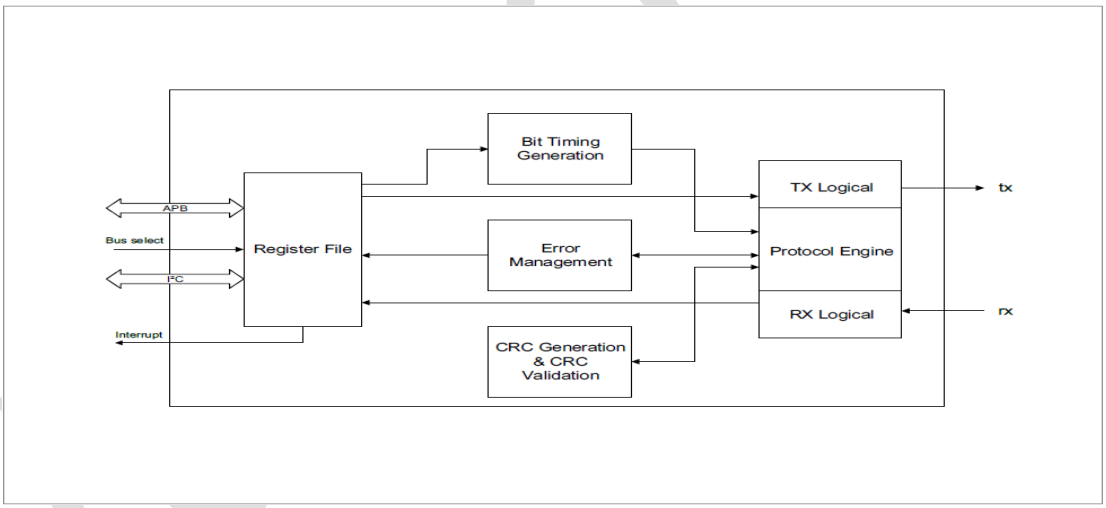


Figure 8-39. Functional Block Diagram

8.9.4. Operations and Functional Descriptions

8.9.4.1. Operation Mode

FTCAN010 has five operation modes:

- Configuration mode
- Normal mode
- Sleep mode
- Listen mode
- Loopback mode

Those modes are chosen by the OMR bits in CE1 register. When modifying the OMR bits to the sleep mode, the sleep mode is entered after current ongoing transmission is completed. Other modes are entered

immediately once OMR bits are set. Configuration mode is the default operation mode after reset.

Configuration Mode

Configuration mode is used to configure registers before transmit or receive frame. Configuration mode is selected automatically after power-up.When configuration mode is entered, error counters will be cleared. It is forbidden to transmit and receive in configuration mode.

Normal Mode

Normal mode is for transmitting and receiving frames.

Sleep Mode

Sleep mode is used to minimize the power consumption of the controller. Setting 3'b010 to OMR bits can enter into this mode, as long as there is no activity on the bus. If any activities are detected on the bus by controller, the sleep mode will be automatically changed to normal mode; the OMR bits will be changed as well. Change the OMR bits by software can exit from this mode, but OMR bits can only be transmitted or received after it has found a valid end of frame sequence (7 recessive bits).

Listen Mode

Listen mode is a silent mode which means that no messages will be transmitted, including error flags or acknowledge signals. In listen mode, the error counters are reset to 0 and deactivated.

Loopback Mode

Loopback mode allows internal transmission of messages from the transmit buffers to the receive buffers without transmitting the frame on the CAN bus, just as if the frame is coming from another node.

8.9.4.2. Transmit and Receive Buffer

Transmit buffer

FTCAN010 has three transmit buffers with firm priority. Transmit buffer0 has the highest priority, transmit buffer1 has the middle priority, and transmit buffer2 has the lowest priority.

Figure 8-40 is structure of the transmit buffer:

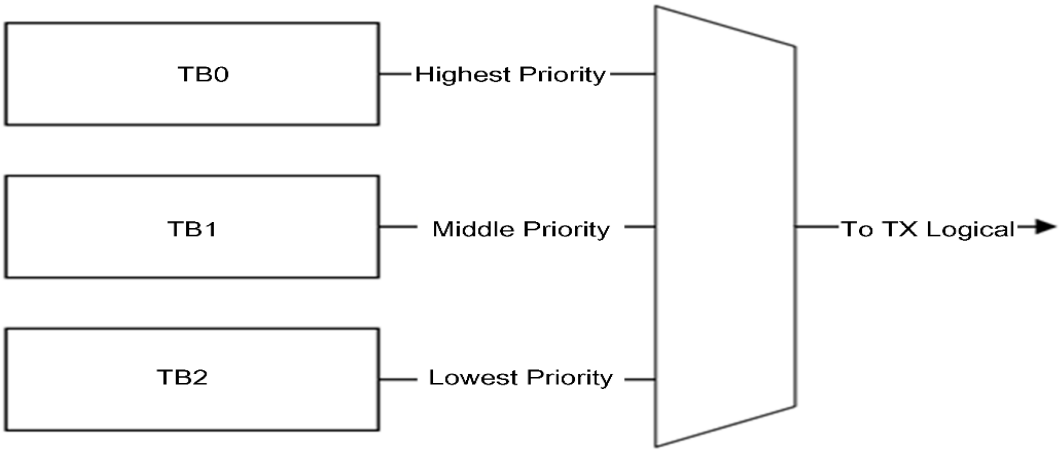


Figure 8-40. Transmit Buffer Structure

Receive buffer

FTCAN010 has two receive buffers, the receive buffer0 and the receive buffer1. Received frame should pass ID filter and data filter and allow frame type, then receive buffer can store that frame. When an input frame pass frame type filter and receive buffer0's filter, the frame will be stored into RX buffer0 regardless of the filter of receive buffer1 has been passed or not. When an input frame pass frame type filter and

receive buffer1's filter, and did not pass receive buffer0's filter, the frame would be stored into RX buffer1. Figure 8-41 shows receive buffer0's filter and registers:

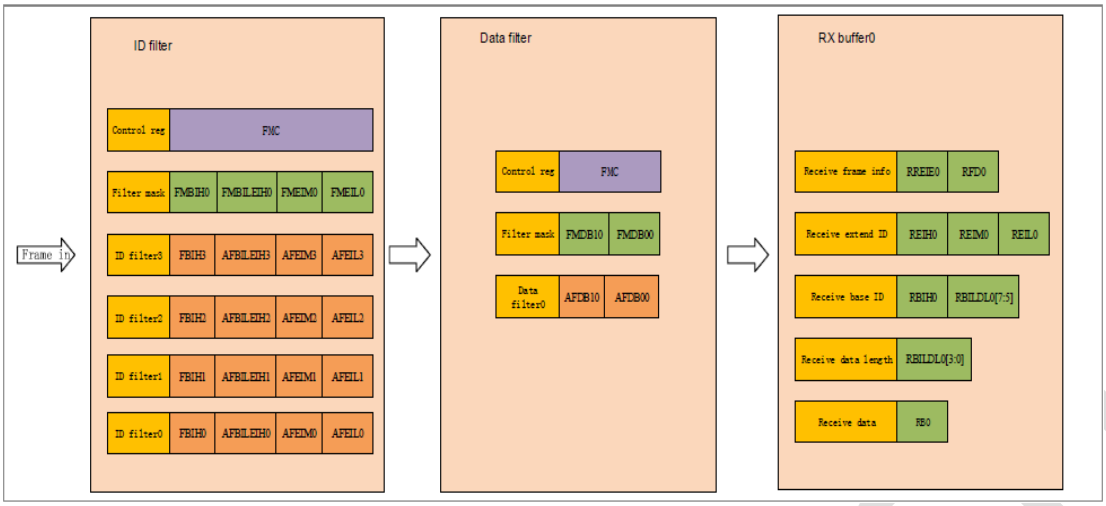


Figure 8-41. Receive Buffer0's Filter and Registers

Figure 8-42 shows receive buffer1's filter and registers:

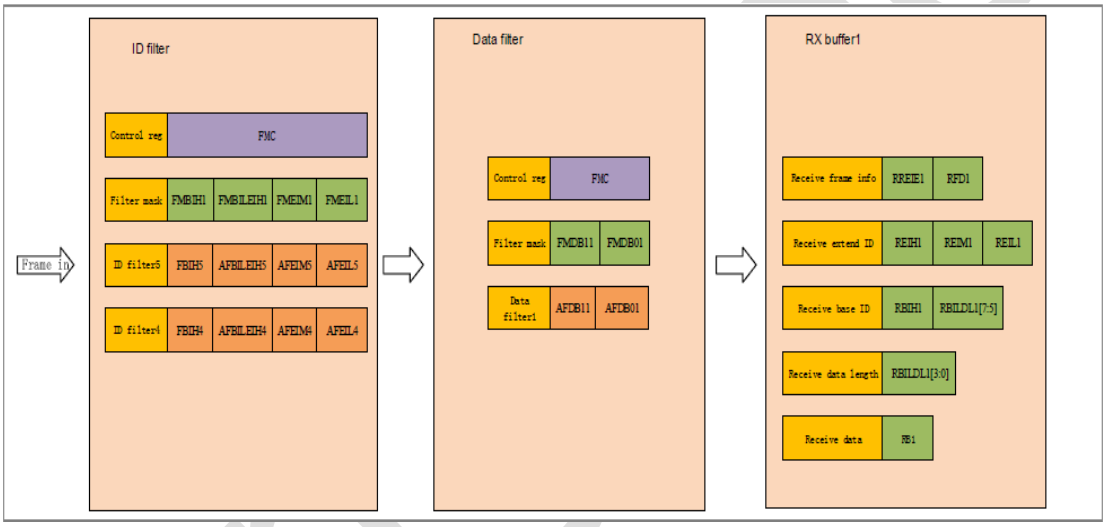


Figure 8-42. Receive Buffer1's Filter and Registers

8.9.4.3. Clock and Reset

Clock

FTCAN010 has two clocks, sys_clk and pclk. sys_clk is for CAN controller and pclk is for AMBA APB interface. Bit rate calculation is based on sys_clk.

In order to decrease power consumption in sleep mode, FTCAN010 has one built-in clock gate. This clock gate must be replaced with a clock gate from the library to avoid glitches on the gated clock output. When CAN bus is free and CE1[2:0] has been set to 3'b010 (sleep mode), clock gating will be activated. A dominant bit observed on CAN bus will let FTCAN010 exit from sleep mode, and clock gating will be deactivated.

Reset

FTCAN010 has two input resets, sys_rst_n and presetn. sys_rst_n is for CAN controller and presetn is for AMBA APB interface. sys_rst_n will reset all registers to default values. FTCAN010 also has one software reset, but software reset can only reset part of registers in FTCAN010, please refer to CE1.RR bit for more details.

Reset Sequence

For FTCAN01 reset sequence, deassert sys_rst_n before presetn does.

8.9.5. Programming Guidelines

Transmit programming flow chart

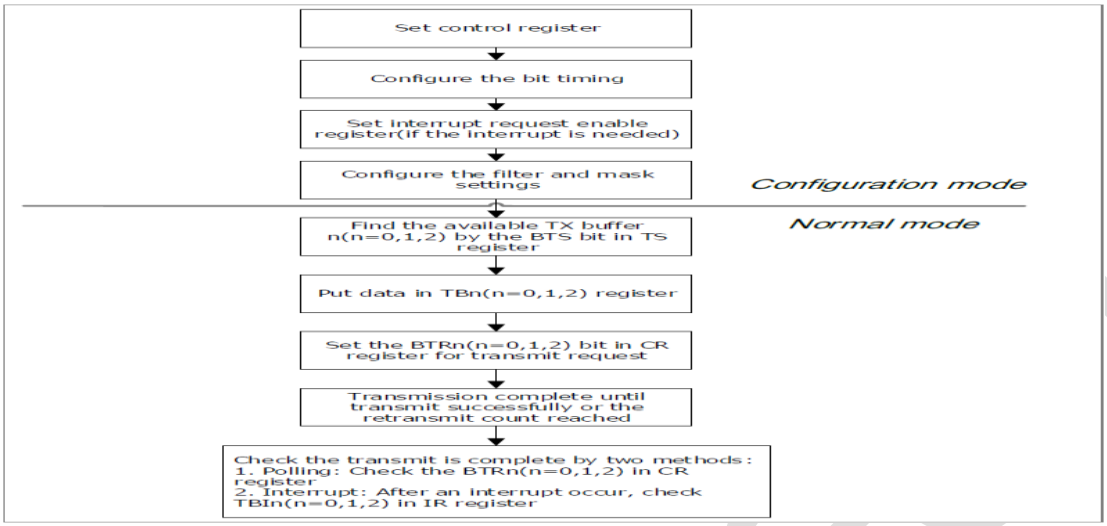


Figure 8-43. Transmit Programming Flow Chart

Receive programming flow chart

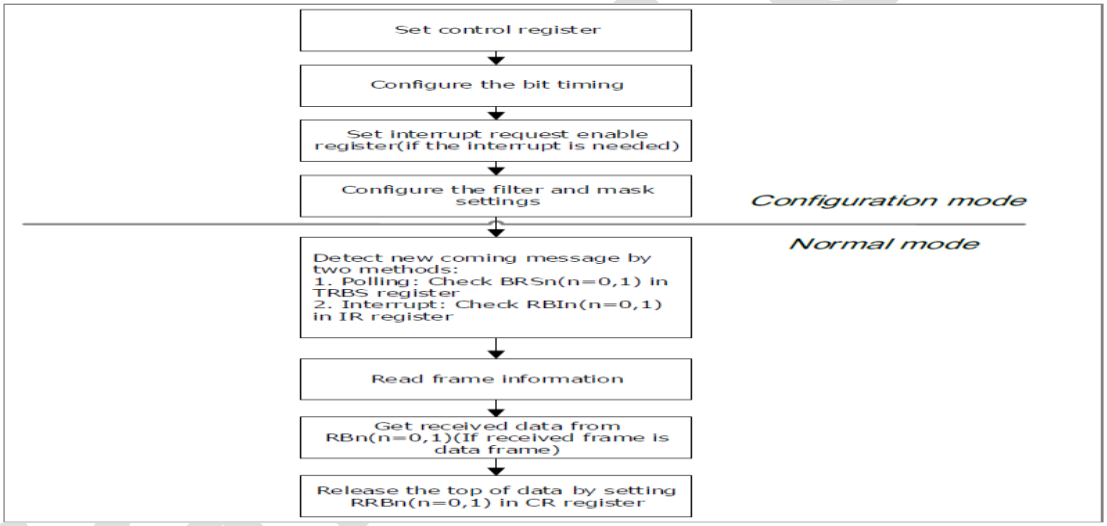


Figure 8-44. Receive Programming Flow Chart

8.9.6. Register List

Module Name	Base Address
CAN_0	0x56300000
CAN_1	0x54900000

Table 8-251 and describes the CAN controller registers. The detailed information of the CAN controller registers will be described in the following subsections. Register access in FTCAN010 is 4-byte based, address of each access should be aligned to 32-bit boundary (0x0, 0x4, 0x8, 0xC, …).

Table 8-251. Register Summary Table

Base Offset (Hex)	Name	Size (Byte)	Type	Description	Default Value (Hex)

00	CR	1	RW	Command request register	00
01	IRE	1	RW	Interrupt enable register	00
02	CE0	1	RW	Control enable 0 register	00
03	CE1	1	RW	Control enable 1 register	00
04	BS	1	RO	Bus status register	00
05	TRBS	1	RO	Transmit and receive buffer status register	00
06	TS	1	RO	Transmit Status	00
07	Rsvd	-	-	Reserved	-
08	TFD0	1	RW	Transmit flexible data rate register 0	00
09	TREIE0	1	RW	Transmit remote and extended identifier enable register 0	00
0A	TEIL0	1	RW	Transmit extended identifier low register 0	00
0B	TEIM0	1	RW	Transmit extended identifier middle register 0	00
0C	TEIH0	1	RW	Transmit extended identifier high register 0	00
0D	TBILDL0	1	RW	Transmit base identifier low and data length register 0	00
0E	TBIH0	1	RW	Transmit base identifier high register 0	00
0F	Rsvd	-	-	Reserved	-
10	TFD1	1	RW	Transmit flexible data rate register 1	00
11	TREIE1	1	RW	Transmit remote and extended identifier enable register 1	00
12	TEIL1	1	RW	Transmit extended identifier low register 1	00
13	TEIM1	1	RW	Transmit extended identifier middle register 1	00
14	TEIH1	1	RW	Transmit extended identifier high register 1	00
15	TBILDL1	1	RW	Transmit base identifier low and data length register 1	00
16	TBIH1	1	RW	Transmit base identifier high register 1	00
17	Rsvd	-	-	Reserved	-
18	TFD2	1	RW	Transmit flexible data rate register 2	00

Vangotech

19	TREIE2	1	RW	Transmit remote and extended identifier enable register 2	00
1A	TEIL2	1	RW	Transmit extended identifier low register 2	00
1B	TEIM2	1	RW	Transmit extended identifier middle register 2	00
1C	TEIH2	1	RW	Transmit extended identifier high register 2	00
1D	TBILDL2	1	RW	Transmit base identifier low and data length register 2	00
1E	TBIH2	1	RW	Transmit base identifier high register 2	00
1F	Rsvd	-	-	Reserved	-
20	AFEIL0	1	RW	Acceptance filter extended identifier low register 0	00
21	AFEIM0	1	RW	Acceptance filter extended middle register 0	00
22	AFBILEIH0	1	RW	Acceptance filter base identifier low and extended identifier high 0	00
23	AFBIH0	1	RW	Acceptance filter base identifier high register 0	00
24	AFEIL1	1	RW	Acceptance filter extended identifier low register 1	00
25	AFEIM1	1	RW	Acceptance filter extended middle register 1	00
26	AFBILEIH1	1	RW	Acceptance filter base identifier low and extended identifier high 1	00
27	AFBIH1	1	RW	Acceptance filter base identifier high register 1	00
28	AFEIL2	1	RW	Acceptance filter extended identifier low register 2	00
29	AFEIM2	1	RW	Acceptance filter extended middle register 2	00
2A	AFBILEIH2	1	RW	Acceptance filter base identifier low and extended identifier high 2	00
2B	AFBIH2	1	RW	Acceptance filter base identifier high register 2	00
2C	AFEIL3	1	RW	Acceptance filter extended identifier low register 3	00
2D	AFEIM3	1	RW	Acceptance filter extended middle register 3	00

Vangotech

2E	AFBILEIH3	1	RW	Acceptance filter base identifier low and extended identifier high 3	00
2F	AFBIH3	1	RW	Acceptance filter base identifier high register 3	00
30	AFEIL4	1	RW	Acceptance filter extended identifier low register 4	00
31	AFEIM4	1	RW	Acceptance filter extended middle register 4	00
32	AFBILEIH4	1	RW	Acceptance filter base identifier low and extended identifier high 4	00
33	AFBIH4	1	RW	Acceptance filter base identifier high register 4	00
34	AFEIL5	1	RW	Acceptance filter extended identifier low register 5	00
35	AFEIM5	1	RW	Acceptance filter extended middle register 5	00
36	AFBILEIH5	1	RW	Acceptance filter base identifier low and extended identifier high 5	00
37	AFBIH5	1	RW	Acceptance filter base identifier high register 5	00
38	AFDB00	1	RW	Acceptance filter data byte0 register 0	00
39	AFDB10	1	RW	Acceptance filter data byte1 register 0	00
3A	AFDB01	1	RW	Acceptance filter data byte0 register 1	00
3B	AFDB11	1	RW	Acceptance filter data byte1 register 1	00
3C	AFC0	1	RW	Acceptance filter control register 0	00
3D	AFC1	1	RW	Acceptance filter control register 1	00
3E	Rsvd	-	-	Reserved	-
3F	Rsvd	-	-	Reserved	-
40	FMEIL0	1	RW	Filter mask extended identifier low register 0	00
41	FMEIM0	1	RW	Filter mask extended middle register 0	00
42	FMBILEIH0	1	RW	Filter mask base identifier low and extended identifier high 0	00
43	FMBIH0	1	RW	Filter mask base identifier high register 0	00
44	FMEIL1	1	RW	Filter mask extended identifier low	00

Vangotech

				register 1	
45	FMEIM1	1	RW	Filter mask extended middle register 1	00
46	FMBILEIH1	1	RW	Filter mask base identifier low and extended identifier high 1	00
47	FMBIH1	1	RW	Filter mask base identifier high register 1	00
48	FMDB00	1	RW	Filter mask data byte0 register 0	00
49	FMDB10	1	RW	Filter mask data byte1 register 0	00
4A	FMDB01	1	RW	Filter mask data byte0 register 1	00
4B	FMDB11	1	RW	Filter mask data byte1 register 1	00
4C	FMC	1	RW	Filter mask control register	00
4D	Rsvd	-	-	Reserved	-
4E	Rsvd	-	-	Reserved	-
4F	Rsvd	-	-	Reserved	-
50	RFD0	1	RO	Received flexible data rate register 0	00
51	RFEIE0	1	RO	Received remote and extended identifier enable register 0	00
52	REIL0	1	RO	Received extended identifier low register 0	00
53	REIM0	1	RO	Received extended identifier middle register 0	00
54	REIH0	1	RO	Received extended identifier high register 0	00
55	RBILDL0	1	RO	Received base identifier low and data length register 0	00
56	RBIH0	1	RO	Received base identifier high register 0	00
57	Rsvd	-	-	Reserved	-
58	RFD1	1	RO	Received flexible data rate register 1	00
59	RFEIE1	1	RO	Received remote and extended identifier enable register 1	00
5A	REIL1	1	RO	Received extended identifier low register 1	00
5B	REIM1	1	RO	Received extended identifier middle register 1	00
5C	REIH1	1	RO	Received extended identifier high register 1	00
5D	RBILDL1	1	RO	Received base identifier low and	00

Vangotech

				data length register 1	
5E	RBIH1	1	RO	Received base identifier high register 1	00
5F	Rsvd	-	-	Reserved	-
60	NBTC0	1	RW	Nominal bit timing configuration 0 register	00
61	NBTC1	1	RW	Nominal bit timing configuration 1 register	00
62	NBTC2	1	RW	Nominal bit timing configuration 2 register	00
63	NBTC3	1	RW	Nominal bit timing configuration 3 register	00
64	NBTC4	1	-	Nominal bit timing configuration 4 register	00
65	Rsvd	-	-	Reserved	-
66	Rsvd	-	-	Reserved	-
67	Rsvd	-	-	Reserved	-
68	DBTC0	1	RW	Data bit timing configuration 0 register	00
69	DBTC1	1	RW	Data bit timing configuration 1 register	00
6A	DBTC2	1	RW	Data bit timing configuration 2 register	00
6B	DBTC3	1	RW	Data bit timing configuration 3 register	00
6C	RECL	1	RO	Receiver error counter low register	00
6D	RECH	1	RO	Receiver error counter high register	00
6E	TECL	1	RO	Transmitter error counter low register	00
6F	TECH	1	RO	Transmitter error counter high register	00
70	ET	1	RO	Error type register	00
71	TED	1	RO	Transmit and receive error detect register	00
72	Rsvd	-	-	Reserved	-
73	Rsvd	-	-	Reserved	-
74	RBTSL0	1	RO	Receive buffer time stamp low 0 register	00
75	RBTSH0	1	RO	Receive buffer time stamp high 0 register	00
76	RBTSL1	1	RO	Receive buffer time stamp low 1	00

Vangotech

				register	
77	RBTSH1	1	RO	Receive buffer time stamp high 1 register	00
78	IR	1	RW1C	Interrupt request register	00
79	Rsvd	-	-	Reserved	-
7A	Rsvd	-	-	Reserved	-
7B	Rsvd	-	-	Reserved	-
7C	TB0D	64	RW	Transmit buffer0 data register	512'h0
BC	TB1D	64	RW	Transmit buffer1 data register	512'h0
FC	TB2D	64	RW	Transmit buffer2 data register	512'h0
13C	RB0D	64	RO	Receive buffer0 data register	512'h0
17C	RB1D	64	RO	Receive buffer1 data register	512'h0

8.9.7. Register Definition

8.9.7.1. Command Request Register

Table 8-252. Command Request Register (CR, Offset: 0x0000)

Bit	Name	Type	Description	Default Value
7	RRB0	RW	Release receive buffer0 Setting this bit will pop one frame data from receive buffer0 when receive buffer0 is occupied, it is no use to set this bit when receive buffer0 is empty. Refer to TBRS[4:3] for status of receive buffer0. This bit will be self-cleared at next sys_clk cycle when this bit is set to 1'b1.	0
6	RRB1	RW	Release receive buffer1 Setting this bit will pop one frame data from receive buffer1 when receive buffer1 is occupied, it is no use to set this bit when receive buffer1 is empty. Refer to TBRS[2:1] for status of receive buffer1. This bit will be self-cleared at next sys_clk cycle when this bit is set to 1'b1.	0
5	BTR0	RW	Buffer0 transmission request 1'b1: A data frame or a remote frame transmit request. This bit will be cleared after buffer0's transmission has been done or retransmission count has been reached. 1'b0: No action. This will not cancel a previously requested transmission. Buffer0 transmission request has the highest priority to transmit.	0
4	BTR1	RW	Buffer1 transmission request 1'b1: A data frame or a remote frame transmit request. This bit will be cleared after buffer1's transmission has been done or retransmission count has been reached. 1'b0: No action. This will not cancel a previously requested	0

			transmission. Buffer1 transmission request has the middle priority to transmit.	
3	BTR2	RW	Buffer2 transmission request 1'b1: A data frame or a remote frame transmit request. This bit will be cleared after buffer2's transmission has been done or retransmission count has been reached. 1'b0: No action. This will not cancel a previously requested transmission. Buffer2 transmission request has the lowest priority to transmit.	0
2	RRBA0	RW	Release receive buffer0 all Setting this bit will clear all of the receive buffer0(3 depth FIFO) and allow the buffer to be used by a further received message. Overrun flag will be cleared by this action. This bit is self-cleared.	0
1	RRBA1	RW	Release receive buffer1 all Setting this bit will clear all of the receive buffer1(3 depth FIFO) and allow the buffer to be used by a further received message. Overrun flag will be cleared by this action. This bit is self-cleared.	0
0	-	Rsvd	Reserved	0

8.9.7.2. Interrupt Request Enable Register

Table 8-253. Interrupt Request Enable Register (IRE, Offset: 0x0001)

Bit	Name	Type	Description	Default Value
7	WIE	RW	Enable wake-up interrupt 1'b1: Enable wake up interrupt. Whenever the bus activity occurs, an interrupt will be asserted if current operation mode is in sleep mode. 1'b0: Disable wake up interrupt	0
6	OIE	RW	Enable overrun interrupt 1'b1: Enable overrun interrupt. Whenever the receiver buffer overrun occurs, an interrupt will be asserted (BS[3:2] is not 2'b00). 1'b0: Disable overrun interrupt	0
5	EIE	RW	Enable error interrupt 1'b1: Enable error interrupt. Whenever an error is detected, an interrupt will be asserted. 1'b0: Disable error interrupt	0
4	TIE	RW	Enable transmit interrupt 1'b1: Enable transmit interrupt. Whenever a message has been successfully transmitted, an interrupt will be asserted. 1'b0: Disable transmit interrupt	0
3	RIE	RW	Enable receive interrupt 1'b1: Enable receive interrupt. Whenever a message has been	0

Vangotech

			received correctly, an interrupt will be asserted. 1'b0: Disable receive interrupt	
[2:0]	-	Rsvd	Reserved	0

8.9.7.3. Control Enable 0 Register

Table 8-254. Control Enable 0 Register (CE0, Offset: 0x0002)

Bit	Name	Type	Description	Default Value
7	TSE	RW	Enable time stamp 1'b1: Enable time stamp. For transmitting logic, it will replace the last two bytes with the time stamp value when transmit data length is more than or equal to 2 bytes. For receiving logic, it will store the local time stamp value into receive buffer time stamp register when a new message is received correctly. 1'b0: Disable time stamp	0
6	EnDB	RW	Enable debounce logic for RX 1'b1: Enable debounce logic 1'b0: Disable debounce logic	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.4. Control Enable 1 Register

Table 8-255. Control Enable 1 Register (CE1, Offset: 0x0003)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	Reserved	0
6	RR	RW	Reset request Setting this bit will perform a part of reset in CAN controller. The following registers will be reset to 0: 1. RBIH0, RBIL0, RBIH1, RBIL1 (Receive ID register) 2. REIH0, REIM0, REIL0, REIH1, REIM1, REIL1 (Receive extended identifier register) 3. RREIE0, RREIE1 (Receive remote and extended identifier enable register) 4. RFD0, RFD1 (Receive flexible data-rate register) 5. TECH, TECL, RECH, RECL (Transmit error counter and receive error counter) 6. ET, TED (Error type register, transmit and receive error detected register) 7. RBTSL0, RBTSH0, RBTSL1, RBTSH1 (Receive buffer time stamp) 8. RB0D, RB1D (Receive buffer 0 data, receive buffer 1 data) There will be no change for other register's value. This bit is	0

Vangotech

			self-cleared after reset has been done.	
5	TOF	R/W	Transmit 1 OVERLOAD fame for RX FIFO full condition 1'b1: Enable transmit one OVERLOAD frame when RX FIFO full 1'b0: Disable transmit one OVERLOAD frame when RX FIFO full	0
[4:3]	RT	RW	Retransmission Max. retransmission times control due to arbitration loss and errors are shown as below: 2'b00: Message will always be retransmitted if the previous transmission is failed. 2'b01: Message will not be retransmitted if one message transmission is failed. 2'b10: Message will be retransmitted at most 3 times if one message transmission is failed. 2'b11: Message will be retransmitted at most 8 times if one message transmission is failed.	0
[2:0]	OMR	RW	Operation mode request 3'b000: Configuration mode 3'b001: Normal mode 3'b010: Sleep mode 3'b011: Listen mode 3'b100: Loopback mode	0

8.9.7.5. Bus Status Register

Table 8-256. Bus Status Register (BS, Offset: 0x0004)

Bit	Name	Type	Description	Default Value
7	BO	RO	Bus-Off 1'b1: Indicate bus-off status. This bit is set when the transmit error counter reaches to 256. At this point, CAN controller will wait for 128 occurrences of 11 consecutive recessive bits to leave bus-off status. 1'b0: Indicates not at the bus-off status	0
6	EW	RO	Error warning 1'b1: Transmit error counter or receive error counter has reached (>=) to the warning limit of 96 1'b0: No error warning	0
5	TS	RO	Transmit status 1'b1: Indicate that a message is transmitting. 1'b0: No message is transmitting.	0
4	RS	RO	Receive status 1'b1: Indicate that a message is receiving. 1'b0: No message is transmitting.	0

Vangotech

[3:2]	DO	RO	Data overrun 2'b00: No buffer occurs overrun 2'b01: Receive buffer0 occurs overrun 2'b10: Receive buffer1 occurs overrun 2'b11: Both buffer0 and buffer1 occur overrun	0
1	-	Rsvd	Reserved	0
0	-	Rsvd	Reserved	0

8.9.7.6. Transmit and Receive Buffer Status Register

Table 8-257. Transmit and Receive Buffer Status Register (TRBS, Offset: 0x0005)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	Reserved	0
6	-	Rsvd	Reserved	0
5	-	Rsvd	Reserved	0
[4:3]	BRS0	RO	Buffer0 receive status. Receive buffer0 is a 3-depth of FIFO. 2'b00: Receive buffer0 is empty. 2'b01: Receive buffer0 has one message. 2'b10: Receive buffer0 has two messages. 2'b11: Receiver buffer0 is full.	0
[2:1]	BRS1	RO	Buffer1 receive status. Receive buffer1 is a 3-depth of FIFO. 2'b00: Receive buffer1 is empty. 2'b01: Receive buffer1 has one message. 2'b10: Receive buffer1 has two messages. 2'b11: Receiver buffer1 is full.	0
0	-	Rsvd	Reserved	0

8.9.7.7. Transmit Status Register

Table 8-258. Transmit Status Register (TS, Offset: 0x0006)

Bit	Name	Type	Description	Default Value
7	BTA0	RO	Buffer0 transmit abort 1'b1: Retransmit times exceed RT's configured times. It will be cleared at the next transmit request. 1'b0: Transmit is not aborted.	0
6	BTA1	RO	Buffer1 transmit abort 1'b1: Retransmit times exceed RT's configured times. It will be cleared at the next transmit request.	0

Vangotech

			1'b0: Transmit is not aborted.	
5	BTA2	RO	Buffer2 transmit abort 1'b1: Retransmit times exceed RT's configured times. It will be cleared at the next transmit request. 1'b0: Transmit is not aborted.	0
4	BTLA0	RO	Buffer0 transmit lost arbitration 1'b1: Buffer0 transmit lost arbitration occurs. It will be cleared at next transmit request. 1'b0: Buffer0 transmit lost arbitration is not occurred.	0
3	BTLA1	RO	Buffer1 transmit lost arbitration 1'b1: Buffer1 transmit lost arbitration occurs. It will be cleared at the next transmit request. 1'b0: Buffer1 transmit lost arbitration is not occurred.	0
2	BTLA2	RO	Buffer2 transmit lost arbitration 1'b1: Buffer2 transmit lost arbitration occurs. It will be cleared at the next transmit request. 1'b0: Buffer2 transmit lost arbitration is not occurred.	0
[1:0]	BTS	RO	Buffer transmit status 2'b00: No buffer is transmitting. 2'b01: Buffter0 is transmitting. 2'b10: Buffter1 is transmitting. 2'b11: Buffter2 is transmitting.	0

8.9.7.8. Reserved

Table 8-259. Reserved (Offset: 0x0007)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.9. Transmit Flexible Data-Rate Register 0

Table 8-260. Transmit Flexible Data-rate Register 0 (TFD0, Offset: 0x0008)

Bit	Name	Type	Description	Default Value
7	TEDL	RW	Extended data length bit for TX buffer0 It only exists in CAN FD format frames. It distinguishes between CAN format and CAN FD format frames. In a CAN format frame, the dominant bit is transmitted instead of EDL. 1'b1: FD frame 1'b0: Non-FD frame	0
6	TBRS	RW	Bit rate switch bit for TX buffer0 It decides whether the bit rate is switched from the standard bit rate of the arbitration phase to the preconfigured alternate bit rate of the data	0

			phase inside a CAN FD format frame. 1'b1: Bit rate switch 1'b0: Bit rate doesn't switch.	
[5:0]	-	Rsvd	Reserved	0

8.9.7.10. Transmit Remote and Extended Identifier Enable Register 0

Table 8-261. Transmit Remote and Extended Identifier Enable Register 0 (TREIE0, Offset: 0x0009)

Bit	Name	Type	Description	Default Value
7	TRTR	RW	Transmit remote transmission request bit for TX buffer0 1'b1: Remote transmit frame request 1'b0: Data frame request Please note that, in CAN FD format frames, it must be set to 0, since there is no remote frame in CAN FD format. Otherwise, a form error would be detected.	0
6	TEIR	RW	Transmit extended identifier request bit for TX buffer0 1'b1: Use extended format for transmit request 1'b0: Use standard format for transmit request	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.11. Transmit Extended Identifier Low Register 0

Table 8-262. Transmit Extended Identifier Low Register 0 (TEIL0, Offset: 0x000A)

Bit	Name	Type	Description	Default Value
[7:6]	TEIL	RW	The last two bits of 18-bit extended transmit identifier for TX buffer0	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.12. Transmit Extended Identifier Middle Register 0

Table 8-263. Transmit Extended Identifier Middle Register 0 (TEIM0, Offset: 0x000B)

Bit	Name	Type	Description	Default Value
[7:0]	TEIM	RW	The middle 8 bits of 18-bit extended transmit identifier for TX buffer0	0

8.9.7.13. Transmit Extended Identifier High Register 0

Table 8-264. Transmit Extended Identifier High Register 0 (TEIH0, Offset: 0x000C)

Bit	Name	Type	Description	Default Value
[7:0]	TEIH	RW	The most high 8 bits of 18-bit extended transmit identifier for TX buffer0	0

8.9.7.14. Transmit Base Identifier Low and Data Length Register 0

Table 8-265. Transmit Base Identifier Low and Data Length Register 0 (TBILDL0, Offset: 0x000D)

Bit	Name	Type	Description	Default Value
[7:5]	TBIL	RW	The low 3 bits of 11-bit base transmit identifier for TX buffer0	0
[4:4]	-	Rsvd	Reserved	0
[3:0]	TDL	RW	The transmit data payload length for TX buffer0 is as follows: 4'h0: 0 byte 4'h1: 1 byte 4'h2: 2 bytes 4'h3: 3 bytes 4'h4: 4 bytes 4'h5: 5 bytes 4'h6: 6 bytes 4'h7: 7 bytes 4'h8: 8 bytes 4'h9 ~ 4'hf: 8 bytes (for CAN format) The following values are for CAN FD frame: 4'h8: 8 bytes 4'h9: 12 bytes 4'hA: 16 bytes 4'hB: 20 bytes 4'hC: 24 bytes 4'hD: 32 bytes 4'hE: 48 bytes 4'hF: 64 bytes	0

8.9.7.15. Transmit Base Identifier High Register 0

Table 8-266. Transmit Base Identifier High Register 0 (TBIH0, Offset: 0x000E)

Bit	Name	Type	Description	Default Value
[7:0]	TBIH	RW	The high 8 bits of the base transmit identifier for TX buffer0 The 7 most significant bits must not be all recessive.	0

8.9.7.16. Reserved

Table 8-267. Reserved (Offset: 0x000F)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.17. Transmit Flexible Data-Rate Register 1

Table 8-268. Transmit flexible data-rate Register 1 (TFD1, Offset: 0x0010)

vangotech

Bit	Name	Type	Description	Default Value
7	TEDL	RW	Extended data length bit for TX buffer1 It only exists in CAN FD format frames, it distinguishes between CAN format and CAN FD format frames. In a CAN format frame, the dominant bit is transmitted instead of EDL. 1'b1: FD frame 1'b0: Non-FD frame	0
6	TBRS	RW	Bit rate switch bit for TX buffer1 It decides whether the bit rate is switched from the standard bit rate of the arbitration phase to the preconfigured alternate bit rate of the data phase inside a CAN FD format frame. 1'b1: Bit rate switch 1'b0: Bit rate doesn't switch.	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.18. Transmit Remote and Extended Identifier Enable Register 1

Table 8-269. Transmit Remote and Extended Identifier Enable Register 1 (TREIE1, Offset: 0x0011)

Bit	Name	Type	Description	Default Value
7	TRTR	RW	Transmit remote transmission request bit for TX buffer1 1'b1: Remote transmit frame request 1'b0: Data frame request Please note that, in CAN FD format frames, it must be set to 0, since there is no remote frame in CAN FD format. Otherwise, a form error would be detected.	0
6	TEIR	RW	Transmit extended identifier request bit for TX buffer1 1'b1: Use extended format for transmit request 1'b0: Use standard format for transmit request	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.19. Transmit Extended Identifier Low Register 1

Table 8-270. Transmit Extended Identifier Low Register 1 (TEIL1, Offset: 0x0012)

Bit	Name	Type	Description	Default Value
[7:6]	TEIL	RW	The last two bits of 18-bit extended transmit identifier for TX buffer1	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.20. Transmit Extended Identifier Middle Register 1

Table 8-271. Transmit Extended Identifier Middle Register 1 (TEIM1, Offset: 0x0013)

Bit	Name	Type	Description	Default Value

Vango Tech

[7:0]	TEIM	RW	The middle 8 bits of 18-bit extended transmit identifier for TX buffer1	0

8.9.7.21. Transmit Extended Identifier High Register 1

Table 8-272. Transmit Extended Identifier High Register 1 (TEIH1, Offset: 0x0014)

Bit	Name	Type	Description	Default Value
[7:0]	TEIH	RW	The most high 8 bits of 18-bit extended transmit identifier for TX buffer1	0

8.9.7.22. Transmit Base Identifier Low and Data Length Register 1

Table 8-273. Transmit Base Identifier Low and Data Length Register 1 (TBILDL1, Offset: 0x0015)

Bit	Name	Type	Description	Default Value
[7:5]	TBIL	RW	The low 3 bits of 11-bit base transmit identifier for TX buffer1	0
4	-	Rsvd	Reserved	0
[3:0]	TDL	RW	The transmit data payload length for TX buffer1 is as follows: 4'h0: 0 byte 4'h1: 1 byte 4'h2: 2 bytes 4'h3: 3 bytes 4'h4: 4 bytes 4'h5: 5 bytes 4'h6: 6 bytes 4'h7: 7 bytes 4'h8: 8 bytes 4'h 9~ 4'hf: 8 bytes (for CAN format) The following values are for CAN FD frame: 4'h8: 8 bytes 4'h9: 12 bytes 4'hA: 16 bytes 4'hB: 20 bytes 4'hC: 24 bytes 4'hD: 32 bytes 4'hE: 48 bytes 4'hF: 64 bytes	0

8.9.7.23. Transmit Base Identifier High Register 1

Table 8-274. Transmit Base Identifier High Register 1 (TBIH1, Offset: 0x0016)

Vangotech

Bit	Name	Type	Description	Default Value
[7:0]	TBIH	RW	The high 8 bits of base transmit identifier for TX buffer1 The 7 most significant bits must not be all recessive.	0

8.9.7.24. Reserved

Table 8-275. Reserved (Offset: 0x0017)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.25. Transmit Flexible Data-rate Register 2

Table 8-276. Transmit Flexible Data-rate Register 2 (TFD2, Offset: 0x0018)

Bit	Name	Type	Description	Default Value
7	TEDL	RW	Extended data length bit for TX buffer2 It only exists in CAN FD format frames, it distinguishes between CAN format and CAN FD format frames. In a CAN format frame, the dominant bit is transmitted instead of EDL. 1'b1: FD frame 1'b0: Non-FD frame	0
6	TBRS	RW	Bit rate switch bit for TX buffer2 It decides whether the bit rate is switched from the standard bit rate of the arbitration phase to the preconfigured alternate bit rate of the data phase inside a CAN FD format frame. 1'b1: Bit rate switch 1'b0: Bit rate doesn't switch.	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.26. Transmit Remote and Extended Identifier Enable Register 2

Table 8-277. Transmit Remote and Extended Identifier Enable Register 2 (TREIE2, Offset: 0x0019)

Bit	Name	Type	Description	Default Value
7	TRTR	RW	Transmit remote transmission request bit for TX buffer2 1'b1: Remote transmit frame request 1'b0: Data frame request Please note that, in CAN FD format frames, it must be set to 0, since here is no remote frame in CAN FD format. Otherwise, a form error would be detected.	0
6	TEIR	RW	Transmit extended identifier request bit for TX buffer2 1'b1: Use extended format for transmit request 1'b0: Use standard format for transmit request	0

Vangotech

Bit	Name	Type	Description	Default Value
[5:0]	-	Rsvd	Reserved	0

8.9.7.27. Transmit Extended Identifier Low Register 2

Table 8-278. Transmit Extended Identifier Low Register 2 (TEIL2, Offset: 0x001A)

Bit	Name	Type	Description	Default Value
[7:6]	TEIL	RW	The last two bits of 18-bit extended transmit identifier for TX buffer2	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.28. Transmit Extended Identifier Middle Register 2

Table 8-279. Transmit Extended Identifier Middle Register 2 (TEIM2, Offset: 0x001B)

Bit	Name	Type	Description	Default Value
[7:0]	TEIM	RW	The middle 8 bits of 18-bit extended transmit identifier for TX buffer2	0

8.9.7.29. Transmit Extended Identifier High Register 2

Table 8-280. Transmit Extended Identifier High Register 2 (TEIH2, Offset: 0x001C)

Bit	Name	Type	Description	Default Value
[7:0]	TEIH	RW	The most high 8 bits of 18-bit extended transmit identifier for TX buffer2	0

8.9.7.30. Transmit Base Identifier Low and Data Length Register 2

Table 8-281. Transmit Base Identifier Low and Data Length Register 2 (TBILDL2, Offset: 0x001D)

Bit	Name	Type	Description	Default Value
[7:5]	TBIL	RW	The low 3 bits of 11-bit base transmit identifier for TX buffer2	0
[4:0]	-	Rsvd	Reserved	0
[3:0]	TDL	RW	The transmit data payload length for TX buffer2 is as follows: 4'h0: 0 byte 4'h1: 1 byte 4'h2: 2 bytes 4'h3: 3 bytes 4'h4: 4 bytes 4'h5: 5 bytes 4'h6: 6 bytes 4'h7: 7 bytes 4'h8: 8 bytes	0

Vangotech

Bit	Name	Type	Description	Default Value
			4'h9 ~ 4'hf: 8 bytes (for CAN format) The following values are for CAN FD frame: 4'h8: 8 bytes 4'h9: 12 bytes 4'hA: 16 bytes 4'hB: 20 bytes 4'hC: 24 bytes 4'hD: 32 bytes 4'hE: 48 bytes 4'hF: 64 bytes	

8.9.7.31. Transmit Base Identifier High Register 2

Table 8-282. Transmit Base Identifier High Register 2 (TBIH2, Offset: 0x001E)

Bit	Name	Type	Description	Default Value
[7:0]	TBIH	RW	The high 8 bits of base transmit identifier for TX buffer2 The 7 most significant bits must not be all recessive.	0

8.9.7.32. Reserved

Table 8-283. Reserved (Offset: 0x001F)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.33. Acceptance Filter for Extended Identifier Low Bit Register 0

Table 8-284. Acceptance Filter for Extended Identifier Low Bit Register 0 (AFEIL0, Offset: 0x0020)

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.34. Acceptance Filter for Extended Identifier Middle Bit Register 0

Table 8-285. Acceptance Filter for Extended Identifier Middle Bit Register 0 (AFEIM0, Offset: 0x0021)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

Vangotech

8.9.7.35. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 0

Table 8-286. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 0 (AFBILEIH0, Offset: 0x0022)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0
[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.36. Acceptance Filter for Base Identifier High Bit Register 0

Table 8-287. Acceptance Filter for Base Identifier High Bit Register 0 (AFBIH0, Offset: 0x0023)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier	0

8.9.7.37. Acceptance Filter for Extended Identifier Low Bit Register 1

Table 8-288. Acceptance Filter for Extended Identifier Low Bit Register 1 (AFEIL1, Offset: 0x0024)

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.38. Acceptance Filter for Extended Identifier Middle Bit Register 1

Table 8-289. Acceptance Filter for Extended Identifier Middle Bit Register 1 (AFEIM1, Offset: 0x0025)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

8.9.7.39. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 1

Table 8-290. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 1 (AFBILEIH1, Offset: 0x0026)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0
[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.40. Acceptance Filter for Base Identifier High Bit Register 1

Table 8-291. Acceptance Filter for Base Identifier High Bit Register 1 (AFBIH1, Offset: 0x0027)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier	0

8.9.7.41. Acceptance Filter for Extended Identifier Low Bit Register 2

Table 8-292. Acceptance Filter for Extended Identifier Low Bit Register 2 (AFEIL2, Offset: 0x0028)

Vanguard

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.42. Acceptance Filter for Extended Identifier Middle Bit Register 2

Table 8-293. Acceptance Filter for Extended Identifier Middle Bit Register 2 (AFEIM2, Offset: 0x0029)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

8.9.7.43. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 2

Table 8-294. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 2 (AFBILEIH2, Offset: 0x002A)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0
[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.44. Acceptance Filter for Base Identifier High Bit Register 2

Table 8-295. Acceptance Filter for Base Identifier High Bit Register 2 (AFBIH2, Offset: 0x002B)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier	0

8.9.7.45. Acceptance Filter for Extended Identifier Low Bit Register 3

Table 8-296. Acceptance Filter for Extended Identifier Low Bit Register 3 (AFEIL3, Offset: 0x002C)

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.46. Acceptance Filter for Extended Identifier Middle Bit Register 3

Table 8-297. Acceptance Filter for Extended Identifier Middle Bit Register 3 (AFEIM3, Offset: 0x002D)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

8.9.7.47. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 3

Table 8-298. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 3 (AFBILEIH3, Offset: 0x002E)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0

vangotech

[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.48. Acceptance Filter for Base Identifier High Bit Register 3

Table 8-299. Acceptance Filter for Base Identifier High Bit Register 3 (AFBIH3, Offset: 0x002F)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier.	0

8.9.7.49. Acceptance Filter for Extended Identifier Low Bit Register 4

Table 8-300. Acceptance Filter for Extended Identifier Low Bit Register 4 (AFEIL4, Offset: 0x0030)

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.50. Acceptance Filter for Extended Identifier Middle Bit Register 4

Table 8-301. Acceptance Filter for Extended Identifier Middle Bit Register 4 (AFEIM4, Offset: 0x0031)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

8.9.7.51. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 4

Table 8-302. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 4 (AFBILEIH4, Offset: 0x0032)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0
[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.52. Acceptance Filter for Base Identifier High Bit Register 4

Table 8-303. Acceptance Filter for Base Identifier High Bit Register 4 (AFBIH4, Offset: 0x0033)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier	0

8.9.7.53. Acceptance Filter for Extended Identifier Low Bit Register 5

Table 8-304. Acceptance Filter for Extended Identifier Low Bit Register 5 (AFEIL5, Offset: 0x0034)

Bit	Name	Type	Description	Default Value
[7:3]	AFEIL	RW	The low 5 bits of acceptance filter for extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.54. Acceptance Filter for Extended Identifier Middle Bit Register 5

Table 8-305. Acceptance Filter for Extended Identifier Middle Bit Register 5 (AFEIM5, Offset: 0x0035)

Bit	Name	Type	Description	Default Value
[7:0]	AFEIM	RW	The middle 8 bits of acceptance filter for extended identifier	0

8.9.7.55. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 5

Table 8-306. Acceptance Filter for Base Identifier Low Bit and Extended Identifier High Bit Register 5 (AFBILEIH5, Offset: 0x0036)

Bit	Name	Type	Description	Default Value
[7:5]	AFBIL	RW	The low 3 bits of acceptance filter for base identifier	0
[4:0]	AFEIH	RW	The high 5 bits of acceptance filter for extended identifier	0

8.9.7.56. Acceptance Filter for Base Identifier High Bit Register 5

Table 8-307. Acceptance Filter for Base Identifier High Bit Register 5 (AFBIH5, Offset: 0x0037)

Bit	Name	Type	Description	Default Value
[7:0]	AFBIH	RW	The high 8 bits of acceptance filter for base identifier	0

8.9.7.57. Acceptance Filter Data Byte0 Register 0

Table 8-308. Acceptance Filter Data Byte0 Register 0 (AFDB00, Offset: 0x0038)

Bit	Name	Type	Description	Default Value
[7:0]	AFDB0	RW	The byte0 of acceptance data filter	0

8.9.7.58. Acceptance Filter Data Byte1 Register 0

Table 8-309. Acceptance Filter Data Byte1 Register 0 (AFDB10, Offset: 0x0039)

Bit	Name	Type	Description	Default Value
[7:0]	AFDB1	RW	The byte1 of acceptance data filter	0

8.9.7.59. Acceptance Filter Data Byte0 Register 1

Table 8-310. Acceptance Filter Data Byte0 Register 1 (AFDB01, Offset: 0x003A)

Bit	Name	Type	Description	Default Value
[7:0]	AFDB0	RW	The byte0 of acceptance data filter	0

8.9.7.60. Acceptance Filter Data Byte1 Register 1

Table 8-311. Acceptance Filter Data Byte1 Register 1 (AFDB11, Offset: 0x003B)

Bit	Name	Type	Description	Default Value
[7:0]	AFDB1	RW	The byte1 of acceptance data filter	0

8.9.7.61. Acceptance Filter Control Register 0

Table 8-312. Acceptance Filter Control Register 0 (AFC0, Offset: 0x003C)

Vangotech

Bit	Name	Type	Description	Default Value
7	RBAFG0E	RW	Enable receive buffer0 acceptance filter group 1'b1: Enable receive buffer0. By only setting this bit to "1", receive buffer0 is allowed to receive message. 1'b0: Disable receive buffer0	0
6	RBAFG1E	RW	Receive buffer1 acceptance filter group enable 1'b1: Enable receive buffer1. By only setting this bit to "1", receive buffer0 is allowed to receive message. 1'b0: Disable receive buffer1	0
5	RB0DFB0E	RW	Enable receive buffer0 data filter byte0 1'b1: Enable receive buffer0's data filter byte0. Controller will compare the received data's byte0 with AFDB00, the frame will be discarded when byte0 does not equal to AFDB00. 1'b0: Disable receive buffer0 data filter byte0	0
4	RB0DFB1E	RW	Enable receive buffer0 data filter byte1 1'b1: Enable receive buffer0's data filter byte1. Controller will compare the received data's byte1 with AFDB10, the frame will be discarded when byte0 does not equal to AFDB10. 1'b0: Disable receive buffer0 data filter byte1	0
3	RB1DFB0E	RW	Enable receive buffer1 data filter byte0 1'b1: Enable receive buffer1's data filter byte0. Controller will compare the received data's byte0 with AFDB01, the frame will be discarded when byte0 does not equal to AFDB01. 1'b0: Disable receive buffer1 data filter byte0	0
2	RB1DFB1E	RW	Enable receive buffer1 data filter byte1 1'b1: Enable receive buffer1's data filter byte1. Controller will compare the received data's byte1 with AFDB11, the frame will be discarded when byte0 does not equal to AFDB11. 1'b0: Disable receive buffer1 data filter byte1	0
[1:0]	-	Rsvd	Reserved	0

8.9.7.62. Acceptance Filter Control Register 1

Table 8-313. Acceptance Filter Control Register 1 (AFC1, Offset: 0x003D)

Bit	Name	Type	Description	Default Value
[7:6]	AFT0	RW	Acceptance frame type 0 2'b00: Both standard and extended frame can be accepted. 2'b01: Only accept standard frames 2'b10: Only accept extended frames 2'b11: Has not been defined	0

Vangotech

[5:4]	AFT1	RW	Acceptance frame type 1 2'b00: Both data and remote frame can be accepted. 2'b01: Only accept data frames 2'b10: Only accept remote frames 2'b11: Has not been defined	0
[3:2]	AFT2	RW	Acceptance frame type 2 2'b00: Both non-FD and FD frame can be accepted. 2'b01: Only accept non-FD frames 2'b10: Only accept FD frames 2'b11: Has not been defined	0
[1:0]	-	Rsvd	Reserved	0

8.9.7.63. Reserved

Table 8-314. Reserved (Offset: 0x003E)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.64. Reserved

Table 8-315. Reserved (Offset: 0x003F)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.65. Filter Mask Extended Identifier Low Register 0

Table 8-316. Filter Mask Extended Identifier Low Register 0 (FMEIL0, Offset: 0x0040)

Bit	Name	Type	Description	Default Value
[7:3]	FMEIL	RW	Mask bits for the lowest 5 bits of extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.66. Filter Mask Extended Identifier Middle Register 0

Table 8-317. Filter Mask Extended Identifier Middle Register 0 (FMEIM0, Offset: 0x0041)

Bit	Name	Type	Description	Default Value
[7:0]	FMEIM	RW	Mask bits for the middle 8 bits of extended identifier	0

8.9.7.67. Filter Mask Base Identifier Low and Extended Identifier High 0

Table 8-318. Filter Mask Base Identifier Low and Extended Identifier High 0 (FMBILEIH0, Offset: 0x0042)

Bit	Name	Type	Description	Default Value
[7:5]	FMBIL	RW	Mask bits for the lowest 3 bits of base identifier	0
[4:0]	FMEIH	RW	Mask bits for the high 5 bits of extended identifier	0

Vangotech

8.9.7.68. Filter Mask Base Identifier High Register 0

Table 8-319. Filter Mask Base Identifier High Register 0 (FMBIH0, Offset: 0x0043)

Bit	Name	Type	Description	Default Value
[7:0]	FMBIH	RW	Mask bits for the high 8 bits of base identifier	0

8.9.7.69. Filter Mask Extended Identifier Low Register 1

Table 8-320. Filter Mask Extended Identifier Low Register 1 (FMEIL1, Offset: 0x0044)

Bit	Name	Type	Description	Default Value
[7:3]	FMEIL	RW	Mask bits for the lowest 5 bits of extended identifier	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.70. Filter Mask Extended Identifier Middle Register 1

Table 8-321. Filter Mask Extended Identifier Middle Register 1 (FMEIM1, Offset: 0x0045)

Bit	Name	Type	Description	Default Value
[7:0]	FMEIM	RW	Mask bits for the middle 8 bits of extended identifier	0

8.9.7.71. Filter Mask Base Identifier Low and Extended Identifier High 1

Table 8-322. Filter Mask Base Identifier Low and Extended Identifier High 1 (FMBILEIH1, Offset: 0x0046)

Bit	Name	Type	Description	Default Value
[7:5]	FMBIL	RW	Mask bits for the lowest 3 bits of base identifier	0
[4:0]	FMEIH	RW	Mask bits for the high 5 bits of extended identifier	0

8.9.7.72. Filter Mask Base Identifier High Register 1

Table 8-323. Filter Mask Base Identifier High Register 1 (FMBIH1, Offset: 0x0047)

Bit	Name	Type	Description	Default Value
[7:0]	FMBIH	RW	Mask bits for the high 8 bits of base identifier	0

8.9.7.73. Filter Mask Data Byte0 Register 0

Table 8-324. Filter Mask Data Byte0 Register 0 (FMDB00, Offset: 0x0048)

Bit	Name	Type	Description	Default Value
[7:0]	FMDB0	RW	Mask bits for acceptance data byte 0 filter	0

8.9.7.74. Filter Mask Data Byte1 Register 0

Table 8-325. Filter Mask Data Byte1 Register 0 (FMDB10, Offset: 0x0049)

Bit	Name	Type	Description	Default Value
[7:0]	FMDB1	RW	Mask bits for the acceptance data byte 1 filter	0

8.9.7.75. Filter Mask Data Byte0 Register 0

Table 8-326. Filter Mask Data Byte0 Register 1 (FMDB01, Offset: 0x004A)

Vangotech

Bit	Name	Type	Description	Default Value
[7:0]	FMDB0	RW	Mask bits for acceptance data byte 0 filter	0

8.9.7.76. Filter Mask Data Byte1 Register 1

Table 8-327. Filter Mask Data Byte1 Register 1 (FMDB11, Offset: 0x004B)

Bit	Name	Type	Description	Default Value
[7:0]	FMDB1	RW	Mask bits for the acceptance data byte 1 filter	0

8.9.7.77. Filter Mask Control Register

Table 8-328. Filter Mask Control Register (FMC, Offset: 0x004C)

Bit	Name	Type	Description	Default Value
7	FMA	RW	Filter all mask 1'b1: All messages can pass the filter when the corresponding buffer acceptance filter group enable bit is set. 1'b0: Disable all mask	0
6	RBAFM0E	RW	Enable receive buffer0 acceptance filter mask 1'b1: Enable receive buffer0 ID filter mask 1'b0: Disable receive buffer0 ID filter mask	0
5	RBAFM1E	RW	Enable receive buffer1 acceptance filter mask 1'b1: Enable receive buffer1 ID filter mask 1'b0: Disable receive buffer1 ID filter mask	0
4	RB0DFMB0E	RW	Enable receive buffer0 data filter mask for byte 1'b1: Enable byte0 of receive buffer0 data filter mask 1'b0: Disable byte0 of receive buffer0 data filter mask	0
3	RB0DFMB1E	RW	Enbable receive buffer0 data filter mask for byte1 1'b1: Enable byte1 of receive buffer0 data filter mask 1'b0: Disable byte1 of receive buffer0 data filter mask	0
2	RB1DFMB0E	RW	Enable receive buffer1 data filter mask enable for byte0 1'b1: Enable byte0 of receive buffer1 data filter mask 1'b0: Disable byte0 of receive buffer1 data filter mask	0
1	RB1DFMB1E	RW	Enable receive buffer1 data filter mask enable for byte1 1'b1: Enable byte1 of receive buffer1 data filter mask 1'b0: Disable byte1 of receive buffer1 data filter mask	0
0	-	Rsvd	Reserved	0

8.9.7.78. Reserved

Table 8-329. Reserved (Offset: 0x004D)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

Vangotech

8.9.7.79. Reserved

Table 8-330. Reserved (Offset: 0x004E)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.80. Reserved

Table 8-331. Reserved (Offset: 0x004F)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.81. Received Flexible Data-Rate Register 0

Table 8-332. Received Flexible Data-Rate Register 0 (RFD0, Offset: 0x0050)

Bit	Name	Type	Description	Default Value
7	REDL	RO	The received extended data length bit 1'b1: The received frame is FD frame. 1'b0: The received frame is non-FD frame.	0
6	RBRS	RO	The received bit rate switch bit 1'b1: The bit rate is switched. 1'b0: The bit rate is not switched.	0
5	RESI	RO	The received ESI bit 1'b1: The received frame is from error passive node. 1'b0: The received frame is from error active node. ESI does not exist in CAN format frames.	0
[4:0]	-	Rsvd	Reserved	0

8.9.7.82. Received Remote and Extended Identifier Enable Register 0

Table 8-333. Received Remote and Extended Identifier Enable Register 0 (RREIE0, Offset: 0x0051)

Bit	Name	Type	Description	Default Value
7	RRTR	RO	Received remote transmission request bit 1'b0: The received frame is data frame. 1'b1: The received frame is remote frame. There is no remote frame in CAN FD format.	0
6	REIE	RO	Received ID type 1'b0: The received ID is standard ID. 1'b1: The received ID is extended ID.	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.83. Received Extended Identifier Low Register 0

Table 8-334. Received Extended Identifier Low Register 0 (REIL0, Offset: 0x0052)

Vangotech

Bit	Name	Type	Description	Default Value
[7:6]	REIL	RO	The low 2 bits of the received extended identifier	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.84. Received Extended Identifier Middle Register 0

Table 8-335. Received Extended Identifier Middle Register 0 (REIM0, Offset: 0x0053)

Bit	Name	Type	Description	Default Value
[7:0]	REIM	RO	The middle 8 bits of received extended identifier	0

8.9.7.85. Received Extended Identifier High Register 0

Table 8-336. Received Extended Identifier High Register 0 (REIH0, Offset: 0x0054)

Bit	Name	Type	Description	Default Value
[7:0]	REIH	RO	The high 8 bits of received extended identifier	0

8.9.7.86. Received Base Identifier Low and Data Length Register 0

Table 8-337. Received Base Identifier Low and Data Length Register 0 (RBILDL0, Offset: 0x0055)

Bit	Name	Type	Description	Default Value
[7:5]	RBIL	RO	The low 3 bits of received base identifier	0
4	-	Rsvd	Reserved	0
[3:0]	RDL	RO	The received data payload length is as follows: 4'h0: 0 byte 4'h1: 1 byte 4'h2: 2 bytes 4'h3: 3 bytes 4'h4: 4 bytes 4'h5: 5 bytes 4'h6: 6 bytes 4'h7: 7 bytes 4'h8: 8 bytes 4'h9 ~ 4'hf: 8 bytes (for CAN format) The following values are for CAN FD frame: 4'h8: 8 bytes 4'h9: 12 bytes 4'hA: 16 bytes 4'hB: 20 bytes 4'hC: 24 bytes 4'hD: 32 bytes 4'hE: 48 bytes 4'hF: 64 bytes	0

Vangotech

8.9.7.87. Received Base Identifier High Register 0

Table 8-338. Received Base Identifier High Register 0 (RBIH0, Offset: 0x0056)

Bit	Name	Type	Description	Default Value
[7:0]	RBIH	RO	The high 8 bits of received base identifier	0

8.9.7.88. Reserved

Table 8-339. Reserved (Offset: 0x0057)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.89. Received Flexible Data-Rate Register 1

Table 8-340. Received Flexible Data-Rate Register 1 (RFD1, Offset: 0x0058)

Bit	Name	Type	Description	Default Value
7	REDL	RO	The received extended data length bit 1'b1: The received frame is FD frame. 1'b0: The received frame is non-FD frame.	0
6	RBRS	RO	The received bit rate switch bit 1'b1: The bit rate is switched. 1'b0: The bit rate is not switched.	0
5	RESI	RO	The received ESI bit 1'b1: The received frame is from error passive node. 1'b0: The received frame is from error active node. ESI does not exist in CAN format frames.	0
[4:0]	-	Rsvd	Reserved	0

8.9.7.90. Received Remote and Extended Identifier Enable Register 1

Table 8-341. Received Remote and Extended Identifier Enable Register 1 (RREIE1, Offset: 0x0059)

Bit	Name	Type	Description	Default Value
7	RRTR	RO	Received remote transmission request bit 1'b0: The received frame is data frame. 1'b1: The received frame is remote frame. There is no remote frame in CAN FD format.	0
6	REIE	RO	Received ID type 1'b0: The received ID is standard ID. 1'b1: The received ID is extended ID.	0
[5:0]	-	Rsvd	Reserved	0

8.9.7.91. Received Extended Identifier Low Register 1

Table 8-342. Received Extended Identifier Low Register 1 (REIL1, Offset: 0x005A)

Bit	Name	Type	Description	Default Value
[7:6]	REIL	RO	The low 2 bits of the received extended identifier	0

Vangotech

[5:0]	-	Rsvd	Reserved.	0

8.9.7.92. Received Extended Identifier Middle Register 1

Table 8-343. Received Extended Identifier Middle Register 1 (REIM1, Offset: 0x005B)

Bit	Name	Type	Description	Default Value
[7:0]	REIM	RO	The middle 8 bits of received extended identifier	0

8.9.7.93. Received Extended Identifier High Register 1

Table 8-344. Received Extended Identifier High Register 1 (REIH1, Offset: 0x005C)

Bit	Name	Type	Description	Default Value
[7:0]	REIH	RO	The high 8 bits of Received extended identifier	0

8.9.7.94. Received Base Identifier Low and Data Length Register 1

Table 8-345. Received Base Identifier Low and Data Length Register 1 (RBILDL1, Offset: 0x005D)

Bit	Name	Type	Description	Default Value
[7:5]	RBIL	RO	The low 3 bits of received base identifier	0
4	-	Rsvd	Reserved	0
[3:0]	RDL	RO	The received data payload length is as follows: 4'h0: 0 byte 4'h1: 1 byte 4'h2: 2 bytes 4'h3: 3 bytes 4'h4: 4 bytes 4'h5: 5 bytes 4'h6: 6 bytes 4'h7: 7 bytes 4'h8: 8 bytes 4'h9 ~ 4'hf: 8 bytes (for CAN format) The following values are for CAN FD frame: 4'h8: 8 bytes 4'h9: 12 bytes 4'hA: 16 bytes 4'hB: 20 bytes 4'hC: 24 bytes 4'hD: 32 bytes 4'hE: 48 bytes 4'hF: 64 bytes	0

8.9.7.95. Received Base Identifier High Register 1

Table 8-346. Received Base Identifier High Register 1 (RBIH1, Offset: 0x005E)

Bit	Name	Type	Description	Default Value

Vangotech

[7:0]	RBIH	RO	The high 8 bits of received base identifier	0

8.9.7.96. Reserved

Table 8-347. Reserved (Offset: 0x005F)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.97. Nominal Bit Timing Configuration 0 Register

Table 8-348. Nominal Bit Timing Configuration 0 Register (NBTC0, Offset: 0x0060)

Bit	Name	Type	Description	Default Value
[7:3]	NSJW	RW	The nominal synchronization jump width length bits [0:31]: 1 ~ 32 TQ which should be set less than the minimum value. (phase_seg1, phase_seg2)	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.98. Nominal Bit Timing Configuration 1 Register

Table 8-349. Nominal Bit Timing Configuration 1 Register (NBTC1, Offset: 0x0061)

Bit	Name	Type	Description	Default Value
[7:0]	NBRP	RW	Nominal baud rate prescaler [1:255]: TQ = (NBRP + 1)/Fosc NBRP cannot be set to 0.	1

8.9.7.99. Nominal Bit Timing Configuration 2 Register

Table 8-350. Nominal Bit Timing Configuration 2 Register (NBTC2, Offset: 0x0062)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	Reserved	0
[6:1]	NPRSEG	RW	The bits of nominal Prop_seg [0:63]: 1 ~ 64 TQ (= NPRSEG + 1)	0
0	-	Rsvd	Reserved	0

8.9.7.100. Nominal Bit Timing Configuration 3 Register

Table 8-351. Nominal Bit Timing Configuration 3 Register (NBTC3, Offset: 0x0063)

Bit	Name	Type	Description	Default Value
[7:3]	NPSEG1	RW	The bits of nominal Phase_seg1 [0:31]: 1 ~ 32 TQ (= NPSEG1 + 1)	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.101. Nominal Bit Timing Configuration 4 Register

Table 8-352. Nominal Bit Timing Configuration 4 Register (NBTC4, Offset: 0x0064)

Vangotech

Bit	Name	Type	Description	Default Value
[7:3]	NPSEG2	RW	The bits of nominal Phase_seg2 [0:31]: 1 ~ 32 TQ (= NPSEG2 + 1)	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.102. Reserved

Table 8-353. Reserved (Offset: 0x0065)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.103. Reserved

Table 8-354. Reserved (Offset: 0x0066)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.104. Reserved

Table 8-355. Reserved (Offset: 0x0067)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.105. Data Bit Timing Configuration 0 Register

Table 8-356. Data Bit Timing Configuration 0 Register (DBTC0, Offset: 0x0068)

Bit	Name	Type	Description	Default Value
[7:0]	DBRP	RW	Data baud rate prescaler [1:255]: TQ = (DBRP + 1)/Fosc DBRP cannot be set to 0.	1

8.9.7.106. Data Bit TimingConfiguration 1 Register

Table 8-357. Data Bit Timing Configuration 1 Register (DBTC1, Offset: 0x0069)

Bit	Name	Type	Description	Default Value
7	-	Rsvd	Reserved	0
[6:3]	DPRSEG	RW	The length of PROP_SEG for data bit time [0:8]: 0 ~ 8 TQ (= DPRSEG)	0
[2:1]	DSJW	RW	The data synchronization jump width length bits 2'b00: 1xTQ 2'b01: 2xTQ 2:b10: 3xTQ 2:b11: 4xTQ	0
0	-	Rsvd	Reserved	0

8.9.7.107. Data Bit Timing Configuration 2 Register

Table 8-358. Data Bit Timing Configuration 2 Register (DBTC2, Offset: 0x006A)

Bit	Name	Type	Description	Default Value
[7:5]	DPSEG1	RW	The bits of data Phase_seg1 [0:7]: 1 ~ 8 TQ (= DPSEG1 + 1)	0
[4:0]	-	Rsvd	Reserved	0

8.9.7.108. Data Bit Timing Configuration 3 Register

Table 8-359. Data Bit Timing Configuration 3 Register (DBTC3, Offset: 0x006B)

Bit	Name	Type	Description	Default Value
[7:5]	DPSEG2	RW	The bits of data Phase_seg2 [0:7]: 1 ~ 8 TQ (= DPSEG2 + 1)	0
[4:0]	-	Rsvd	Reserved	0

8.9.7.109. Receiver Error Counter Low Register

Table 8-360. Receiver Error Counter Low Register (RECL, Offset: 0x006C)

Bit	Name	Type	Description	Default Value
[7:0]	RECL	RO	Receive error counter low 8 bits register	0

8.9.7.110. Receiver Error Counter HighRegister

Table 8-361. Receiver Error Counter High Register (RECH, Offset: 0x006D)

Bit	Name	Type	Description	Default Value
[7:0]	RECH	RO	Receive error counter high 8 bits register	0

8.9.7.111. Transmitter Error Counter Low Register

Table 8-362. Transmitter Error Counter Low Register (TECL, Offset: 0x006E)

Bit	Name	Type	Description	Default Value
[7:0]	TECL	RO	Transmit error counter low 8 bits register	0

8.9.7.112. Transmitter Error Counter High Register

Table 8-363. Transmitter Error Counter High Register (TECH, Offset: 0x006F)

Bit	Name	Type	Description	Default Value
[7:0]	TECH	RO	Transmit error counter high 8 bits register	0

8.9.7.113. Error Type Register

Table 8-364. Error Type Register (ET, Offset: 0x0070)

Bit	Name	Type	Description	Default Value
7	BE	RO	Bit error	0

vangotech

6	SE	RO	Stuff error	0
5	CE	RO	CRC error	0
4	FE	RO	Form error	0
3	AE	RO	Acknowledgement error	0
2	OE	RO	Overrun error	0
[1:0]	-	Rsvd	Reserved	0

8.9.7.114. Transmitter and Receive Error Detected Register

Table 8-365. Transmitter and Receive Error Detected Register (TED, Offset: 0x0071)

Bit	Name	Type	Description	Default Value
7	BTE0	RO	TX buffer0 transmit error 1'b1: TX buffer0 transmit error 1'b0: No TX buffer0 transmit error	0
6	BTE1	RO	TX buffer1 transmit error 1'b1: TX buffer1 transmit error 1'b0: No TX buffer1 transmit error	0
5	BTE2	RO	TX buffer2 transmit error 1'b1: TX buffer2 transmit error 1'b0: No TX buffer2 transmit error	0
4	BRE0	RO	RX buffer0 receive error 1'b1: RX buffer0 receive error 1'b0: No RX buffer0 receive error Note: Due to arbitration loss is not considered as an error.	0
3	BRE1	RO	RX buffer1 receive error 1'b1: RX buffer1 receive error 1'b0: No RX buffer1 receive error Note: Due to arbitration loss is not considered as an error.	0
[2:0]	-	Rsvd	Reserved	0

8.9.7.115. Reserved

Table 8-366. Reserved (Offset: 0x0072)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

8.9.7.116. Reserved

Table 8-367. Reserved (Offset: 0x0073)

Bit	Name	Type	Description	Default Value
[7:0]	Reserved	Rsvd	Reserved	0

Vangotech

8.9.7.117. Receive Buffer Time Stamp Low 0

Table 8-368. Receive Buffer Time Stamp Low 0 (RBTSL0, Offset: 0x0074)

Bit	Name	Type	Description	Default Value
[7:0]	RBTSL	RO	Receive buffer0 time stamp low 8 bits	0

8.9.7.118. Receive Buffer Time Stamp High 0

Table 8-369. Receive Buffer Time Stamp High 0 (RBTSH0, Offset: 0x0075)

Bit	Name	Type	Description	Default Value
[7:0]	RBTSH	RO	Receive buffer0 time stamp high 8 bits	0

8.9.7.119. Receive Buffer Time Stamp Low 1

Table 8-370. Receive Buffer Time Stamp Low 1 (RBTSL1, Offset: 0x0076)

Bit	Name	Type	Description	Default Value
[7:0]	RBTSL	RO	Receive buffer1 time stamp low 8 bits	0

8.9.7.120. Receive Buffer Time Stamp High 1

Table 8-371. Receive Buffer Time Stamp High 1(RBTSH1, Offset: 0x0077)

Bit	Name	Type	Description	Default Value
[7:0]	RBTSH	RO	Receive buffer1 time stamp high 8 bits	0

8.9.7.121. Interrupt Request Register

Table 8-372. Interrupt Request Register (IR, Offset: 0x0078)

Bit	Name	Type	Description	Default Value
7	WIR	RW1C	Wake up interrupt request 1'b1: A wake up interrupt occurs. This bit can be set when WIE is 1 in IRE register. WIR is cleared by writing '1'or by an external reset. 1'b0: No wake up interrupt	0
6	OIR	RW1C	Overrun interrupt request 1'b1: An overrun interrupt occurs. This bit can be set when OIE is 1 in IRE register. OIR is cleared by writing '1' or by an external reset. 1'b0: No overrun interrupt request	0
5	EIR	RW1C	Error interrupt request 1'b1: An error interrupt request occurs. This bit can be set when EIE is 1 in IRE register. EIR is cleared by writing '1' or by an external reset. 1'b0: No error interrupt request	0
4	TBI0	RW1C	Transmit buffer0 interrupt 1'b1: A transmit buffer0 interrupt occurs. This indicates that transmit buffer0 has transmitted a frame successfully. This bit can be set when TIE is 1 in IRE register. TBI0 is cleared by writing '1' or by an external	0

Vangotech

			reset. 1'b0: No transmit buffer0 interrupt	
3	TBI1	RW1C	Transmit buffer1 interrupt 1'b1: A transmit buffer1 interrupt occurs. This indicates that transmit buffer1 has transmitted a frame successfully. This bit can be set when TIE is 1 in IRE register. TBI1 is cleared by writing '1' or by an external reset. 1'b0: No transmit buffer1 interrupt	0
2	TBI2	RW1C	Transmit buffer2 interrupt 1'b1: A transmit buffer2 interrupt occurs. This indicates that transmit buffer2 has transmitted a frame successfully. This bit can be set when TIE is 1 in IRE register. TBI2 is cleared by writing '1' or by an external reset. 1'b0: No transmit buffer2 interrupt	0
1	RBI0	RW1C	Receive buffer0 interrupt 1'b1: A receive buffer0 interrupt occurs. This indicates that receive buffer0 has received a frame successfully. This bit can be set when RIE is 1 in IRE register. RBI0 is cleared by writing '1' or by an external reset. 1'b0: No receive buffer0 interrupt	0
0	RBI1	RW1C	Receive buffer1 interrupt 1'b1: A receive buffer1 interrupt occurs. This indicates that receive buffer1 has received a frame successfully. This bit can be set when RIE is 1 in IRE register. RBI1 is cleared by writing '1' or by an external reset. 1'b0: No receive buffer1 interrupt	0

8.9.7.122. Transmit Buffer0 Data Register

Table 8-373. Transmit Buffer0 Data Register (TB0D, Offset: 0x007C ~ 0x00BB)

Bit	Name	Type	Description	Default Value
0x007C ~ 0x00BB	TB0	RW	Transmit buffer0 data register This buffer is 64 bytes long. For how many bytes to transmit is according to TDL bits in TBILDL0 register. Transmit low address data first.	0

8.9.7.123. Transmit Buffer1 Data Register

Table 8-374. Transmit Buffer1 Data Register (TB1D, Offset: 0x00BC ~ 0x00FB)

Bit	Name	Type	Description	Default Value
0x00BC ~ 0x00FB	TB1	RW	Transmit buffer1 data register This buffer is 64 bytes long. For how many bytes to transmit is according to TDL bits in TBILDL1 register. Transmit low address	0

Vangotech

			data first.	

8.9.7.124. Transmit Buffer2 Data Register

Table 8-375. Transmit Buffer2 Data Register (TB2D, Offset: 0x00FC ~ 0x013B)

Bit	Name	Type	Description	Default Value
0x00FC ~ 0x0013B	TB2	RW	Transmit buffer2 data register This buffer is 64 bytes long. For how many bytes to transmit is according to TDL bits in TBILDL2 register. Transmit low address data first.	0

8.9.7.125. Receive Buffer0 Data Register

Table 8-376. Receive Buffer0 Data Register (RB0D, Offset: 0x0013C ~ 0x017B)

Bit	Name	Type	Description	Default Value
0x0013C ~ 0x0017B	RB0	RO	Receive buffer0 data register This buffer is 64 bytes long. The received data is stored from low address to high address. For how many bytes to receive is according to RDL bits in RBILDL0 register.	0

8.9.7.126. Receive Buffer1 Data Register

Table 8-377. Receive Buffer1 Data Register (RB1D, Offset: 0x0017C ~ 0x01B8)

Bit	Name	Type	Description	Default Value
0x0013C ~ 0x0017B	RB1	RO	Receive buffer1 data register This buffer is 64 bytes long. The received data is stored from low address to high address. For how many bytes to receive is according to RDL bits in RBILDL1 register.	0

Vangotech

8.10. KBC

8.10.1. Overview

Keyboard controller (FTKBC010) is compliant with the IBM PS/2 interface. The interface uses the bidirectional clock and data lines to perform the half-duplex synchronous serial interface. It also provides the configurable scan matrix for the embedded or external keypad.

FTKBC010 performs a serial-to-parallel conversion to receive the data from the keyboard interface,and a parallel-to-serial conversion for transmitting the data to the keyboard interface. CPU accesses data and the control/status information via the AMBA APB interface.

8.10.2. Features

- Compliant with AMBA APB, Revision 2.0
- Includes IBM PS/2-compatible keyboard/mouse interface
- Programmable internal frequency for data transmitter and receiver synchronization
- Programmable reference clock divider to generate system clock
- Configurable scan matrix keypad from 4x4 to 8x16

8.10.3. Block Diagram

Figure 8-45 shows the block diagram of the keyboard/mouse interface controller.

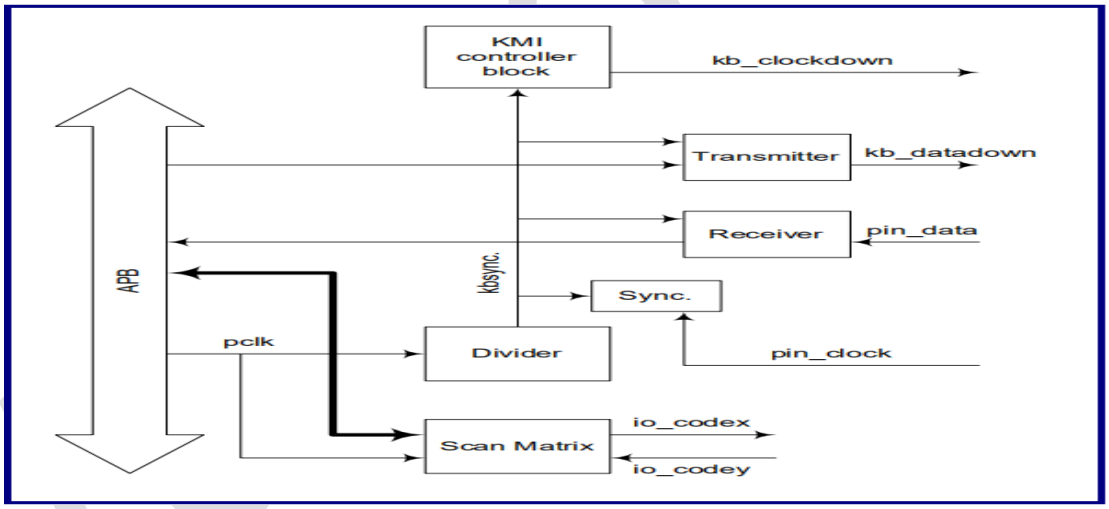


Figure 8-45. Functional Block Diagram

8.10.3.1. APB Interrupt Status and Control Register

All registers can be read and written via the APB bus protocol.

8.10.3.2. Divider

The divider is used to generate the sampling rate according to the outgoing reference clock (Usually the APB bus clock). The sampling rate is used to generate a synchronized signal to sample the clock and data signals.

8.10.3.3. Receiver

The receiver block performs a serial-to-parallel conversion by using the pin_data signal on the serial data stream receiver.

8.10.3.4. Transmitter

The transmitter block converts the parallel transmitting data into a serial bit-stream with a rate depending on the incoming pin_clock signal.

8.10.3.5. Sync.Logic

The synchronous logic circuit is used to synchronize the pin_clock signal for pin_data (Input) and kb_datadown (Output).

8.10.3.6. KMI Controller Logic

The controller block contains a bit counter and a timer circuit to decode the KMI clock and data status.

8.10.3.7. Scan Matrix

The scan matrix block is used to detect the keypad status.

This interface can be used in two different modes:

- Polling mode

  In this mode, a polling routine continuously monitors the keypad.
- Initial interrupt mode

  In this mode, the column (X) outputs are all set to low. When a key is pressed, an interrupt will be asserted.

8.10.4. Operation Procedure

8.10.4.1. Serial Data Frame on the Keyboard/Mouse Interface

When data are transmitted and received on the data line, the value will be a serial of 11-bit data frames,as illustrated in Figure 8-46.

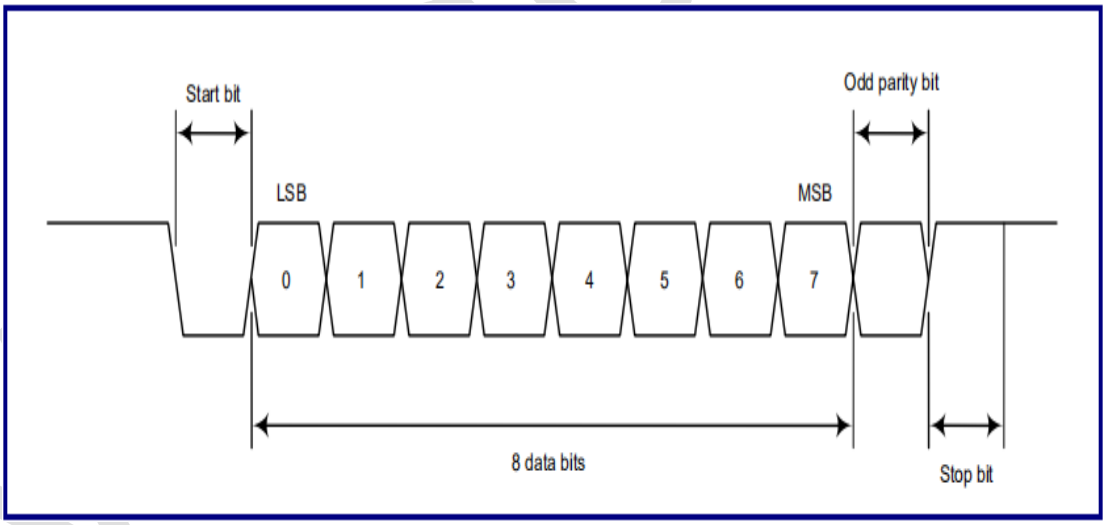


Figure 8-46. Serial Data Frame Format

The parity bit is used for error checking. An odd parity scheme is performed such that:

- If the even number of the data bits is set to '1', the parity bit wil be asserted high.
- If the odd number of the data bits is set to '1', the parity bit will be asserted low.

4.1.1.3.7 Transmitting Data from FTKBC010 to Keyboard/Mouse

When FTKBC010 is ready to transmit data to the keyboard/mouse, this controller will check if the keyboard/mouse is transmitting data.

- If the keyboard/mouse is transmitting data but has not reached the tenth clock, FTKBC010 can override the keyboard/mouse output by forcing the clock to low.

- If the keyboard/mouse transmission has reached the tenth clock, FTKBC010 must receive the transmission.

4.1.1.3.8 Receiving Data from Keyboard/Mouse to FTKBC010

When the keyboard/mouse is ready to transmit data to FTKBC010, it will check for the Inhibit status or FTKBC010 Request-To-Send status on the clock and data lines.

- If the clock line is low (Inhibit status), data values will be stored in the keyboard/mouse buffer.
- If the clock line is high and data line is low (Request-To-Send status), data values will be stored in the keyboard/mouse buffer, and the keyboard/mouse receives data from FTKBC010.
- If both clock and data lines are high, the keyboard/mouse transmits data to FTKBC010.

Each data transmission to the keyboard/mouse requires a response from the keyboard/mouse before FTKBC010 can transmit data again. The keyboard/mouse should respond within 16 ms, unless FTKBC010 prevents the response from the keyboard/mouse.

8.10.4.2. Keypad Interface

The keypad interface consists of two bidirectional ports: X and Y. FTKBC010 can configure the scan matrix ranging from 4x4 to 8x16.

4.1.1.3.9 One Key-pushed Operation

When a key is pressed, an interrupt will be asserted after the corresponding values are stored in the Keypad Code Y Register. FTKBC010 can scan the key-pushed operation by using the manual-scan method,which manually assigns values to the Keypad Code X Register, or using the auto-scan method, which provides the automatic scan code.

4.1.1.3.10 Two Key-pushed Operations in Auto-scan Mode

FTKBC010 accepts two-key pushes at a time, but only in auto-scan mode. When two keys are pressed simultaneously, an interrupt will be asserted after the corresponding values are stored in the Keypad Code Y Register.

8.10.5. Register List

Module Name	Base Address
KBC	0x5610 0000

Base Offset (Hex)	Name	Size (Byte)	Type	Description	Default Value (Hex)
00	CTRL	4	RW, WO	Control register	0000_0000
04	SRDR	4	RW	Sampling Rate Division Register	0000_0000
08	RSCR	4	RW	Request to Send Counter Register	0000_0000
0c	STS	4	RO	Status Register	0000_0010
10	ISR	4	RO	Interrupt Status Register	0000_0000
14	KRR	4	RO	Keyboard Receive Register	0000_0000
18	KTR	4	RW	Keyboard Transmit Register	0000_0000
1C	IMR	4	RW	Interrupt Mask Register	0000_0000
30	KCXR	4	RW, RO	Keypad Code X Register	0000_0000
34	KCYR	4	RO	Keypad Code Y Register	FFFF_FFFF
38	ASPR	4	RW	Auto-scan Period Register	0000_0000

Vangotech

50	REV	4	RO	Revision Register	0000_0000
54	FEATURE	4	RO	Feature Register	0000_0000

8.10.6. Register Description

8.10.6.1. Control Register

Table 8-378. Control Register (CTRL, Offset: 0x00)

Bit	Name	Type	Description	Default Value
[31:12]	-	Rsvd	Reserved	0
11	kbrecintmod	RW	When set, this bit records two key-pushed operations in the auto-scan mode.	0
10	clpadint	WO	When set, this bit clears the keypad interrupt.	0
9	scanmethod	RW	Scan method of the keypad 0: Use the manual-scan method 1: Use the automatic scan method	0
8	enpad	RW	When set, this bit enables the keypad.	0
7	clrxint	WO	When set, this bit clears the receive interrupt.	0
6	cltxint	WO	When set, this bit clears the transmit interrupt.	0
5	enackbit	RW	FTKBC010 type The majority (Keyboard/Mouse) applies the line control bit mode, which adds an acknowledge bit at the end of a serial data frame. 0: Support the line control bit (Default) 1: Not support the line control bit (Set to '1' to use a keyboard that does not issue a line control bit)	0
4	enrxint	RW	When set, this bit enables the receive interrupt.	0
3	entxint	RW	When set, this bit enables the transmit interrupt. [1]	0
2	enkbc	RW	When set, this bit enables FTKBC010.	0
1	datadown	RW	When set, io_data is driven to zero regardless of the state of FTKBC010.	0
0	clockdown	RW	When set, io_clock is driven to zero, which informs the keyboard/mouse not to transmit data to FTKBC010.	0

[1] When bit 5 of the control register is set to '0', "entxint" must be set to '1'.

Vangotech

8.10.6.2. Sampling Rate Division Registers

Table 8-379. Sampling Rate Division register(SRDR, Offset: 0x04)

Bit	Name	Type	Description	Default Value
[31:6]	-	Rsvd	Reserved	0
[5:0]	kbsrate	RW	FTKBC010 sampling rate setup In general, the sampling rate equals to 8 MHz. Sampling rate = (pclk)/(kbsrate + 1)	0

8.10.6.3. Request to Send Counter Register

Table 8-380. Request to Send Counter Register (RSCR, Offset 0x08)

Bit	Name	Type	Description	Default Value
[31:14]	-	Rsvd	Reserved	0
[13:0]	kbcnt	RW	Period of request-to-send setup The value of kbcnt should be determined by equation, (1/pclk) * kbcnt, which almost equals to 160 μs to meet the PS/2 specification.	0

8.10.6.4. Status Register (Offset: 0x0C)

Table 8-381. Status Register (STS, Offset: 0x0C)

Bit	Name	Type	Description	Default Value
[31:5]	-	Rsvd	Reserved	0
4	TxEmpty	RO	0: Tx buffer is full 1: Tx buffer is empty and ready to be written.	0
3	TxBusy	RO	0: Idle 1: Busy sending data	0
2	RxFull	RO	0: Rx buffer is empty 1: Rx buffer is full and ready to be read.	0
1	RxBusy	RO	0: Idle 1: Busy receiving data	0
0	RxParity	RO	Received parity bit It is an odd parity bit for receiving the data from the keyboard/mouse.	0

8.10.6.5. Interrupt Status Register (Offset: 0x10)

Table 8-382. Interrupt Status Register (ISR, Offset: 0x10)

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
2	kbpadint	RO	When set, the keypad interrupt is asserted.	0
1	kbtxint	RO	When set, the transmit interrupt is asserted.	0
0	kbrxint	RO	When set, the receive interrupt is asserted.	0

Vangotech

8.10.6.6. Keyboard Receive Register

Table 8-383. Keyboard Receive Register (KRR, Offset 0x14)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	kbrx	RO	Receive data register for the data received from the keyboard/mouse	0

8.10.6.7. Keyboard Transmit Register

Table 8-384. Keyboard Transmit Register (KTR, Offset: 0x18)

Bit	Name	Type	Description	Default Value
[31:8]	-	Rsvd	Reserved	0
[7:0]	kbtx	RW	Transmit data register for the data transmitted from FTKBC010	0

8.10.6.8. Interrupt Mask Register (Offset 0x1C)

Table 8-385. Interrupt Mask Register (IMR, Offset 0x1C)

Bit	Name	Type	Description	Default Value
[31:3]	-	Rsvd	Reserved	0
2	mpadint	RW	Mask the keypad interrupt	0
1	mtxint	RW	When set, mask the transmit interrupt	0
0	mrxint	RW	When set, mask the receive interrupt	0

8.10.6.9. Keypad Code X Register

Table 8-386. Keypad Code X Register (KCXR, Offset 0x30)

Bit	Name	Type	Description	Default Value
31	padintb	RO	When set, the CodexH interrupt is asserted. This bit cleared when Control Register bit[10] is set.	0
[23:16]	kbcodexH	RO	The second scan code output to the keypad This register is meaningful only when bit 11 of the control register is set to '1'.	0
15	padinta	RO	When set, the CodexL interrupt is asserted. This bit cleared when Control Register bit[10] is set.	0
[7:0]	kbcodexL	RW	The first scan code output to the keypad Writing in the manual-scan mode	0

8.10.6.10. Keypad Code Y Register

Table 8-387. Keypad Code Y Register (KCYR, Offset 0x34)

Bit	Name	Type	Description	Default Value
[31:16]	kbcodeyH	RO	The second code scan result This register is meaningful only when bit 11 of the	0

			control register is set to '1'.	
[15:0]	kbcodeyL	RO	The first code scan result	0

8.10.6.11. Auto-Scan Period Register

Table 8-388. Auto-Scan Period Register (ASPR, Offset 0x38)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:0]	kbpadcnt	RW	Auto-scan period for the keypad This register is meaningful when bit 9 of the control register is set to '1'. The smaller the value is, the faster the scan rate will be. Range: 0x1 ~ 0xFFFFFF; the assigned value depends on the dynamic rate of the manipulator.	0

8.10.6.12. Revision Register

Table 8-389. Revision Register (REV, Offset 0x50)

Bit	Name	Type	Description	Default Value
[31:24]	-	Rsvd	Reserved	0
[23:0]	kbrevision	RO	FTKBC010 IP revision	0001_0104

[3]M is configurable and ranges from 3 to 15.

8.10.6.13. Feature Register

Table 8-390. Feature Register (FEATURE, Offset 0x54)

Bit	Name	Type	Description	Default Value
[31:16]	-	Rsvd	Reserved	0
[15:8]	KBC_portxno	RO	Scan code output number at X-axis for the keypad	0x8
[7:0]	KBC_portyno	RO	Scan code input number at Y-axis for the keypad	0x10

Vangotech

Chapter 9. Electrical Characteristics

9.1. Absolute Maximum Ratings

Absolute Maximum Ratings are those values beyond which damage to the device may occur.Table 9-1 specifies the absolute maximum ratings.

CAUTION

Stresses beyond those listed under Table 9-1 may affect reliability or cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions beyond those indicated under Section 9.2, *Recommended Operating Conditions*, is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

Table 9-1. Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit
Tstg	Storage Temperature	-40	125	°C
VCC11K_RTC	Core Power Supply for RTC Domain	-0.5	1.5	V
VCC3IO_RTC	I/O Power Supply for RTC Domain	-0.5	4.6	V
VCC3IO_RTCOSC	Power Supply for RTC's Xtal Pad	-0.5	4.6	V
VCCK	Core Power Supply for System Domain	-0.5	1.5	V
VCC3IO	I/O Power Supply for System Domain	-0.5	4.6	V
VCC3IO_OSC	Power Supply for System's Xtal Pad	-0.5	4.6	V
VCC3IO_EFUSE	Power Supply for Efuse's Program	-0.5	4.6	V
VCC3IO_EMMC0	I/O Power Supply for EMMC0	-0.5	4.6	V
VCC3IO_EMMC1	I/O Power Supply for EMMC1	-0.5	4.6	V
VCC11A_DLL_0	Power Supply for DLL0	-0.5	1.5	V
VCC11A_DLL_1	Power Supply for DLL1	-0.5	1.5	V
VCC11A_PLL04	Power Supply for PLL0 and PLL4	-0.5	1.5	V
VCC11A_PLL126	Power Supply for PLL1, PLL2 and PLL6	-0.5	1.5	V
VCC11A_PLL35	Power Supply for PLL3 and PLL5	-0.5	1.5	V
VCC11K_DDR	Core Power Supply for DDR PHY	-0.5	1.5	V
VCC11A_PLL_DDR	Power Supply for DDR PHY's PLL	-0.5	1.5	V
VCC15O_DDR	I/O Power Supply for DDR PHY	-0.5	2.5	V
VCC15O_DDRCK	Power Supply for DRAM's CK Pad	-0.5	2.5	V
VCC33A_HSRT_USB0	Power Supply for USB0	-0.5	4.6	V
VCC33A_HSRT_USB1	Power Supply for USB1	-0.5	4.6	V
VCC33A_HSRT_US	Power Supply for USB2	-0.5	4.6	V

vangotech

B2					
VCC33A_PLL_USB0	Power Supply for USB0's PLL		-0.5	4.6	V
VCC33A_PLL_USB1	Power Supply for USB1's PLL		-0.5	4.6	V
VCC33A_PLL_USB2	Power Supply for USB2's PLL		-0.5	4.6	V
VCC33A_TDC	Power Supply for TDC		-0.3	3.9	V
VCC33A_ADC	Power Supply for ADC		-0.3	3.9	V
VCC33D_ADC	Power Supply for ADC		-0.3	3.9	V
VCC33IO_ADC	Power Supply for ADC		-0.3	3.9	V
V_{ESD}	Electrostatic Discharge	Human Body Model(HBM)(1)	-4000	4000	V
		Charged Device Model(CDM)(2)	-250	250	V
$I_{Latch-up}$	Latch-up I-test performance current-pulse injection on each IO pin(3)		Pass		
	Latch-up over-voltage performance voltage injection on each IO pin(4)		Pass		

(1). Test method: JEDEC JS-001-2014(Class-3A). JEDEC document JEP155 states that 500V HBM allows safe manufacturing with a standard ESD control process.

(2). Test method: JESD22-C101F(Class-C1). JEDEC document JEP157 states that 250V CDM allows safe manufacturing with a standard ESD control process.

(3). Current test performance: Pins stressed per JEDEC JESD78D(Class I, Level A) and passed with I/O pin injection current as defined in JEDEC.

(4). Over voltage performance: Supplies stressed per JEDEC JESD78D(Class I, Level A) and passed voltage injection as defined in JEDEC.

9.2. Recommended Operating Conditions

All V8130L modules are used under the operating conditions contained in Table 9-2.

NOTE

Logic functions and parameter values are not assured out of the range specified in the recommended operating conditions.

Table 9-2. Recommended Operating Conditions

Symbol	Parameter	Min	Typ	Max	Unit
Ta	Ambient Operating Temperature	-40	-	85	°C
VCC11K_RTC	Core Power Supply for RTC Domain	1.045	1.1	1.155	V
VCC3IO_RTC	I/O Power Supply for RTC Domain	3.135	3.3	3.465	V
VCC3IO_RTCOSC	Power Supply for RTC's Xtal Pad	3.135	3.3	3.465	V
VCCK	Core Power Supply for System Domain	0.99	1.1	1.21	V
VCC3IO	I/O Power Supply for System Domain	2.97	3.3	3.63	V

VCC3IO_OSC	Power Supply for System's Xtal Pad	3.135	3.3	3.465	V
VCC3IO_EFUSE	Power Supply for Efuse's Program	3.135	3.3	3.465	V
VCC3IO_EMMC0	I/O Power Supply for EMMC0	3.135	3.3	3.465	V
		1.71	1.8	1.89	
VCC3IO_EMMC1	I/O Power Supply for EMMC1	3.135	3.3	3.465	V
		1.71	1.8	1.89	
VCC11A_DLL_0	Power Supply for DLL0 and PLL7	1.045	1.1	1.155	V
VCC11A_DLL_1	Power Supply for DLL1	1.045	1.1	1.155	V
VCC11A_PLL04	Power Supply for PLL0 and PLL4	1.045	1.1	1.155	V
VCC11A_PLL126	Power Supply for PLL1, PLL2 and PLL6	1.045	1.1	1.155	V
VCC11A_PLL35	Power Supply for PLL3 and PLL5	1.045	1.1	1.155	V
VCC11K_DDR	Core Power Supply for DDR PHY	1.045	1.1	1.155	V
VCC11A_PLL_DDR	Power Supply for DDR PHY's PLL	1.045	1.1	1.155	V
VCC15O_DDR	I/O Power Supply for DDR PHY	1.425	1.5	1.575	V
VCC15O_DDRCK	Power Supply for DRAM's CK Pad	1.425	1.5	1.575	V
VCC33A_HSRT_USB0	Power Supply for USB0	3.135	3.3	3.465	V
VCC33A_HSRT_USB1	Power Supply for USB1	3.135	3.3	3.465	V
VCC33A_HSRT_USB2	Power Supply for USB2	3.135	3.3	3.465	V
VCC33A_PLL_USB0	Power Supply for USB0's PLL	3.135	3.3	3.465	V
VCC33A_PLL_USB1	Power Supply for USB1's PLL	3.135	3.3	3.465	V
VCC33A_PLL_USB2	Power Supply for USB2's PLL	3.135	3.3	3.465	V
VCC33A_TDC	Power Supply for TDC	3.135	3.3	3.465	V
VCC33A_ADC	Power Supply for ADC	3.135	3.3	3.465	V
VCC33D_ADC	Power Supply for ADC	3.135	3.3	3.465	V
VCC33IO_ADC	Power Supply for ADC	3.135	3.3	3.465	V
Tj	Junction Temperature Range	-40	-	125	°C

9.3. DC Electrical Characteristics

Table 9-3 summarizes the DC electrical characteristics of V8130L.

Table 9-3. DC Electrical Characteristics

Parameter		Symbol	Condition	Min	Typ	Max	Unit
Digital GPIO	High-Level Input Voltage	V_{IH}	LVTTL	2.0	-	-	V
	Low-Level Input Voltage	V_{IL}	LVTTL	-	-	0.8	V
	Schmitt-trigger negative-to-threshold voltage	V_{T-}	LVTTL	0.8	1.1		V
	Schmitt-trigger positive-to-threshold voltage	V_{T+}	LVTTL		1.6	2.0	V
	Input Pull-up Resistance	R_{PU}	V_{in}=0V	40	75	190	kΩ
	Input Pull-down Resistance	R_{PD}	V_{in}=VCC3IO	30	75	190	kΩ

	High-Level Input Current	I_{IH}		-	-		uA
	Low-Level Input Current	I_{IL}		-	-		uA
	High-Level Output Voltage	V_{OH}	I_{oh}=4.0mA~16mA	2.4	-	-	V
	Low-Level Output Voltage	V_{OL}	I_{ol}=4.0mA~16mA	-	-	0.4	V
	Tri-State Output Leakage Current	I_{OZ}		-	1.0	10	uA
	Input Capacitance	C_{IN}		-	2.3	-	pF
	Output Capacitance	C_{OUT}		-	-		pF

9.4. SDRAM I/O DC Electrical Characteristics

The SDRAM I/O pads support DDR3,DDR3L,LPDDR2,and DDR2 operational modes. The SDRAM Controller(DRAMC) is designed to be compatible with JEDEC-compliant SDRAMs. The DRAMC supports the following memory types:

- DDR3 SDRAM compliant to JESD79-3E DDR3 JEDEC standard release July, 2010
- LPDDR2 SDRAM compliant to JESD209-2B LPDDR2 JEDEC standard release June, 2009
- DDR3L SDRAM compliant to JESD79-3-1A.01 DDR3L JEDEC standard release May, 2013

Table 9-4. DC Input Logic Level

Characteristics	Symbol	Min	Typ	Max	Unit
DC input logic high	$V_{IH(DC)}$	VREF + 100	-	-	mV
DC input logic low	$V_{IL(DC)}$	-	-	VREF – 100	mV
Input reference voltage	Vref	0.49 * VDDQ	-	0.51 * VDDQ	V
Input termination resistance(ODT) to V_{DDQ}/2	R_{TT}	60	120	Open	Ω

Table 9-5. Output DC Current Drive

Characteristics	Symbol	Min	Max	Unit
DC output high voltage	V_{OH}	0.9 * VDDQ	-	V
DC output low voltage	V_{OL}	-	0.1 * VDDQ	V

9.5. Oscillator Electrical Characteristics

V8130L contains two external input clocks: X24MIN and X32KIN, two output clocks: X24MOUT and X32KOUT.

The 24MHz frequency is used to generate the main source clock for PLL and the main digital blocks, the clock is provided through X24MIN. **Table 9-6** lists the 24MHz crystal specifications.

Table 9-6. 24MHz Crystal Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
1/(t_{CPMAIN})	Crystal Oscillator Frequency Range	–	24.000	–	MHz
	Frequency Tolerance at 25 °C	-20	–	+20	ppm
	Oscillation Mode	Fundamental			–
	Maximum Change Over Temperature Range	-20	–	+20	ppm
C_L	Equivalent Load Capacitance	12	18	22	pF
C_O	Shunt Capacitance	–	3	–	pF

The 32.768kHz frequency is used for low frequency operation. It supplies the wake-up domain for operation in lowest power mode. The clock is provided through X32KIN. **Table 9-7** lists the 32.768kHz crystal specifications.

Table 9-7. 32.768kHz Crystal Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
1/(t_{CPMAIN})	Crystal Oscillator Frequency Range	–	32.768	–	kHz
	Frequency Tolerance at 25 °C	-20	–	+20	ppm
	Oscillation Mode	Fundamental			–
	Maximum Change Over Temperature Range	-20	–	+20	ppm
C_L	Equivalent Load Capacitance	–	12.5	–	pF
C_O	Shunt Capacitance	–	1.1	–	pF

9.6. External Peripherals Electrical Characteristics

9.6.1. LCD AC Electrical Characteristics

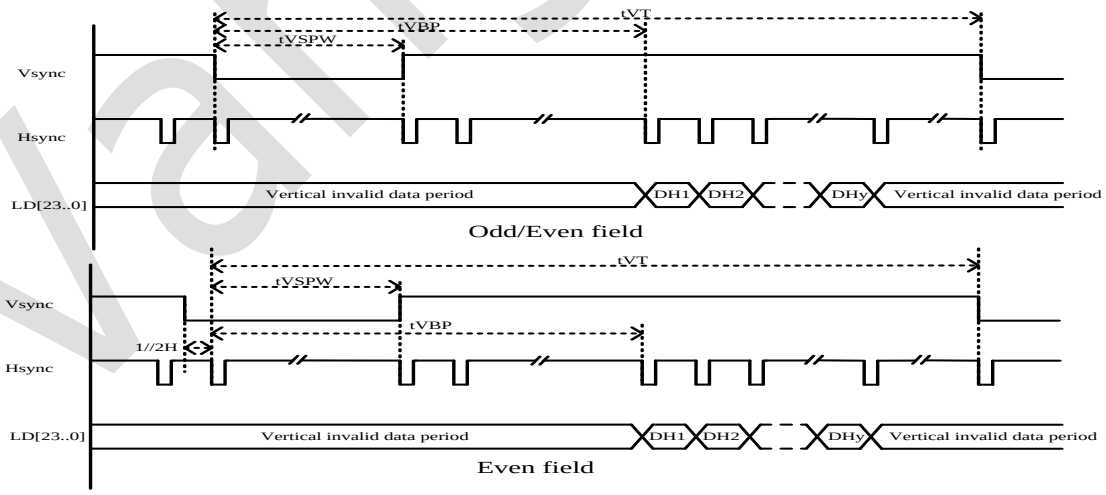


Figure 9-1. HV_IF Interface Vertical Timing

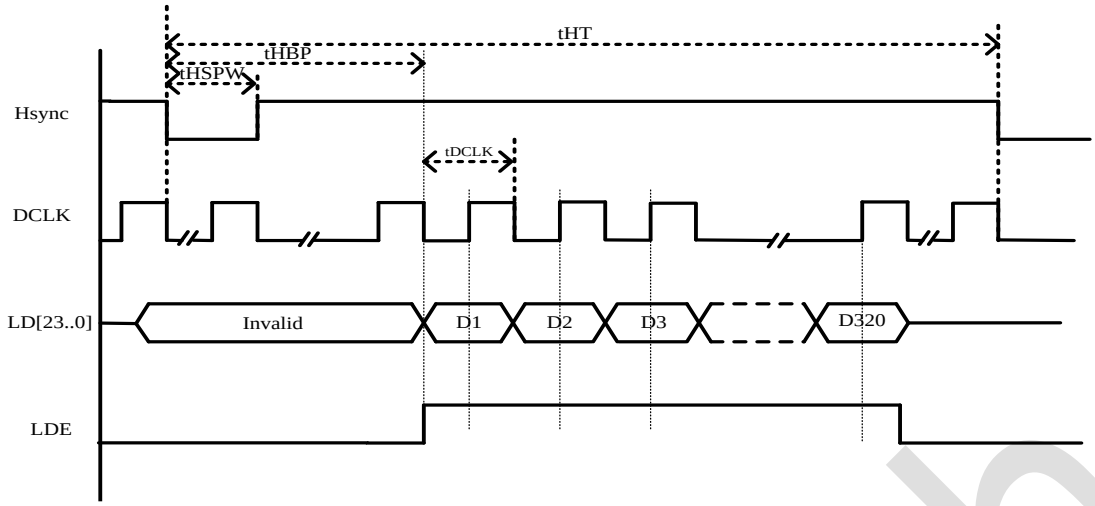


Figure 9-2. HV_IF Interface Parallel Mode Horizontal Timing

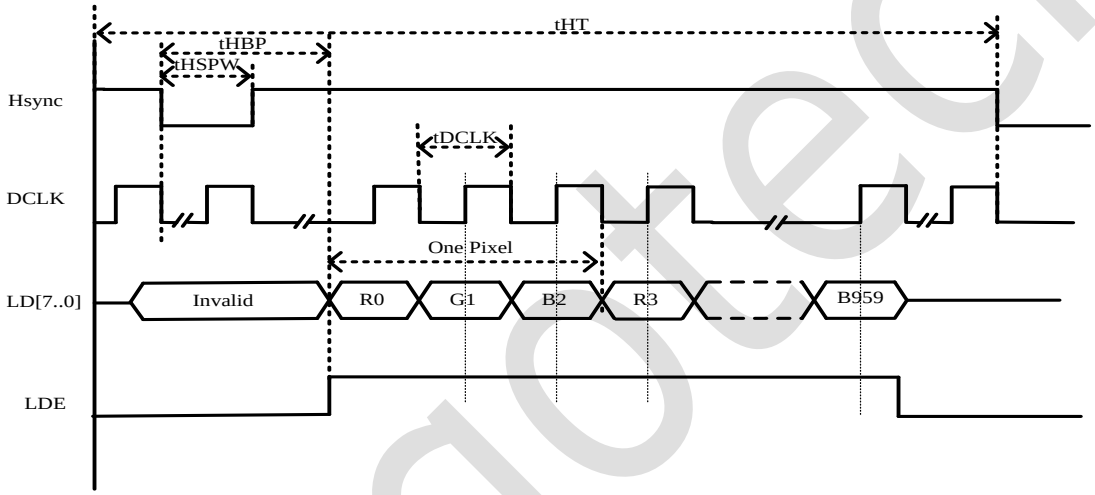


Figure 9-3. HV_IF Interface Serial Mode Horizontal Timing

Table 9-8. LCD HV_IF Interface Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
DCLK cycle time	tDCLK	5	-	-	ns
HSYNC period time	tHT	-	HT+1	-	tDCLK
HSYNC width	tHSPW	-	HSPW+1	-	tDCLK
HSYNC back porch	tHBP	-	HBP+1	-	tDCLK
VSYNC period time	tVT	-	VT/2	-	tHT
VSYNC width	tVSPW	-	VSPW+1	-	tHT
VSYNC back porch	tVBP	-	VBP+1	-	tHT

(1). Vsync: Vertical sync, indicates one new frame.

(2). Hsync: Horizontal sync, indicate one new scan line.

(3). DCLK: Dot clock, pixel data are sync by this clock.

(4). LDE: LCD data enable.

(5). LD[23..0]: 24Bit RGB/YUV output from input FIFO for panel.

vangotech

9.6.2. SPI AC Electrical Characteristics

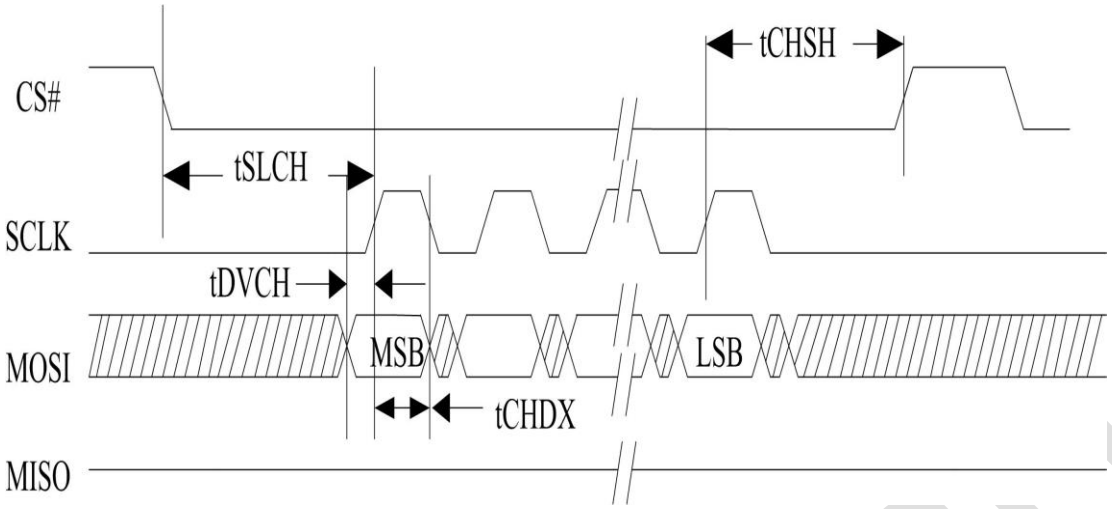


Figure 9-4. SPI MOSI Timing

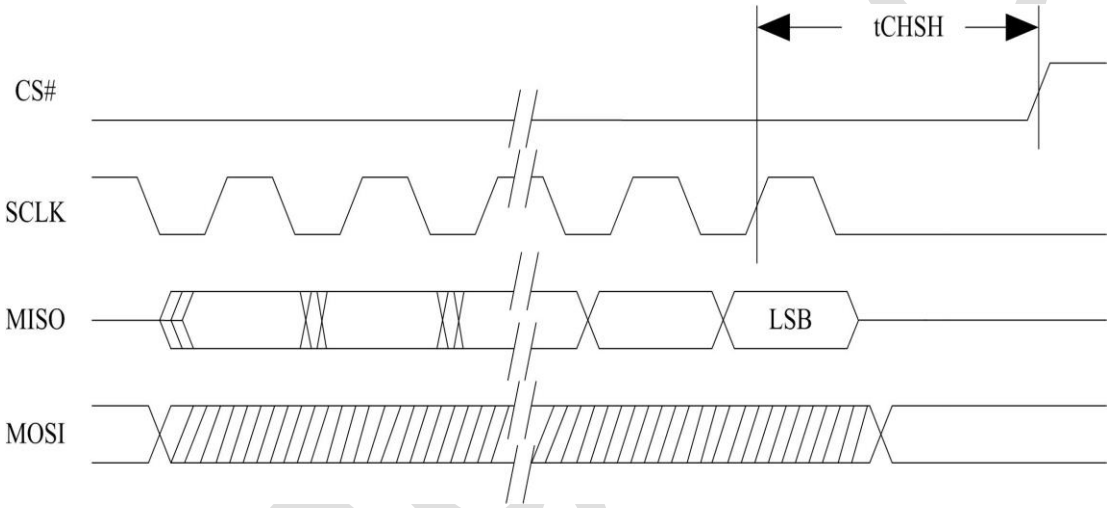


Figure 9-5. SPI MISO Timing

Table 9-9. SPI Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
CS# Active Setup Time	tSLCH	-	2T	-	ns
CS# Active Hold Time	tCHSH	-	2T(1)	-	ns
Data In Setup Time	tDVCH	-	T/2-3	-	ns
Data In Hold Time	tCHDX	-	T/2-3	-	ns

NOTE

(1):T is the cycle of clock.

9.6.3. UART AC Electrical Characteristics

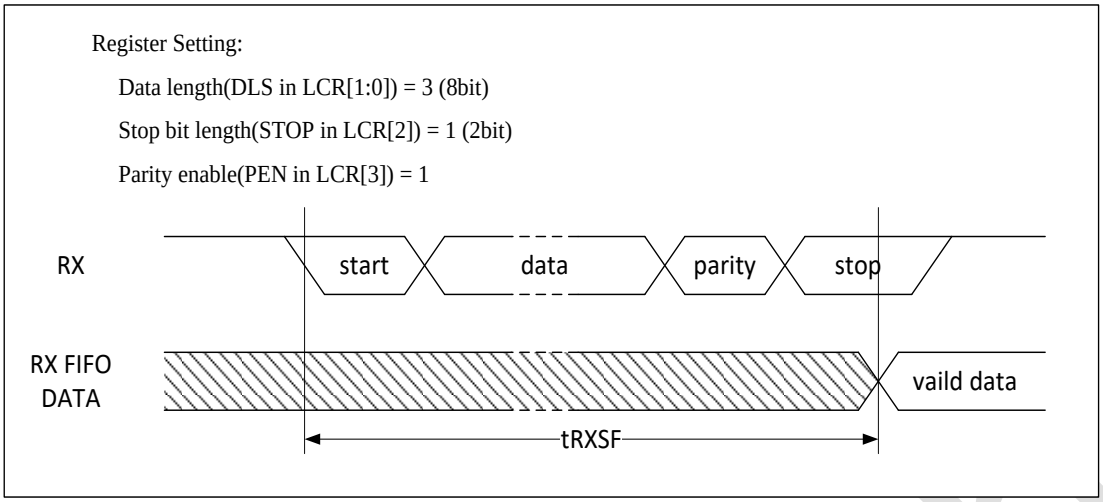


Figure 9-6. UART RX Timing

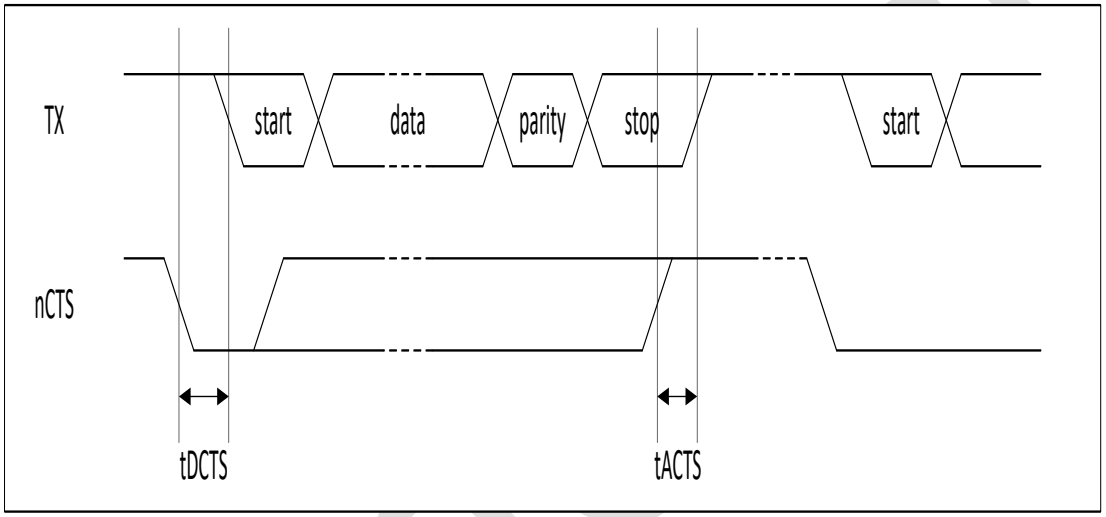


Figure 9-7. UART nCTS Timing

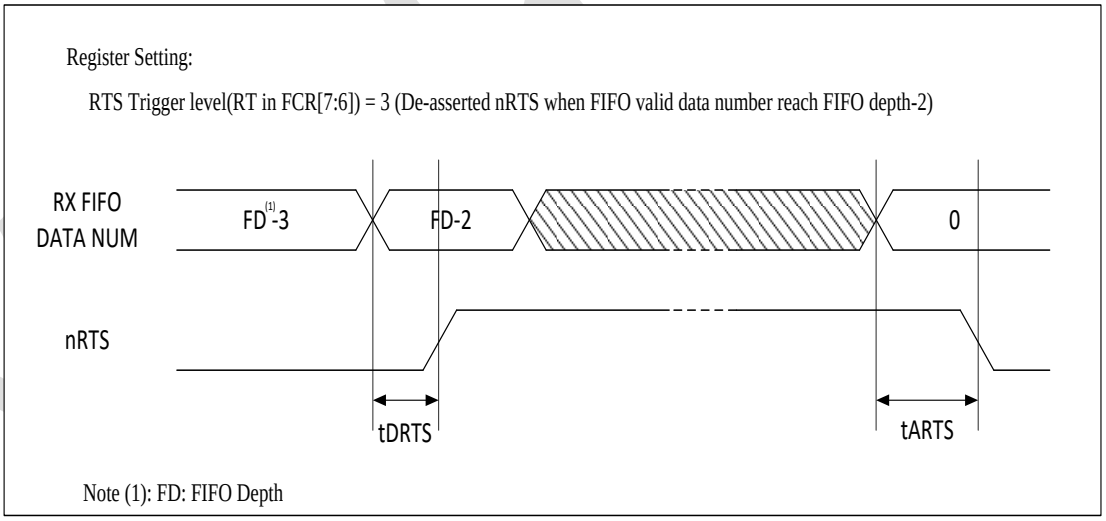


Figure 9-8. UART nRTS Timing

Table 9-10. UART Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
RX start to RX FIFO	tRXSF	10.5× BRP(1)	-	11× BRP(1)	ns
Delay time of de-asserted nCTS to TX start	tDCTS	-	-	BRP(1)	ns

Step time of asserted nCTS to stop next transmission	tACTS	BRP(1)/4	-	-	ns
Delay time of de-asserted nRTS	tDRTS	-	-	BRP(1)	ns
Delay time of asserted nRTS	tARTS	-	-	BRP(1)	ns

NOTE

(1): BRP(Baud-Rate Period).

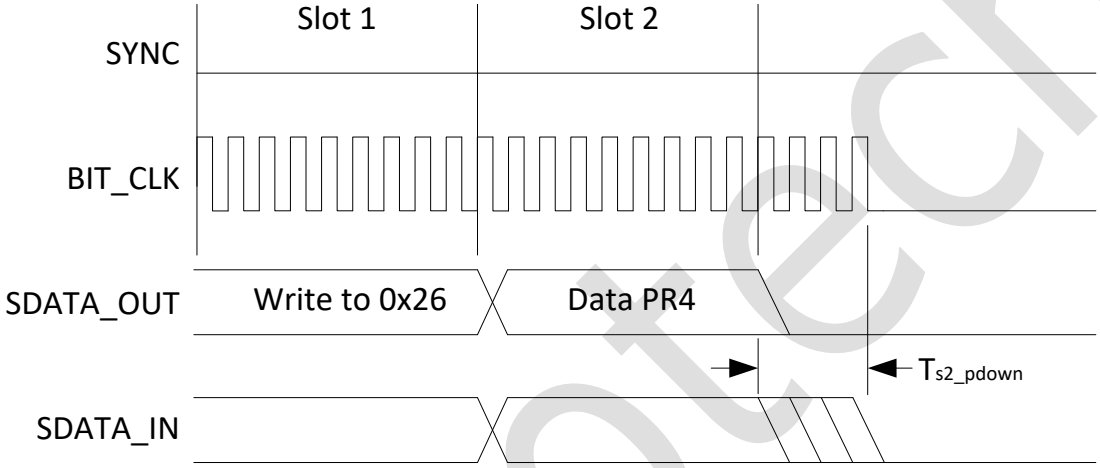
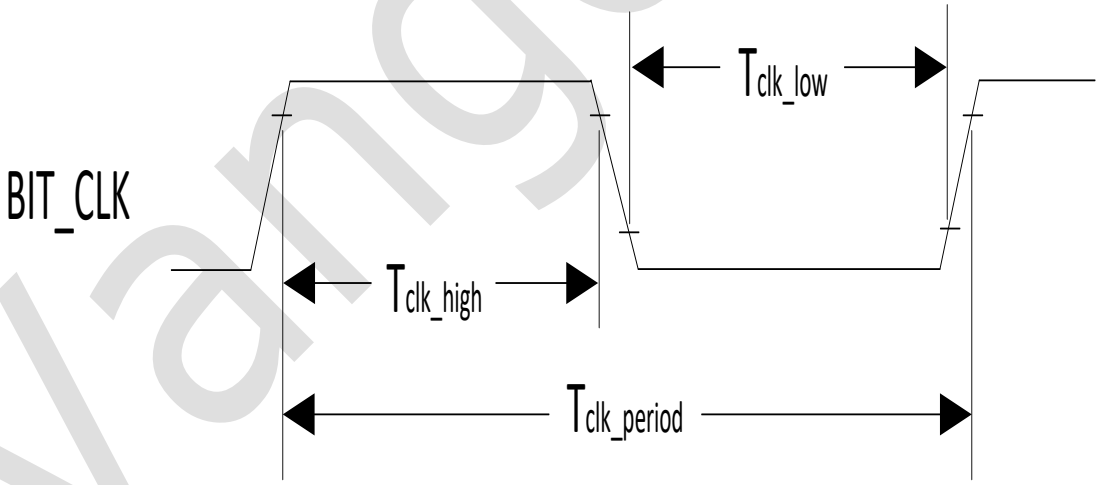


Figure 9-9. AC-link Low Power Mode Timing



Vangotech

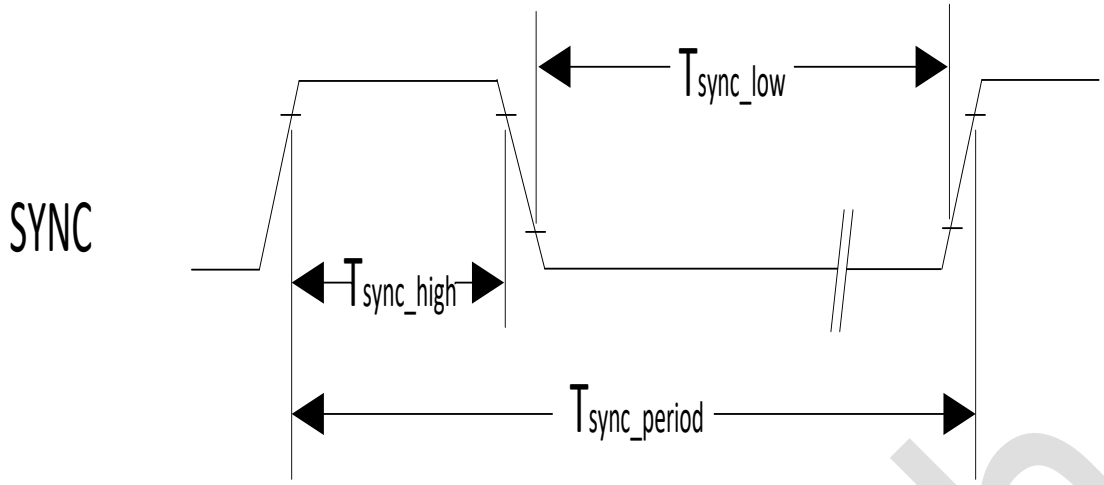


Figure 9-10. BIT_CLK and SYNC Timing

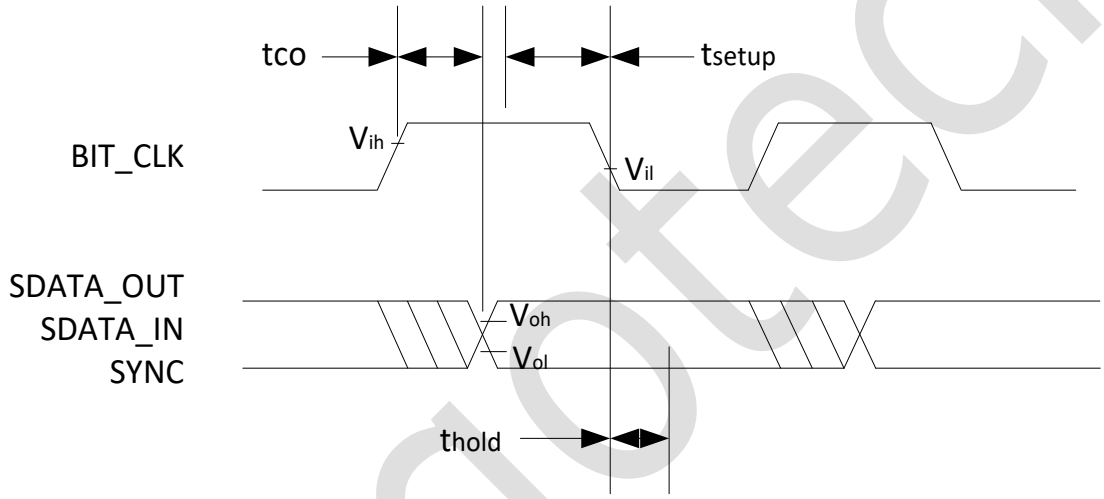


Figure 9-11. AC-link Data Transmission Output and Input Timing

Figure 9-12. Signal Rise and Fall Timing

Table 9-11. AC97 Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
RESET# active low pulse width	T_{rst_low}	1.0	-	-	us
RESET# inactive to SDATA_IN Or BIT_CLK active delay	$T_{tri2actv}$	-	-	25	ns

Vangotech

RESET# inactive to BIT_CLK Startup delay	$T_{rst2clk}$	162.8	-	-	ns
SYNC active high pulse width	T_{sync_high}	1.0	-	-	us
SYNC inactive to BIT_CLK startup delay	$T_{sync2clk}$	162.8	-	-	ns
End of Slot 2 to BIT_CLK, SDATA_IN low	T_{s2_pdown}	-	-	1.0	us
BIT_CLK frequency		-	12.288	-	MHz
BIT_CLK period	Tclk_period	-	81.4	-	ns
BIT_CLK output jitter		-	-	750	ps
BIT_CLK high pulse width	Tclk_high	36	40.7	45	ns
BIT_CLK low pulse width	Tclk_low	36	40.7	45	ns
SYNC frequency		-	48.0	-	kHz
SYNC period	Tsync_period	-	20.8	-	us
SYNC high pulse width	Tsync_high	-	1.3	-	us
SYNC low pulse width	Tsync_low	-	19.5	-	us
Output Valid Delay from rising edge of BIT_CLK	tco	-	-	15	ns
Input Setup to falling edge of BIT_CLK	tsetup	10	-	-	ns
Input Hold from falling edge of BIT_CLK	thold	10	-	-	ns
BIT_CLK combined rise or fall plus flight time (Primary Codec to Controller or Secondary)		-	-	7	ns
SDATA combined rise or fall plus flight time (Output to Input)		-	-	7	ns
BIT_CLK rise time	Triseclk	-	-	6	ns
BIT_CLK fall time	Tfallclk	-	-	6	ns
SYNC rise time	Trisesync	-	-	6	ns
SYNC fall time	Tfallsync	-	-	6	ns
SDATA_IN rise time	Trisedin	-	-	6	ns
SDATA_IN fall time	Tfalldin	-	-	6	ns
SDATA_OUT rise time	Trisedout	-	-	6	ns
SDATA_OUT fall time	Tfalldout	-	-	6	ns

(1). Worst case duty cycle restricted to 45/55.

(2). Combined rise or fall plus flight times are provided for worst case scenario modeling purpose.

(3). BIT_CLK rise/fall times with an external load of 75 pF.

(4). SYNC and SDATA_OUT rise/fall times with a external load of 75 pF.

Vangotech

(5). SDATA_IN rise/fall times with an external load of 60 pF.

(6). Rise is from 10% to 90% of Vdd (Vol to Voh).

(7). Fall is from 90% to 10% of Vdd (Voh to Vol).

9.6.4. SCR AC Electrical Characteristics

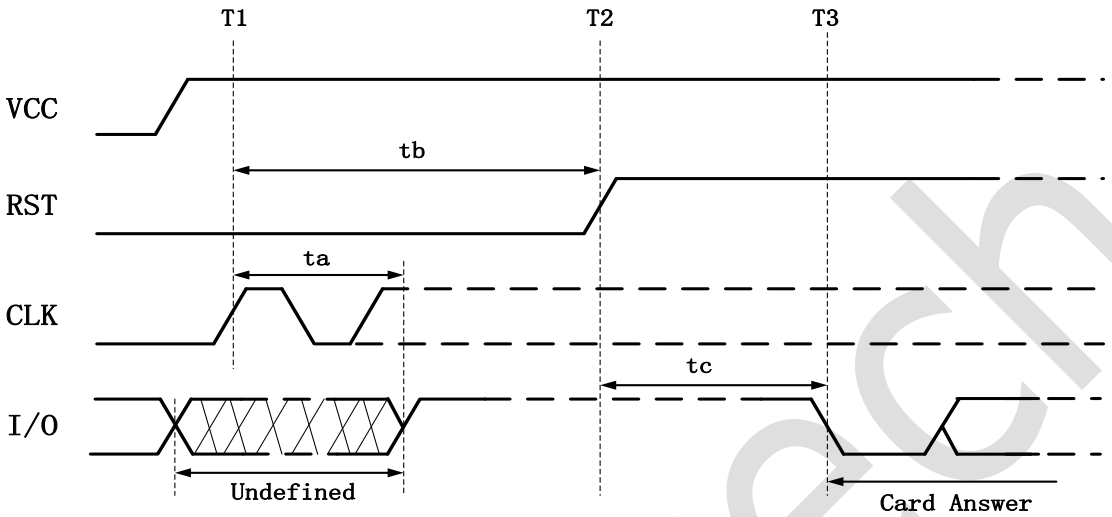


Figure 9-13. SCR Activation and Cold Reset Timing

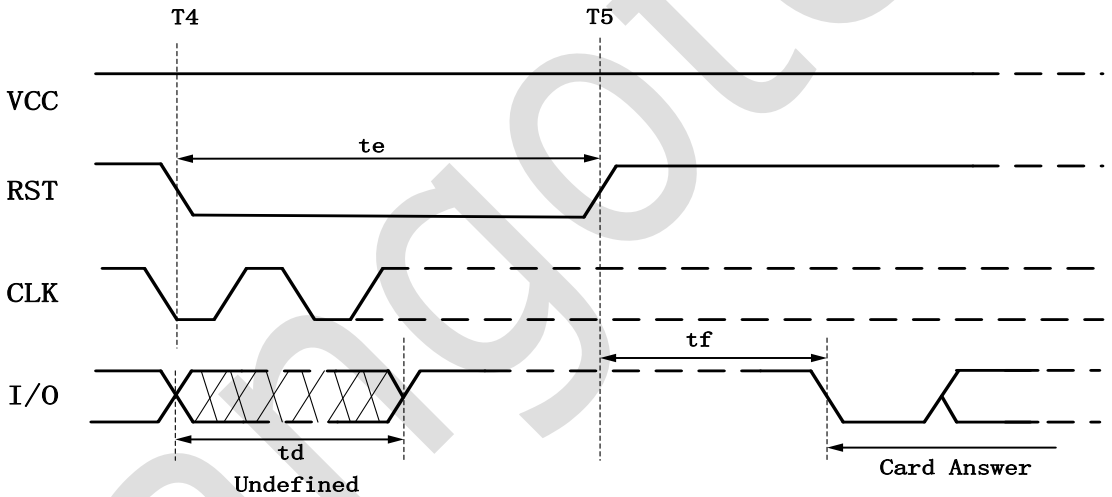


Figure 9-14. SCR Warm Reset Timing

Table 9-12. SCR Timing Constants

Symbol	Min	Typ	Max	Unit
ta	-	-	200/f	us
tb	400/f	-	-	us
tc	400/f	-	40000/f	us
td	-	-	200/f	us
te	400/f	-	-	us
tf	400/f	-	40000/f	us

(1). Activation: Before time T1

(2). Cold Reset: After time T1

(3). T1: The clock signal is applied to CLK at time T1.

(4). T2: The RST is put to state H.

Vangotech

(5). T3: The card begin answer at time T3

(6). ta: The card shall set I/O to state H within 200 clock cycles (delay ta) after the clock signal is applied to CLK (at time T1+ta).

(7). tb: The cold reset results from maintaining RST at state L for at least 400 clock cycles (delay tb) after the clock signal is applied to CLK (at time T1+tb).

(8). tc: The answer on I/O shall begin between 400 and 40000 clock cycles (delay tc) after the rising edge of the signal on RST (at time T2+tc).

(9). td: The card shall set I/O to state H within 200 clock cycles (delay td) after state L is applied to RST (at time T4+td).

(10). te: The controller initiates a warm reset (at time T4) by putting RST to state L for at least 400 clock cycles (delay te) while VCC remains powered and CLK provided with a suitable and stabled clock signal.

(11). tf: The card answer on I/O shall begin between 400 and 40000 clock cycles (delay tf) after the rising edge of the signal on RST (at time T5+tf).

(12). f is the frequency of clock.

9.7. Power-up and Power-down Sequence

Figure 9-15 shows the power-up sequence for V8130L. It's recommended for 1.1V core power(VCCK) to be stable before 3.3V PAD power (VCC3IO) is. After 3.3V is stable, X_RESET_N should be low level for at least 1ms. 1.8V/3.3V power for eMMC IO, VCC3IO_EMMC0 and VCC3IO_EMMC1, needs to be stable within 6.76ms after X_RESET de-asserted. 1.5V power for DDR IO, VCC15O_DDR and VCC15O_DDRCK, needs to be stable within 10.16ms after X_RESET_N de-asserted.

All power supplies can ramp down except VCC_RTC. The ramping down rate is determined by the load on the power source.

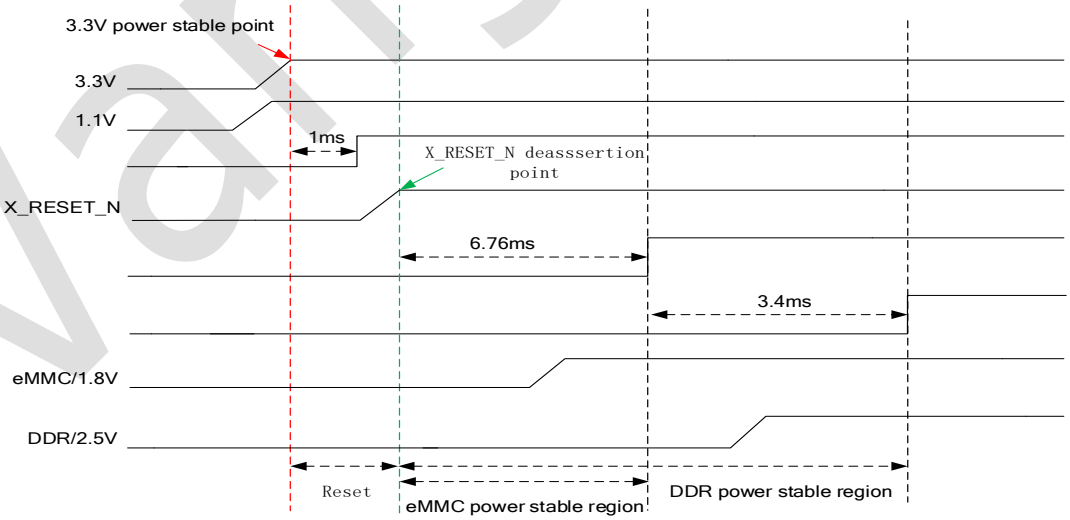


Figure 9-15. V8130L Power Up Sequence

Chapter 10. Package Thermal Characteristics

For reliability and operability concerns, the absolute maximum junction temperature of V8130L has to be below 125°C. The testing PCB is based on 4 layers. The following thermal resistance characteristics in Table 10-1 is based on JEDEC JESD51 standard, because the system design and temperature could be different with JEDEC JESD51, the simulating resulting data is a reference only, please prevail in the actual application condition test.

Table 10-1. V8130L Thermal Resistance Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
θ_{JA}	Junction-to-Ambient Thermal Resistance	-	17	-	°C/W
θ_{JB}（**TBD**）	Junction-to-Board Thermal Resistance	-	5.648	-	°C/W
θ_{JC}	Junction-to-Case Thermal Resistance	-	4.98	-	°C/W

(1). These values are based on a JEDEC-defined 2S2P system and will change based on environment as well as application.

(2). °C/W : degrees Celsius per watt.

Vangotech

Chapter 11. Pin Assignment

11.1. Pin Map

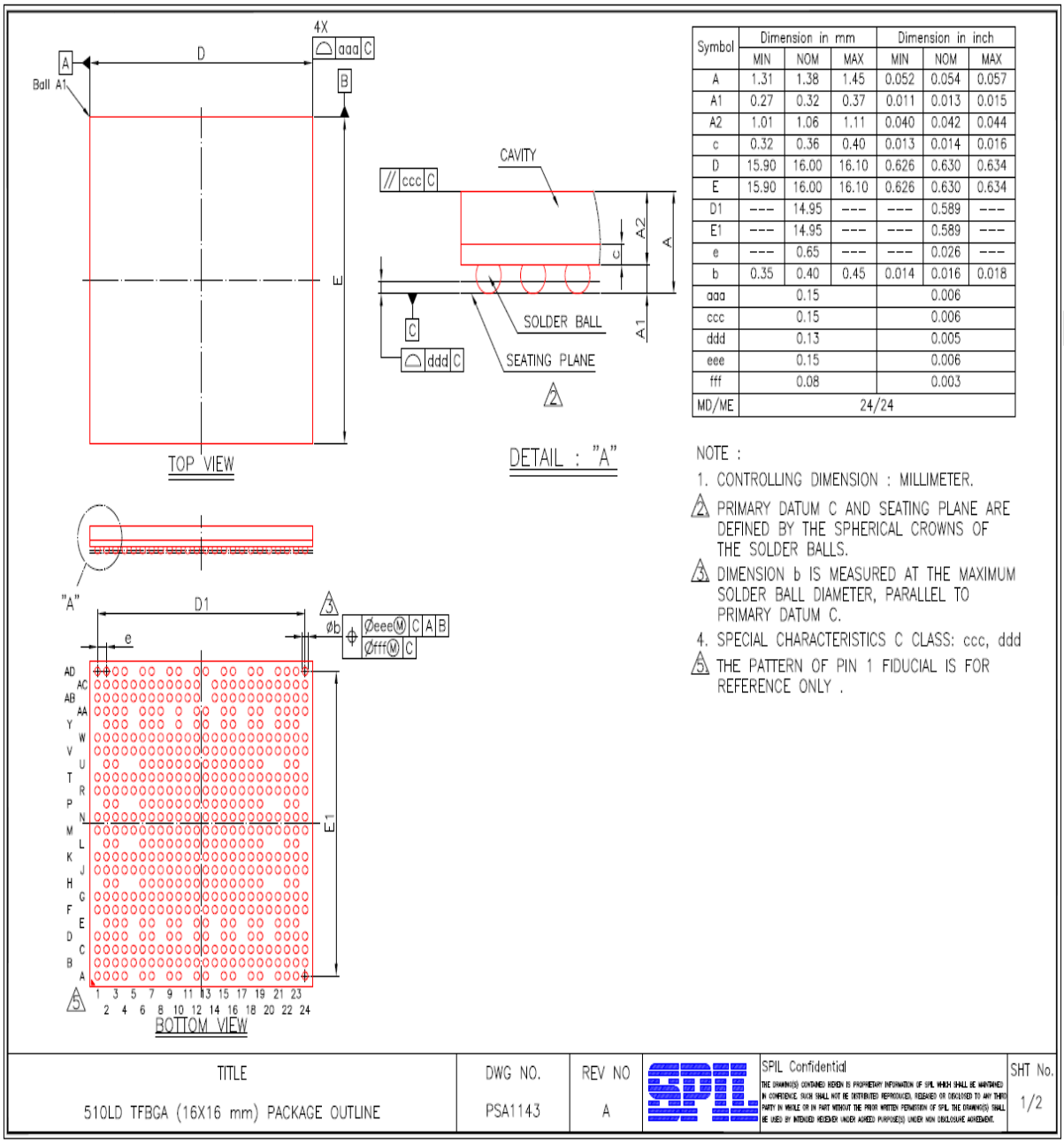
For V8130L, TFBGA 510 balls, 16 mm x 16 mm, 0.65 mm pitch package is offered. The pin maps are illustrated in Figure 11-1 for this package.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	GND	X_DDR_RDRVUP	X_DDR_DQ_1[6]	X_DDR_DQ_1[4]		X_DDR_DM_1	X_DDR_DQ_1[7]		X_DDR_DQ_3[4]	X_DDR_DQS_3		X_DDR_DQ_3[1]	X_DDR_DQ_3[5]		X_DDR_DM_4	X_DDR_DQ_4[4]		X_MII_COL	X_MII_MDIO		X_MII_RXD[0]	X_MII_RXD[1]	X_RMII_TXD[1]	GND
B	X_DDR_RASN	X_DDR_RDRVDN	X_DDR_DQ_1[2]	X_DDR_DQ_1[0]	X_DDR_DQS_1	X_DDR_DQ_1[1]	X_DDR_DQ_1[5]	X_DDR_DQ_3[2]	X_DDR_DQ_3[6]	X_DDR_DQSB_3	X_DDR_DQ_3[0]	X_DDR_DM_3	X_DDR_DQ_3[3]	X_DDR_DQ_4[5]	X_DDR_DQ_4[1]	X_DDR_DQ_4[7]	X_DDR_DQ_4[6]	GND	X_MII_MDC	X_MII_RX_DV	X_MII_RX_CK	X_MII_RXD[2]	X_RMII_RXD[1]	X_RMII_TXD[0]
C	X_DDR_CKE[0]	X_DDR_CASN	X_DDR_VREF_D0	X_DDR_VREF_D1	X_DDR_DQSB_1	X_DDR_DQSB_0	X_DDR_DQ_1[3]	GND	X_DDR_VREF_D2	X_DDR_VREF_D3	GND	X_DDR_DQSB_2	X_DDR_DQ_3[7]	X_DDR_DQ_4[3]	GND	X_DDR_DQ_4[0]	X_DDR_DQ_4[2]	GND	X_MII_TXD[0]	X_MII_RX_ER	X_MII_TXD[3]	X_RMII_RX_ER	X_RMII_RXD[0]	X_RMII_TX_EN
D	X_DDR_WEN	X_DDR_ADDR[10]	GND	X_DDR_DQ_0[7]		GND	X_DDR_DQS_0		GND	X_DDR_DQ_2[7]		GND	X_DDR_DQS_2		X_DDR_DQS_4	X_DDR_VREF_D4		X_MII_CRS	X_MII_TX_EN		X_UART0_RX	X_RMII_REF_CLK	X_RMII_MDIO	X_RMII_RX_CRS_DV
E		X_DDR_CSN[0]	X_DDR_DQ_0[5]	X_DDR_DQ_0[0]		X_DDR_DQ_0[4]	X_DDR_DQ_0[6]		X_DDR_DQ_2[5]	X_DDR_DQ_2[0]		X_DDR_DQ_2[4]	X_DDR_DQ_2[6]		X_DDR_DQSB_4	GND		X_MII_TX_ER	X_MII_TXD[1]		X_I2C2_CLK	X_UART0_TX	X_RMII_MDC	
F	X_DDR_CKB	X_DDR_CK	X_DDR_DM_0	X_DDR_DQ_0[3]	GND	X_DDR_DQ_0[1]	X_DDR_DQ_0[2]	GND	X_DDR_DM_2	X_DDR_DQ_2[3]	GND	X_DDR_DQ_2[1]	X_DDR_DQ_2[2]	GND	GND	X_MII_RXD[3]	GND	X_MII_TX_CK	X_MII_TXD[2]	GND	X_I2C2_DATA	X_RGMII_RXCTL	X_RGMII_GTX_CK	X_RGMII_MDIO
G	X_DDR_ADDR[15]	X_DDR_CSN[1]	X_DDR_ADDR[0]	X_DDR_ADDR[2]	X_DDR_CKE[1]	GND	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC3IO	X_HSPI_DO	X_RGMII_RXD[0]	X_RGMII_RX_CK	X_RGMII_MDC	X_RGMII_RXD[2]	X_RGMII_RXD[1]
H		X_DDR_ADDR[3]	X_DDR_BA[1]			X_DDR_ODT[1]	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	GND	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	GND	VCC15O_DDR	VCC15O_DDR	VCC15O_DDR	VCC3IO	X_HSPI_CS			X_RGMII_TXD[0]	X_RGMII_TXD[1]	
J	X_DDR_ADDR[11]	X_DDR_ADDR[1]	X_DDR_ADDR[13]	GND	X_DDR_ADDR[9]	X_DDR_ODT[0]	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	VCCK	X_SPI1_DO	X_SPI1_CS	X_RGMII_RXD[3]	X_RGMII_TXCTL	X_RGMII_TXD[2]	X_RGMII_TXD[3]
K	X_DDR_ADDR[4]	X_DDR_ADDR[14]	X_DDR_BA[2]	X_DDR_ADDR[5]	X_DDR_RESETN	X_DDR_ADDR[7]	GND	VCCK	GND	VCCK	GND	VCCK	GND	VCCK	VCCK	GND	GND	VCCK	X_SPI6_DO	X_SPI6_CS	GND	X_SPI5_DO	X_HSPI_DI	X_HSPI_CLK
L		X_DDR_ADDR[6]	X_DDR_ADDR[8]			X_DDR_BA[0]	VCC15O_DDRCK	VCCK	GND	VCCK	GND	VCCK	GND	VCCK	GND	GND	GND	VCC3IO	X_SPI4_DO			X_SPI5_CS	X_SPI1_CLK	
M	X_RTC_XTAL_IN	GND	GND	VCC3IO_RTCOSC	GND	X_DDR_ADDR[12]	GND11A_PLL_DDR	VCC11A_PLL_DDR	GND	GND	GND	GND	GND	GND	GND	GND	GND	VCC3IO	X_JTAG_SWDITMS	X_SPI4_CS	X_I2S0_RXD	X_I2S_MCLK	X_SPI6_DI	X_SPI1_DI
N	X_RTC_XTAL_IO	X_RESET_N	X_OM	X_SPI_DCX1	VCC3IO_RTC	GND	VCC11A_PLL04	GND11A_PLL04	GND	GND	GND	GND	GND	GND	GND	GND	GND	VCCK	X_SPI3_DO	X_JTAG_TDO	GND	X_SPI0_CLK	X_SPI5_CLK	X_SPI6_CLK
P		X_SPI_DCX2	X_UART9_TX			VCC11K_RTC	GND33A_HSRT_USB2	GND33A_PLL_USB2	GND	GND	GND	GND	GND	GND	GND	GND	GND	VCCK	X_SPI3_CS			X_SPI0_DO	X_SPI4_CLK	
R	X_UART5_RX_H	X_UART6_TX	X_UART9_RX	X_PSW_DDRCK	X_UART6_RX	X_UART5_TX	VCC33A_HSRT_USB2	VCC33A_PLL_USB2	VCCK	GND	GND	GND	GND	GND	GND	GND	GND	VCC3IO	X_UART3_NRTS	X_I2C3_CLK	X_UART4_TX	X_SPI0_CS	X_SPI0_DI	X_SPI5_DI
T	X_UART7_TX	X_UART7_RX	X_UART5_RX	X_UART8_RX	X_UART8_RX_H	X_PSW_0	GND33A_HSRT_USB1	GND33A_PLL_USB1	VCCK	VCCK	VCCK	VCC3IO	VCC3IO	VCCK	VCCK	VCC3IO	VCC3IO	VCC3IO	X_SPI_HOLD_N	X_UART3_TX	X_I2C0_CLK	X_UART3_RX	X_JTAG_TDI	X_SPI4_DI
U		GND	X_UART8_TX			GND	VCC33A_HSRT_USB1	VCC33A_PLL_USB1	GND11A_PLL126	GND	GND11A_PLL35	GND33A_ADC	VCC33IO_ADC	GND33A_HSRT_USB0	VCC33A_PLL_USB0	GND	VCC11A_DLL_1	GND11A_DLL_1	X_SPI_CLK			X_JTAG_SWCLKTCK	X_SPI3_CLK	
V	X_USB2_DP	X_USB2_DM	GND	GND	X_I2C1_DATA	X_I2C1_CLK	VCC3IO_OSC	GND	VCC11A_PLL126	GND	VCC11A_PLL35	VCC33A_ADC	GND33IO_ADC	GND33A_PLL_USB0	VCC33A_HSRT_USB0	GND	VCC11A_DLL_0	GND11A_DLL_0	X_USB0_VBUS	X_UART2_NCTS	GND	X_UART2_NRTS	X_SPI3_DI	X_JTAG_TRST_N
W	X_USB1_DP	X_USB1_DM	GND	X_SPI2_CS	X_LC_DATA[18]	X_USB2_VBUS	X_USB1_VBUS	X_USB2_DRVBUS	GND	X_USB1_DRVBUS	GND	GND33D_ADC	VCC11A_TDC	GND	GND	X_EMMC1_D2	X_EMMC1_D0	X_EMMC1_CLK	GND	X_SPI_DI	X_UART1_RX	X_UART2_RX	X_UART3_NCTS	X_I2C0_DATA
Y		GND	X_SPI2_DO	X_LC_DATA[2]		X_LC_DATA[19]	X_LC_PCLK	X_LC_HS		GND		VCC33D_ADC	GND33A_TDC		GND	X_EMMC1_D4		X_EMMC1_D1	X_EMMC1_CMD		X_SPI7_DO	X_SPI_DO	X_SPI_WP_N	
AA	X_SPI2_CLK	X_SPI2_DI	X_LC_DATA[0]	X_LC_DATA[3]		X_LC_DATA[20]	X_LC_DATA[23]	X_LC_VS		GND33A_ADC		GND33A_ADC	VCC3IO_EFUSE		VCC3IO_EMMC1	X_EMMC1_D6		X_EMMC1_D5	X_EMMC1_D3		X_SPI7_CS	X_SPI_CS_N	X_I2S0_TXD	X_I2S0_FS
AB	X_LC_DATA[1]	X_LC_DATA[5]	X_LC_DATA[4]	X_LC_DATA[9]	X_LC_DATA[12]	X_LC_DATA[22]	X_LC_DATA[16]	X_LC_DE	GND33A_ADC	X_ADC_XAIN[2]	X_ADC_XVRT	X_ADC_XAIN[6]		X_USB0_DM	GND33A_HSRT_USB0	X_EMMC1_RSTN	X_EMMC0_D1	X_EMMC0_CLK	VCC3IO_EMMC0	X_EMMC0_RSTN	GND	X_USB0_DRVBUS	GND	X_I2S0_SCLK
AC	X_SYS_XTAL_IN	X_SYS_XTAL_IO	X_LC_DATA[7]	X_LC_DATA[11]	X_LC_DATA[13]	X_LC_DATA[14]	X_LC_DATA[21]	GND	GND33A_ADC	X_ADC_XAIN[1]	X_ADC_XAIN[3]	X_ADC_XAIN[4]		X_USB0_DP	GND33A_HSRT_USB0	X_EMMC0_D5	X_EMMC0_D7	X_EMMC1_D7	X_EMMC0_D6	X_EMMC0_D2	GND	X_UART1_TX	X_UART4_RX	X_I2C3_DATA
AD	GND	X_LC_DATA[6]	X_LC_DATA[8]	X_LC_DATA[10]		X_LC_DATA[15]	X_LC_DATA[17]		X_ADC_XAIN[0]	X_ADC_XVRB		X_ADC_XAIN[5]	X_ADC_XAIN[7]		X_EMMC0_CMD	X_EMMC0_D3		X_EMMC0_D4	X_EMMC0_D0		X_SPI7_DI	X_SPI7_CLK	X_UART2_TX	GND

Figure 11-1. V8130L Pin Map

11.2. Package Dimension

Figure 11-2 shows the top, bottom, and side views of V8130L package dimension.



Symbol	Dimension in mm			Dimension in inch		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.31	1.38	1.45	0.052	0.054	0.057
A1	0.27	0.32	0.37	0.011	0.013	0.015
A2	1.01	1.06	1.11	0.040	0.042	0.044
c	0.32	0.36	0.40	0.013	0.014	0.016
D	15.90	16.00	16.10	0.626	0.630	0.634
E	15.90	16.00	16.10	0.626	0.630	0.634
D1	---	14.95	---	---	0.589	---
E1	---	14.95	---	---	0.589	---
e	---	0.65	---	---	0.026	---
b	0.35	0.40	0.45	0.014	0.016	0.018
aaa	0.15			0.006		
ccc	0.15			0.006		
ddd	0.13			0.005		
eee	0.15			0.006		
fff	0.08			0.003		
MD/ME	24/24					

Figure 11-2. SCM701 Package Dimension

Chapter 12. Carrier, Storage and Baking Information

12.1. Carrier

12.1.1. Matrix Tray Information

Table 12-1 shows the V8130L matrix tray carrier information.

Table 12-1. Matrix Tray Carrier Information

Item	Color	Size	Note
Tray	Black	315mm x 136mm x 7.62mm	84 Qty/Tray
Aluminum foil bags	Silvery white	540mm x 300mm x 0.14mm	Surface impedance:$10^9\Omega$ Vacuum packing Including HIC and desiccant Printing: RoHS symbol
Pearl cotton cushion(Vacuum bag)	White	12mm x 680mm x 185mm	
Pearl cotton cushion (The Gap between vacuum bag and inner box)	White	Left-Right:12mm x 180mm x 85mm Front-Back:12mm x 350mm x 70mm	
Inner Box	White	396mm x 196mm x 96mm	Printing: RoHS symbol 10 Tray/Inner box
Carton	White	420mm x 410mm x 320mm	6 Inner box/Carton

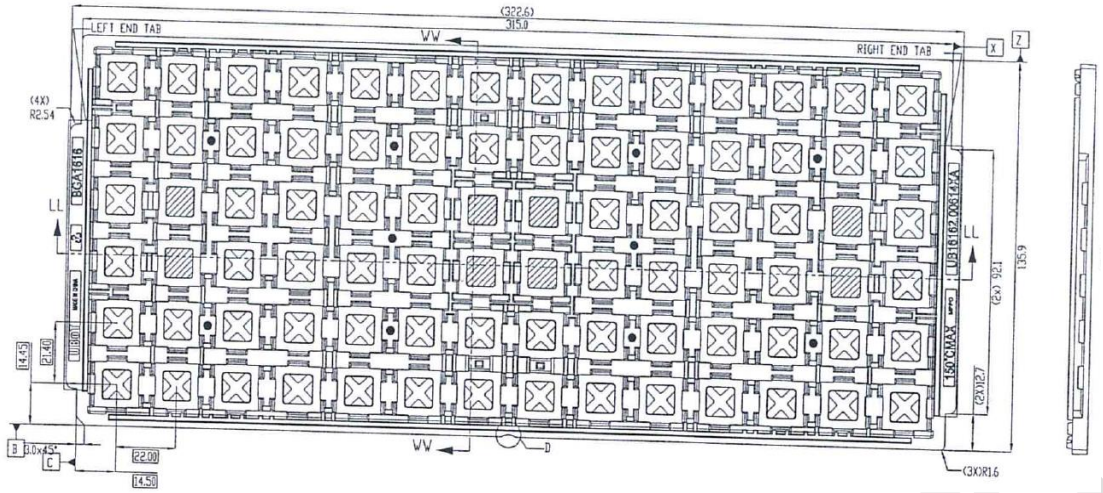
Table 12-2 shows the V8130L packing quantity.

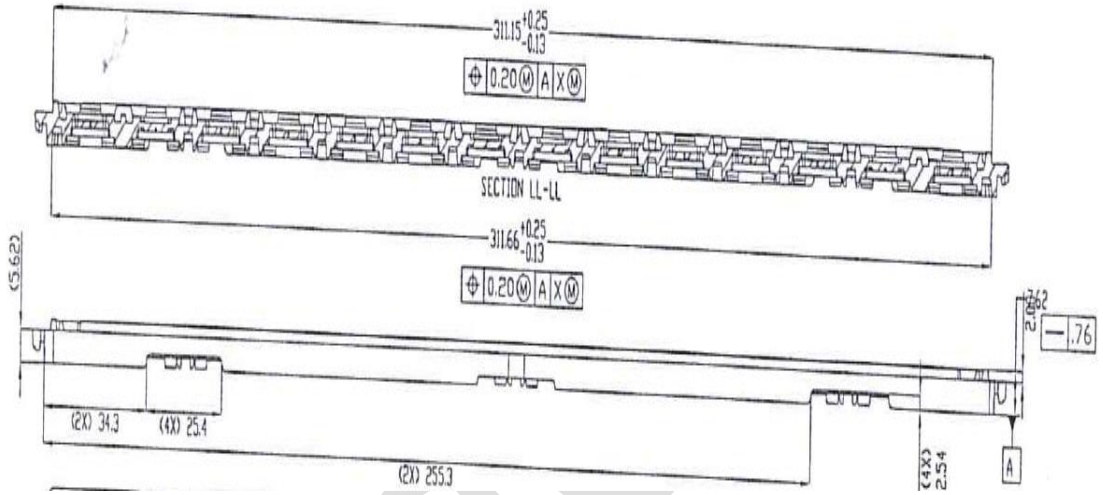
Table 12-2. Packing Quantity Information

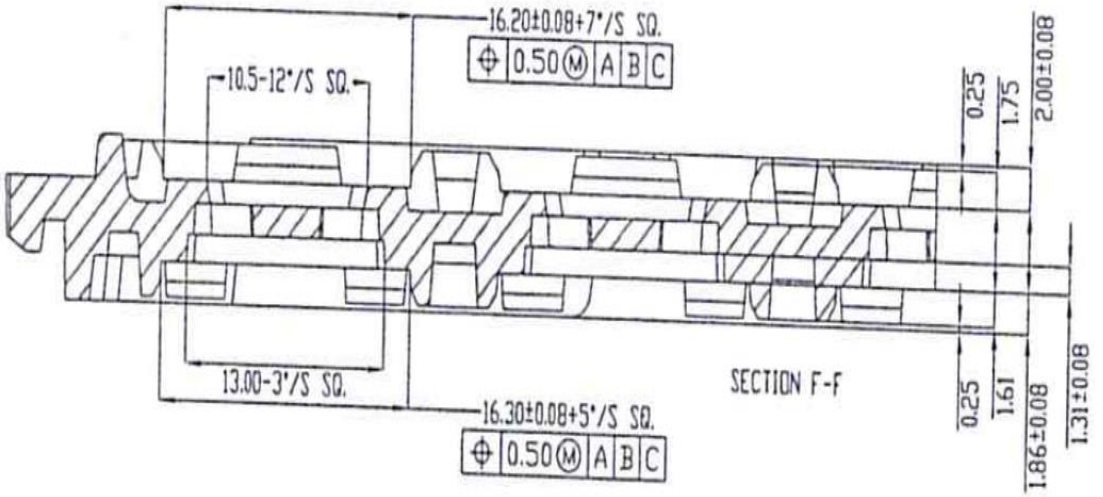
Sample	Size(mm)	Qty/Tray	Tray/Inner Box	Full Inner Box Qty	Inner Box/Carton	Full Carton Qty
V8130L	16 x 16	84	10	840	6	5040

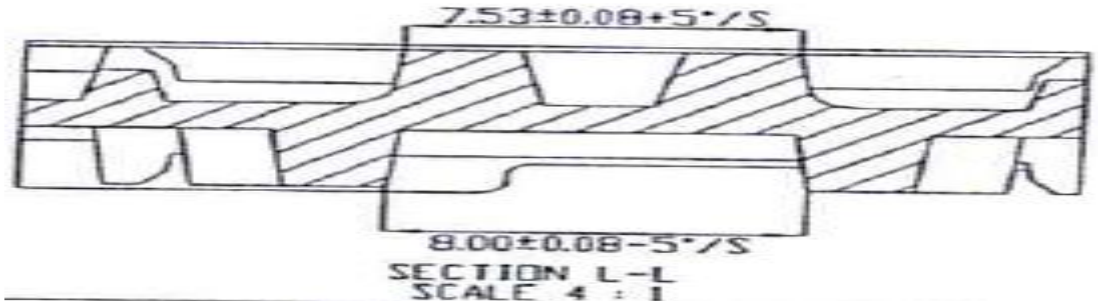
Figure 12-1 shows tray dimension drawing of the V8130L.

Vangotech


LEFT END TAB
RIGHT END TAB
(322.6)
315.0
WW
(4X)
R2.54
BGA1616
LL
UB1616Z.00614XA
150°CMAX
(2X) 92.1
135.9
(2X)12.7
14.45
21.40
22.00
14.50
(3X)R1.6


311.15 +0.25 -0.13
0.20 M A X M
SECTION LL-LL
311.66 +0.25 -0.13
0.20 M A X M
(5.62)
.76
(2X) 34.3
(4X) 25.4
(2X) 255.3
(4X) 2.54
A


16.20±0.08+7°/S SQ.
0.50 M A B C
10.5-12°/S SQ.
0.25
1.75
2.00±0.08
13.00-3°/S SQ.
SECTION F-F
16.30±0.08+5°/S SQ.
0.50 M A B C
0.25
1.61
1.86±0.08
1.31±0.08


7.53±0.08+5°/S
8.00±0.08-5°/S
SECTION L-L
SCALE 4 : 1

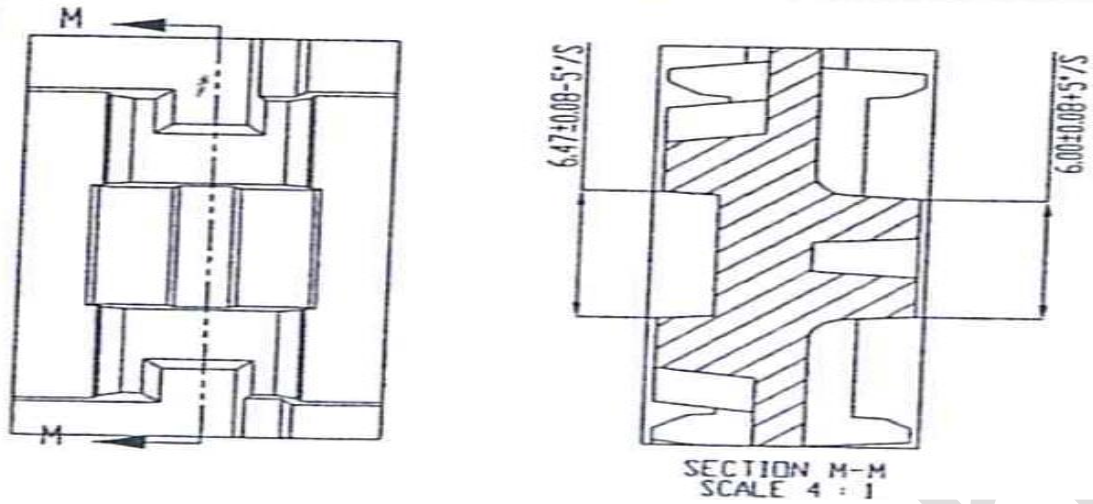


Figure 12-1. Tray Dimension Drawing

12.2. Storage

Reliability is affected if any condition specified in section 12.2.2 and section 12.2.3 has been exceeded.

12.2.1. Matrix Tray Information

A package's MSL indicates its ability to withstand exposure after it is removed from its shipment bag, a low MSL device sample can be exposed on the factory floor longer than a high MSL device sample. All MSL are defined in Table 12-3.

Table 12-3. MSL Summary

MSL	Out-of-bag floor life	Comments
1	Unlimited	≤30°C / 85%RH
2	1 year	≤30°C / 60%RH
2a	4 weeks	≤30°C / 60%RH
3	168 hours	≤30°C / 60%RH
4	72 hours	≤30°C / 60%RH
5	48 hours	≤30°C / 60%RH
5a	24 hours	≤30°C / 60%RH
6	Time on Label(TOL)	≤30°C / 60%RH

NOTE

The V8130L device samples are classified as MSL3.

12.2.2. Bagged Storage Conditions

The shelf life of the V8130L device samples is defined in Table 12-4.

Table 12-4. Bagged Storage Conditions

Packing mode	Vacuum packing
Storage temperature	20°C ~26°C
Storage humidity	40%~60%RH

Shelf life	12 months

12.2.3. Out-of-bag Duration

It is defined by the device MSL rating, the out-of-bag duration of the V8130L is as follows.

Table 12-5. Out-of-bag Duration

Storage temperature	20°C ~26°C
Storage humidity	40%~60%RH
Moisture sensitive level(MSL)	3
Floor life	168 hours

For no mention of storage rules in this document, please refer to the latest ***IPC/JEDEC J-STD-020C***.

12.3. Baking

It is not necessary to bake the V8130L if the conditions specified in section 12.2.2 and section 12.2.3 have not been exceeded. It is necessary to bake the V8130L if any condition specified in section 12.2.2 and section 12.2.3 has been exceeded.

It is necessary bake the V8130L if the storage humidity condition has been exceeded, we recommend that the device sample removed from its shipment bag more than 2 days shall be baked to guarantee production.

Baking conditions: 125°C, 8 hours, nitrogen protection. Note that baking should not exceed 3 times.

Vangotech

Chapter 13. Reflow Profile

All VANGO chips provided for clients are lead-free RoHS-compliant products.

The reflow profile recommended in this document is a lead-free reflow profile that is suitable for pure lead-free technology of lead-free solder paste. If customers need to use lead solder paste, please contact with VANGO FAE.

The lead-free reflow profile conditions are given in Table 13-1. The table is for reference only.

Table 13-1. Lead-free Reflow Profile Conditions

	QTI typical SMT reflow profile conditions(for reference only)	
	Step	**Reflow condition**
Environment	N2 purge reflow usage (yes/no)	Yes, N2 purge used
	If yes, O2 ppm level	O2 < 1500 ppm
A	Preheat ramp up temperature range	25°C -> 150°C
B	Preheat ramp up rate	1.5~2.5 °C /sec
C	Soak temperature range	150°C -> 190°C
D	Soak time	80~110 sec
E	Liquidus temperature	217°C
F	Time above liquidus	60-90 sec
G	Peak temperature	240-250°C
H	Cool down temperature rate	≤4°C /sec

Figure 13-1 shows the typical lead-free reflow profile.

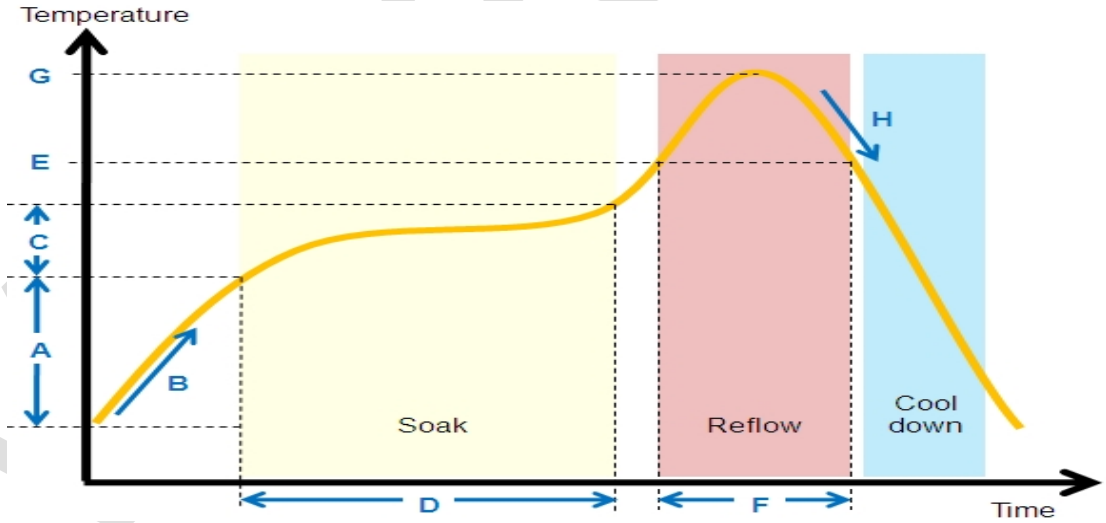


Figure 13-1. Typical Lead-free Reflow Profile

NOTE

The above reflow profile is solder joint testing result, it is for reference only, please adjust depending on actual production conditions.

The method of measuring the reflow soldering process is as follows.

Fix the thermocouple probe of the temperature measuring line at the connection point between the pin

(solderable end) of the packaged device and the pad by using high-temperature solder wire or high-temperature tape, fix the packaged device at the pad by using high-temperature tape or other methods, and cover over the thermocouple probe. See Figure 13-2.

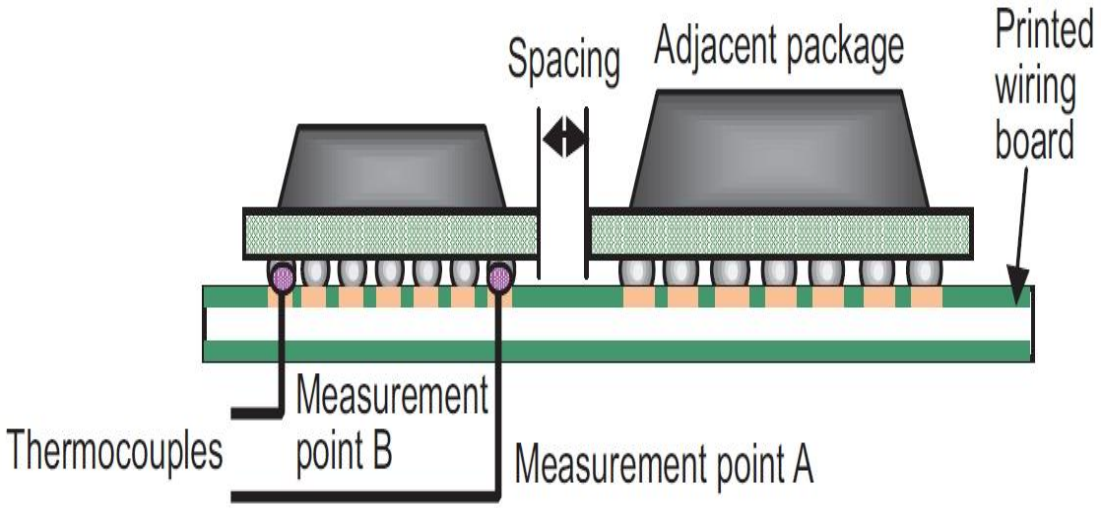


Figure 13-2. Measuring the Reflow Soldering Process

NOTE

To measure the temperature of QFP-packaged chip, place the temperature probe directly at the pin.

If possible, the more accurate measuring way is to drill the packaged device, or drill the PCB, and fix the thermocouple probe through the drilled hole at the pad.

Vangotech

Chapter 14. Part Marking

Figure 14-1 shows the V8130L marking.

Figure 14-1. V8130L Marking

Table 14-1 describes the V8130L marking definitions.

Table 14-1. V8130L Marking Definitions

NO.	Marking	Description	Fixed/Dynamic
1	Vango	Logo	
2	V8130L	Part number	
3	XXXXX-XXX	Wafer lot ID	
4	H121YYWW	H121+Date code Date code : YYWW	

Vangotec