乐升电子
LEVETOP

LT32U02

LT32A02

High Performance 32bit Micro Controller

Data Sheet

V1.0

www.levetop.cn
Levetop Electronics Co., Ltd.

Contents

1. Introduction 15
2. Pin Assignment 15
3. Features 16
4. Block Diagram 19
5. Signal Properties Summary 20
6. Signal Descriptions 22
7. Embedded Interrupt Controller 26
7.1 Introduction 26
7.2 Features 26
7.3 Memory Map and Registers 27
7.3.1 Memory Map (Base: 0xe000_0000) 27
7.3.2 Registers 28
7.3.2.1 Interrupt Control Status Register 28
7.3.2.2 Interrupt Enable Register 29
7.3.2.3 Interrupt Pend Set Register 30
7.3.2.4 Interrupt Pend Clear Register 31
7.3.2.5 Priority Level Select Registers 32
7.3.2.6 System Priority Level Select Registers 33
7.4 Function Description 34
7.4.1 Interrupt Handling Without Confliction 34
7.4.2 Interrupt With Confliction 35
7.4.3 Pend Trap Function 35
7.5 Interrupts 36
8. C0 Introduction 38
8.1 Features 38
8.2 Microarchitecture Summary 38
8.3 Programming Model 39
8.4 Data Format Summary 40
8.5 Operand Addressing Capabilities 40
8.6 Instruction Set Overview 41

©Levetop Electronics Co., Ltd.

9. Embedded Programmable Timer 43
9.1 Introduction 43
9.2 Memory Map and Registers 43
9.2.1 Memory Map (Base: 0xe000_1000) 43
9.2.2 Registers 44
9.2.2.1 EPT Control Status Register 44
9.2.2.2 EPT Reload Register 45
9.2.2.3 EPT Count Register 46
9.3 Function Description 47
9.3.1 Count Timing 47
10. Chip Configuration Module (CCM) 48
10.1 Introduction 48
10.2 Features 48
10.3 Memory Map and Registers 48
10.3.1 Memory Map (Base: 0x4001_0000) 48
10.3.2 Register Descriptions 49
10.3.2.1 WKUPC — Wakeup Configuration Register 49
10.3.2.2 CRPDC — Chip Reduce Pin Drive Configuration Register 51
10.3.2.3 CPSRC — Chip Pin Slow Rate Configuration Register 53
10.3.2.4 CPPDC — Chip Pin Pull Down Configuration Register 56
10.3.2.5 CTR — Chip Test Register 58
10.3.2.6 CIR — Chip Identification Register 58
11. Clock and Power Control Module 59
11.1 Overview 59
11.2 Features 59
11.3 Clock Structure 59
11.4 Clock Source Select 60
11.4.1 Low-Power Options 60
11.4.1.1 Wait and Doze Modes 60
11.4.1.2 Stop Mode 60
11.5 Memory Map and Registers 61
11.5.1 Module Memory Map (Base: 0x4003_0000) 61
11.5.2 Register Description 62
11.5.2.1 Synthesizer Control Register 62
11.5.2.2 Low Speed Oscillator Control and Status Register 66
11.5.2.3 Module Stop Control Register 68

©Levetop Electronics Co., Ltd.

11.5.2.4 EPT External Clock Source Enable Control Register 70
11.5.2.5 OSC Bist Test Configuration Register1 71
11.5.2.6 OSC Bist Test Configuration Register2 72
11.5.2.7 OSC Bist Test Control Register 73
11.5.2.8 OSC BIST Test Counter Register 74
11.5.2.9 OSC BIST Test Result Register 75

12. Reset Controller Module 76
12.1 Overview 76
12.2 Features 76
12.3 Block Diagram 76
12.4 Memory Map and Registers (Base: 0x4002_0000) 77
12.4.1 Reset Test Register 77
12.4.2 Reset Status Register 78
12.4.3 Reset Control Register 79
12.5 Functional Description 80
12.5.1 Reset Sources 80
12.5.1.1 Power-On Reset (POR) 80
12.5.1.2 Watchdog Timer Reset 80
12.5.1.3 Software Reset 80
12.5.1.4 Programmable Voltage Detect Reset 80
12.5.2 Reset Control Flow 81

13. Static Random Access Memory (SRAM) 82
13.1 Introduction 82
13.2 Modes of Operation 82
13.3 Low-Power Modes 82
13.4 Reset Operation 82
13.5 Interrupts 82

14. Cache (CACHE) 83
14.1 Features 83
14.2 Address Space Model 83
14.3 Cache Organization 84
14.4 Cache Operation 85
14.4.1 Cache Line Tag Format 86
14.4.2 Cache Control 87
14.4.2.1 Cache Control Register Format 87

©Levetop Electronics Co., Ltd.

14.4.3 Cache Access Operations 88
14.4.3.1 Cache Search 88
14.4.3.2 Cache Fill 88
14.5 Cache Management 88
14.6 Cache Memory Map 89
15. Embedded Flash Module 91
15.1 Introduction 91
15.2 Features 91
15.3 Modes of Operation 91
15.4 Block Diagram 92
15.5 Module Memory Map (Base: 0x4012_0000) 93
15.6 Register Descriptions 94
15.6.1 EFM Configuration Register (EFMCR) 94
15.6.2 EFM Security Read Register0 (EFMSEC0) 96
15.6.3 EFM Security Read Register1 (EFMSEC1) 97
15.6.4 EFM Security Read Register2 (EFMSEC2) 98
15.6.5 EFM Timing Register0 (EFMTIM0) 99
15.6.6 EFM Timing Register1 (EFMTIM1) 100
15.6.7 EFM Command Register (EFMCMD) 101
15.6.8 EFM Status Register (EFMSTAT) 101
15.7 Functional Description 102
15.7.1 Program and Erase Operations 102
15.7.1.1 Setting the EFMTIM0/1 Register 102
15.7.1.2 Program, Erase and Verify Sequences 102
15.7.1.3 Flash Illegal Operations 102
16. Option Byte 103
16.1 Register Memory Map (Base: 0x4012_0000) 103
16.1.1 Register Descriptions 103
16.1.1.1 PVDC— Programmable Voltage Detector Configuration Register 103
16.1.1.2 CCR — Customer Configuration Register 105
16.1.1.3 EOSCST— External Oscillator Stable Time Configuration Register 108
16.1.1.4 IOSCST— Internal High Speed Osc. Stable Time Configuration Register . 109
16.1.1.5 RFEVR— RESET Pin Filter Enable and Value Register 109
16.1.1.6 PVDFEVR— Programmable Voltage Detector Filter Control Register 110
16.1.1.7 FCR — Factory Configuration Register 111
16.1.1.8 IOSCTC— Internal High Speed Oscillator Trimming Config. Register 113
16.1.1.9 ADCCDISR— ADC Channel Disable Configuration Register 114

©Levetop Electronics Co., Ltd.

16.1.1.10 VREFTCR— VREF Trimming Configuration Register 114
16.1.1.11 LDO1P5TC— LDO1P5 Trimming Configuration Register 115
16.1.1.12 MPUCONFR— Memory Protect Unit Trimming Config. Register 116
16.1.1.13 LDO3P3TC— LDO3P3Trimming Configuration Register 117

17. LDMA Controller 118
17.1 Introduction 118
17.2 Features 118
17.3 Low-Power Mode Operation 118
17.4 Block Diagram 118
17.5 Module Memory Map (Base: 0x4000_0000) 119
17.6 Register Descriptions 120
17.6.1 LDMA Status Register (LDMASR) 120
17.6.2 LDMA Memory Base Address Register (LDMAMBARx) 121
17.6.3 LDMA Byte Count Register (LDMABCRx) 121
17.6.4 LDMA Control Register (LDMACRx) 122
17.7 Function Description 125
17.7.1 Circular Mode 125
17.7.2 Channel Configuration Procedure 125

18. Programmable Interrupt Timer Modules (PIT) 126
18.1 Introduction 126
18.2 Block Diagram 126
18.3 Modes of Operation 127
18.3.1 Wait Mode 127
18.3.2 Doze Mode 127
18.3.3 Stop Mode 127
18.3.4 Debug Mode 127
18.4 Signals 127
18.5 Memory Map and Registers 128
18.5.1 Memory Map 128
18.5.2 Registers 128
18.5.2.1 PIT Modulus Register 128
18.5.2.2 PIT Control and Status Register 129
18.5.2.3 PIT Count Register 131
18.6 Functional Description 132
18.6.1 Set-and-Forget Timer Operation 132
18.6.2 Free-Running Timer Operation 132

©Levetop Electronics Co., Ltd.

18.6.3 Timeout Specifications 133
18.7 Interrupt Operation 133
19. Watchdog Timer Module 134
19.1 Introduction 134
19.2 Modes of Operation 134
19.2.1 Wait Mode 134
19.2.2 Doze Mode 134
19.2.3 Stop Mode 134
19.2.4 Debug Mode 134
19.3 Block Diagram 135
19.4 Signals 135
19.5 Memory Map and Registers 136
19.5.1 Memory Map (Base: 0x4013_0000, 0x4014_0000) 136
19.5.2 Registers 136
19.5.2.1 Watchdog Modules Register 137
19.5.2.2 Watchdog Control Register 137
19.5.2.3 Watchdog Service Register 139
19.5.2.4 Watchdog Count Register 139
20. Edge Port Module (EPORT) 140
20.1 Introduction 140
20.2 Low-Power Mode Operation 141
20.2.1 Wait and Doze Modes 141
20.2.2 Stop Mode 141
20.3 Interrupt/General-Purpose I/O Pin Descriptions 141
20.4 Memory Map and Registers 142
20.4.1 Memory Map (Base: 0x400f_0000, 0x4010_0000) 142
20.4.2 Registers 142
20.4.2.1 Edge Port Interrupt Enable Register 143
20.4.2.2 EPORT Data Direction Register 143
20.4.2.3 EPORT Pin Assignment Register 144
20.4.2.4 EPORT Pin Pull-up Enable Register 145
20.4.2.5 Edge Port Flag Register 145
20.4.2.6 Edge Port Pin Data Register 146
20.4.2.7 Edge Port Data Register 146
20.4.2.8 EPORT Port Bit Set Register 147
20.4.2.9 EPORT Digital Filter Control Register 147
20.4.2.10 EPORT Open Drain Enable Register 148

©Levetop Electronics Co., Ltd.

20.4.2.11 EPORT Level Polarity Register 148
20.4.2.12 EPORT Port Bit Clear Register 148

21. Sensor Controller Module (SCM) 149
21.1 Overview 149
21.2 Features 149
21.3 Block Diagram 149
21.4 Signal Description 150
21.5 Memory Map and Registers (Base: 0x400c_0000) 150
21.5.1 Register Description 151
21.5.1.1 SCM Control Register 151
21.5.1.2 SCM Status Register 154
21.5.1.3 SCM channel-0 Buffer Register 156
21.5.1.4 SCM Channel-1 Buffer Register 158
21.5.1.5 SCM Baud Rate Register 159
21.5.1.6 SCM Bit Period Select Register 160
21.5.1.7 SCM GPIO Control and Status Register 161
21.5.1.8 SCM DMA Control Register 162
21.6 Operation 163
21.6.1 Variable Clock Mode 163
21.6.2 Clock Polarity 163
21.6.3 Serial Communication 163

22. Serial Communications Interface Module (SCI) 164
22.1 Introduction 164
22.2 Features 164
22.3 Modes of Operation 164
22.4 Block Diagram 165
22.5 Modes of Operation 166
22.5.1 Doze Mode 166
22.5.2 Stop Mode 166
22.6 Signal Description 166
22.6.1 RXD 166
22.6.2 TXD 166
22.7 Memory Map and Registers (Base: 0x4009_0000) 167
22.7.1 SCI Control Register 2 168
22.7.2 SCI Control Register 1 170
22.7.3 SCI Baud Rate Divisor Registers 172

©Levetop Electronics Co., Ltd.

22.7.4 SCI Data Registers 174
22.7.5 SCI Status Register 2 175
22.7.6 SCI Status Register 1 176
22.7.7 SCI Data Direction Register 177
22.7.8 SCI Port Data Register 178
22.7.9 SCI Pullup and Reduced Drive Register 179
22.7.10 SCI InfraRed Divisor Register 180
22.7.11 SCI InfraRed Control Register 181
22.7.12 SCI Test Register 182
22.8 Functional Description 183
22.9 Data Format 183
22.10 Serial IR (SIR) 184
22.11 Baud Rate Generation 185
22.12 Transmitter 186
22.12.1 Frame Length 187
22.12.2 Transmitting A Frame 188
22.12.3 Break Frames 189
22.12.4 Idle Frames 189
22.13 Receiver 190
22.13.1 Frame Length 190
22.13.2 Receiving A Frame 190
22.13.3 Data Sampling 190
22.13.4 Framing Errors 195
22.13.5 Baud Rate Tolerance 196
22.13.5.1 Slow Data Tolerance 196
22.13.5.2 Fast Data Tolerance 197
22.13.6 Receiver Wakeup 198
22.13.6.1 Idle Input Line Wakeup (WAKE = 0) 198
22.13.6.2 Address Mark Wakeup (WAKE = 1) 198
22.14 Single-Wire Operation 199
22.15 Loop Operation 200
22.16 I/O Ports 200
22.17 Reset 200
22.18 Interrupts 201
22.18.1 Transmit Data Register Empty 201
22.18.2 Transmission Complete 201
22.18.3 Receive Data Register Full 201
22.18.4 Idle Receiver Input 201

©Levetop Electronics Co., Ltd.

22.18.5 Overrun 201

23. USB2.0 Full-Speed Device Controller 202

23.1 Introduction 202

23.2 Features 202

23.3 Block Diagram 203

23.4 Modes of Operation 204

23.4.1 Wait Mode 204

23.4.2 Doze Mode 204

23.4.3 Stop Mode 204

23.5 Memory Map and Registers 205

23.5.1 Memory Map 205

23.5.2 Registers 207

23.5.2.1 USBPHY Control Register 1 (USBPHY_CTRL1) 207

23.5.2.2 Interrupt Status Register (INT_STAT) 208

23.5.2.3 Interrupt Enable Register (INT_ENB) 209

23.5.2.4 Error Interrupt Status Register (ERR_STAT) 210

23.5.2.5 Error Interrupt Enable Register (ERR_ENB) 211

23.5.2.6 Status Register (STAT) 213

23.5.2.7 Control Register (CTL) 214

23.5.2.8 Address Register (ADDR) 215

23.5.2.9 EBT Page Register 1 (EBT_PAGE_01) 216

23.5.2.10 Frame Number Register (FRMNUML) 217

23.5.2.11 Frame Number Register (FRMNUMH) 218

23.5.2.12 EBT Page Register 2 (EBT_PAGE_02) 218

23.5.2.13 EBT Page Register 3 (EBT_PAGE_03) 219

23.5.2.14 Endpoint Control Registers (ENDPTn, n=0-7) 219

23.5.2.15 USB Resume Enable Register (USB_RESMEN) 221

23.5.2.16 USB PHY Control Register3 (USBPHY_CTRL3) 222

23.6 Function Description 223

23.6.1 Data Structure 223

23.6.2 Endpoint Buffer Table 223

23.6.3 Rx vs. Tx as a USB Target Device 224

23.6.4 Addressing Endpoint Buffer Table Entries 224

23.6.5 Endpoint Buffer Table Formats 225

23.6.6 USB Transaction 227

24. I2C 229

24.1 Introduction 229

©Levetop Electronics Co., Ltd.

24.2 Features 229
24.3 System and Block Diagram 230
24.4 Memory Map and Registers (Base: 0x4015_0000) 231
24.4.1 I2C Status Register (I2CS) 231
24.4.2 I2C Clock Prescalar Register (I2CP) 233
24.4.3 I2C Control Register (I2CC) 233
24.4.4 I2C Slave Address Register (I2CSA) 235
24.4.5 I2C Port Control Register (I2CPCR) 235
24.4.6 I2C Slave High-Speed Mode Indicator Register (I2CSHIR) 236
24.4.7 I2C Slave SDA Hold Time Register (I2CSHT) 236
24.4.8 I2C Data Register (I2CD) 237
24.4.9 I2C Port Direction Register (I2CDDR) 237
24.4.10 I2C Port Data Register (I2CPDR) 238
24.5 Functional Description 239
24.5.1 Master Mode 239
24.5.2 Slave Mode 239
24.5.3 Protocol 239
24.5.4 Arbitration Procedure 241
24.5.5 Clock Synchronization 241
24.5.6 Handshaking 241
24.5.7 Clock Stretching 241
24.5.8 High-Speed Mode Operation 242
24.5.9 Software Transaction Flow Diagrams 244
25. Pulse Width Modulator (PWM) 246
25.1 Introduction 246
25.2 Features 246
25.3 Block Diagram 247
25.4 Signal Description 247
25.5 Memory Map and Registers 248
25.5.1 Memory Map 248
25.5.2 Registers 249
25.5.2.1 PWM Pre-Scale Register (PPR) 249
25.5.2.2 PWM Clock Select Register (PCSR) 250
25.5.2.3 PWM Control Register (PCR) 251
25.5.2.4 PWM Counter Register (PCNR0/1/2/3) 253
25.5.2.5 PWM Comparator Register (PCMR0/1/2/3) 255
25.5.2.6 PWM Timer Register (PTR0/1/2/3) 258
25.5.2.7 PWM Interrupt Enable Register (PIER) 260

©Levetop Electronics Co., Ltd.

25.5.2.8 PWM Interrupt Flag Register (PIFR) 261
25.5.2.9 PWM Capture Control Register (PCCR0/1) 262
25.5.2.10 PWM Capture Rising Latch Register (PCRLR0/1/2/3) 264
25.5.2.11 PWM Capture Falling Latch Register (PCFLR0/1/2/3) 266
25.5.2.12 PWM Port Control Register (PPCR) 268
25.6 Functional Descriptions 269
25.6.1 PWM Double Buffering and Automatic Reload 269
25.6.2 Modulate Duty Ratio 269
25.6.3 Dead-Zone Generator 270
25.6.4 PWM Timer Start Procedure 270
25.6.5 PWM Timer Stop Procedure 270
25.6.6 Capture Start Procedure 271
25.6.7 Capture Basic Timer Operation 271
26. Comparator Modules (COMP) 272
26.1 Introduction 272
26.2 Block Diagram 272
26.3 Modes of Operation 273
26.3.1 Wait Mode 273
26.3.2 Doze Mode 273
26.3.3 Stop Mode 273
26.4 Memory Map and Registers 274
26.4.1 Memory Map (Base: 0x400a_0000, 0x400b_0000) 274
26.4.2 Registers 274
26.4.2.1 Comparator Control Register 274
26.4.2.2 Comparator Mode Selection Register 275
26.4.2.3 Comparator MUX Selection Register 276
26.4.2.4 Comparator Output Filter Selection Register 277
26.5 Function Description 278
27. Analog-to-Digital Converter (ADC) 279
27.1 Introduction 279
27.2 ADC Main Features 279
27.3 ADC Functional Description 280
27.3.1 ADC On-Off Control (ADEN, ADDIS, ADRDY) 280
27.3.2 ADC Clock 281
27.3.3 Configuring the ADC 282
27.3.4 Channel Selection (CCWi) 282
27.3.5 Programmable Sampling Time (SMP) 283

©Levetop Electronics Co., Ltd.

27.3.6 Single Conversion Mode (CONT=0) 283
27.3.7 Continuous Conversion Mode (CONT=1) 284
27.3.8 Starting Conversions (ADSTART) 284
27.3.9 Timings 285
27.3.10 Stopping An Ongoing Conversion (ADSTP) 285
27.4 Conversion On External Trigger and Trigger Polarity 286
27.4.1 Discontinuous Mode (DISCEN) 287
27.4.2 Programmable Resolution (RES) - Fast Conversion Mode 287
27.4.3 End of Conversion, End of Sampling Phase (EOC, EOSMP Flags) 287
27.4.4 End of Conversion Sequence (EOSEQ Flag) 287
27.4.5 Example Timing Diagrams 288
27.5 Data management 290
27.5.1 Data FIFO & Data Alignment (ADC_FIFO, ALIGN) 290
27.5.2 ADC Overrun (OVR, OVRMOD) 291
27.5.3 Managing A Sequence of Data Converted Without Using The DMA 291
27.5.4 Managing Converted Data Without Using The DMA Without Overrun 291
27.5.5 Managing Converted Data Using DMA 292
27.6 Low Power Features 292
27.6.1 Wait Mode Conversion 292
27.6.2 Auto-Off Mode (AUTOFF) 292
27.7 Analog Window Watchdog 293
27.8 Temperature Sensor 294
27.9 ADC Interrupts 294
27.10 Memory Map and Registers 295
27.10.1 Memory Map (Base: 0x4011_0000) 295
27.10.2 Registers 296
27.10.2.1 ADC Interrupt and Status Register (ADC_ISR) 296
27.10.2.2 ADC Interrupt Enable Register (ADC_IER) 297
27.10.2.3 ADC Control Register (ADC_CR) 298
27.10.2.4 ADC Configuration Register 1 (ADC_CFGR1) 300
27.10.2.5 ADC Configuration Register 2 (ADC_CFGR2) 301
27.10.2.6 ADC Sampling Time Register (ADC_SMPR) 302
27.10.2.7 ADC Watchdog Register (ADC_WDG) 303
27.10.2.8 ADC Watchdog Threshold Register (ADC_TR) 304
27.10.2.9 ADC Channel Selection Register (ADC_CHSELR1, ADC_CHSELR2) 305
27.10.2.10 ADC FIFO Access Register (ADC_FIFO) 306
28. Mechanical Specifications 307
28.1 LT32U02(QFN48) Mechanical Drawing 307

©Levetop Electronics Co., Ltd.

28.2 LT32A02(QFN32) Mechanical Drawing 309
29. Absolute Maximum Ratings 310
30. DC Electrical Specifications 310
31. Revision 312
32. Copyright 312

©Levetop Electronics Co., Ltd.

LT32U02
LT32A02

High Performance 32bit Micro Controller

1. Introduction

The LT32x02 incorporates the High-performance C0 32-bit RISC core operating at a 72MHz maximum frequency with high-speed embedded Flash memories (Flash up to 64Kbytes which contains an 1kbytes two-way set associative read Cache and SRAM up to 8Kbytes), and an extensive range of enhanced peripherals and I/Os. All devices offer standard communication interfaces, including two Master SPI, one I2C, and one UART. The LT32U02 also provide one USB2.0 Fast Speed controller. Both of LT3202 support up to four General-purpose 16-bit timers, one embedded programmable 24-bit timers, eight advanced-control PWM timers and two asynchronous Watch Dog Timers. It also provides Analog modules including one 1MSPS ADC with 8-channels and two Comparators.

The LT32U02 is QFN-48pin package, LT32A02 is QFN-32pin package. The operation temperature range is -40°C to 85°C and the operating voltage is 2.0/2.8V~5.5V.

Package Types



LT32U02
QFNWB6X6-48L



LT32A02
QFNWB5X5-32L

2. Pin Assignment

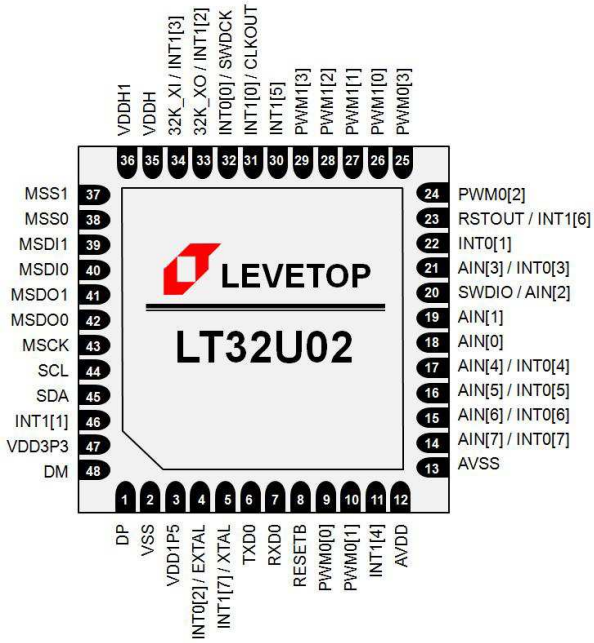


Figure 2–1: LT32U02 Pin Assignment

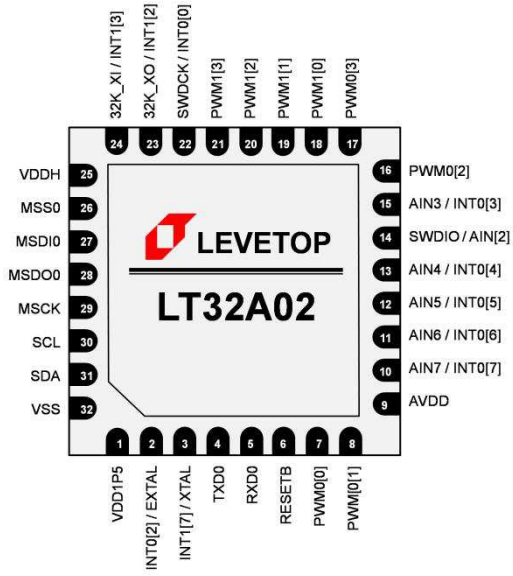


Figure 2–2: LT32A02 Pin Assignment

©Levetop Electronics Co., Ltd.

3. Features

■ C0 Processor

_ 32-bit Load/Store Reduced Instruction set Computer (RISC) Architecture with fixed 16-bit Instruction Length
_ 16 Entry 32-bit General-Purpose Register File
_ Efficient 3-Stage Execution Pipeline, Hidden from Application Software
_ Single-cycle Instruction Execution for many Instructions, Three Cycles for Branches
_ Support for Byte / Half-word / Word Memory Accesses
_ Embedded Interrupt Controller, support Nested Vector Interrupts.
_ Single-cycle 32-bit x 32-bit Hardware Integer Multiplier Array
_ 3~13 Cycles Hardware Integer Divider Array

■ 8K Bytes of Static Random-Access Memory (SRAM)

_ Single Cycle Byte, Half-word (16-bit), and Word (32-bit) Reads and Writes

■ 64K Bytes Embedded Flash (EFLASH)

_ Page Erase Capability: 512 bytes per Page
_ Read Cycle Time: 40ns (min)
_ Endurance: 100000 Cycles (min)
_ Greater than 20 Years Data Retention
_ Fast Page Erase/ Byte Program
- Half-word Program Time: 7.5us (max)
- Page Erase Time: 5ms (max)
- Mass Erase Time: 40ms (max)

_ Single Cycle Byte, Half-word (16-bit) and Word (32-bit) Read Access.

■ Reset

_ Internal power on reset circuit
_ Five sources of reset:
- Power-on Reset
- External Pin
- Software Reset
- Watchdog Timer
- Program Voltage Detect Reset

_ Status Flag Indicates Source of Last Reset

■ Four Periodic Interval Timer

_ 16-bit Counter with Modulus "initial count" Register
_ Selectable as Free Running or Count down
_ 16 Selectable Prescalars → 2^0 to 2^{15}

■ Two Watchdog Timer

_ 16-bit Counter with Modulus "Initial Count" Register
_ Pause Option for Low-power Modes
_ Up to 2000ms Service Time

©Levetop Electronics Co., Ltd.

■ Two External Interrupts Port (EPORT)

_ Eight Channels for Each EPORT
_ Rising/falling Edge Select
_ Low/High-Level Sensitive
_ Interrupt Pins Configurable as General-purpose I/O

■ Two Serial Peripheral Interface Master Module (SPI)

_ SPI Master Mode
_ Shared SPICLK Ports
_ Two 4 Entries deep Read FIFO
_ Two 4 Entries deep Write FIFO
_ Interrupt Generation after 1, 2, 3, or 4 Transferred Bytes
_ MSB/LSB Selectable
_ Variable Baud-rate During Communication
_ Serial Clock with Programmable Polarity and Phase
_ 16/32 bit Transmit/Receive Data Width
_ Byte Re-order
_ Controllable Slave Select (spiss0/1) bit

■ One Serial Communications Interface Module (SCI)

_ Standard Mark/Space Non-Return-to-Zero (NRZ) format
_ The Baud Rate Divisor is a 22-bit Number Consisting of 16-bit Integer and 6-bit Fractional Part
_ Programmable 7-bit, 8-bit or 9-bit Data Format
_ Separately enabled Transmitter and Receiver
_ Separate Receiver and Transmitter Central Processor unit (CPU) Interrupt Requests
_ Programmable Transmitter Output Polarity
_ Two Receiver Wakeup Methods:
  • Idle Line Wakeup
  • Address Mark Wakeup
_ Interrupt-driven Operation with Eight Flags:
  • Transmitter Empty
  • Transmission Complete
  • Receiver Full
  • Idle receiver Input
  • Receiver Overrun
  • Noise Error
  • Framing Error
  • Parity Error
_ Receiver Framing Error Detection
_ Hardware Parity Checking
_ 1/16 bit-time Noise Detection
_ General-purpose, I/O Capability
_ Serial IR Interface Low-speed, IrDA-Compatible (up to 115.2Kbit/s)

■ One USB2.0 Full Speed Compatible Device (LT32U02 Only)

_ Supports Internal Reference Clock or External 12MHz Crystal Reference Clock
_ Compliant with USB2.0 Full Speed Specification with on-chip Integrated PHY Module
_ Supports FS (12Mbps) Mode
_ Up to 8 Endpoints Supported Including Endpoint 0
_ All Endpoints except Endpoint 0 can support Interrupt and Bulk Transfer
_ All Endpoints except Endpoint 0 can be Configured as 8, 16, 32, 64 bytes FIFO size
_ Endpoint 0 Support Control Transfer

©Levetop Electronics Co., Ltd.

■ Two Pulse Width Modulator (PWM)

_ Four Channel each PWM Controller
_ Programmable Period
_ Programmable Duty Cycle
_ Two Dead-Zone Generator
_ Capture Function
_ Pins can be Configured as General-purpose I/O

■ ADC with 8-Channel

_ High Performance
- 12-bit, 10-bit, 8-bit or 6-bit configurable resolution
- ADC Conversion Time: 1.0μs for 12-bit Resolution (1 MHz), 0.88μs Conversion Time for 10 bit Resolution, Faster Conversion times can be obtained by Lowering Resolution.
- Programmable Sampling Time
- Data Alignment with Built-in Data Coherency
- DMA Support

_ Low Power
- Application cans Reduce PLCK Frequency for Low Power Operation while still Keeping Optimum ADC Performance. For Example, 1.0μs Conversion Time is kept, whatever the Frequency of PCLK.
- Wait mode: Prevents ADC Overrun in Applications with Low Frequency PLCK
- Auto off mode: ADC is Automatically Powered Off Except during the Active Conversion phase. This Dramatically Reduces the Power Consumption of the ADC.

_ Analog Input Channels
- 8 External Analog Inputs
- 1 Channel for Internal Reference Voltage
- 1 Channel for Internal Temperature Sensor

_ Start-of-Conversion can be Initiated:
- By Software
- By Hardware Triggers with Configurable Polarity

_ Conversion Modes
- Can Convert a Single Channel or can Scan a Sequence of Channels.
- Single Mode Converts Selected Inputs once per Trigger
- Continuous Mode Converts Selected Inputs Continuously
- Discontinuous Mode

_ Interrupt Generation at the end of Sampling, end of Conversion, end of Sequence Conversion, and in case of Analog Watchdog or Overrun Events.
_ Analog Watchdog
_ Single-ended and Differential-input Configurations
- Converter uses an Internal Reference or an External Reference

■ Two Analog Comparators

_ Programmable Response Time
_ Programmable Hysteresis
_ Support Analog input Multiplexer with Nine Selection
_ Two Optional Outputs: Filtered or Asynchronous Output
_ Selectable Rising/Falling Edge Interrupt

©Levetop Electronics Co., Ltd.

■ PMU

_ Support 3.3V LDO (for USB PHY Power Supply) with Maximum Load Current 100mA
_ 3.3V LDO Support Power Down for Save Power Consumption
_ Support 1.5V LDO with Maximum Load Current 50mA
_ 1.5V LDO Support Four Mode: Normal, Lower power, High power

■ Programmable Voltage Detector

■ Internal Oscillator

_ 128KHz Oscillator Clock for Watchdog and PMU
_ 72MHz Oscillator Clock which can be used for System Clock
_ 48MHz USBPLL Clock which can be used for System Clock

■ External Crystal Oscillator

_ Up to 20Mhz External Crystal Oscillator Clock which can be used for System Clock
_ 32.768Khz External Crystal Oscillator Clock which can be used for Watchdog

4. Block Diagram

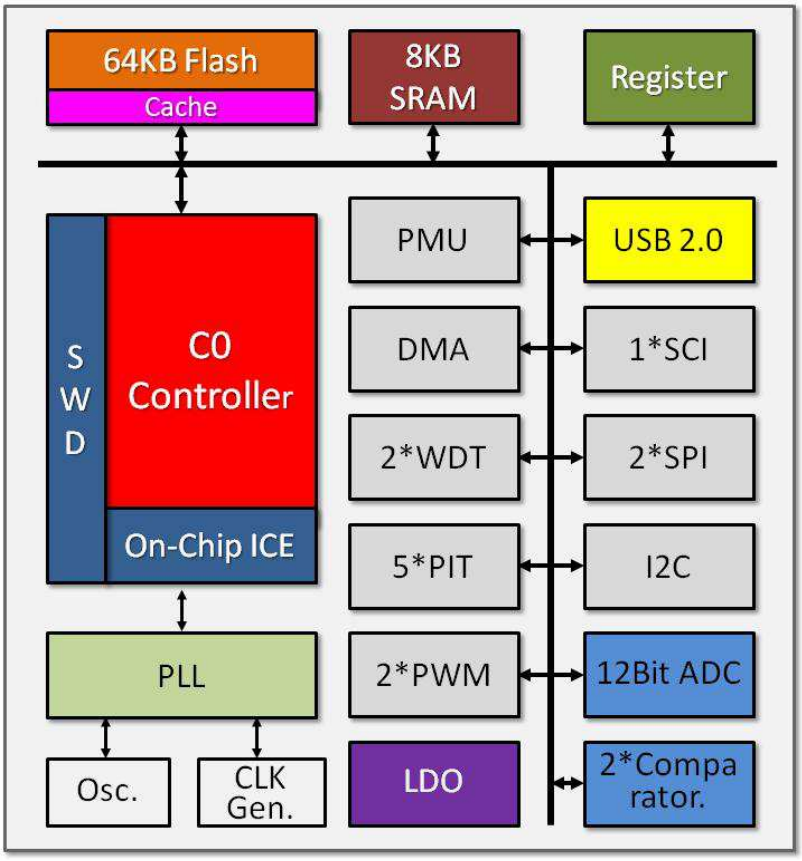


Figure 4–1: LT32U02 Block Diagram

©Levetop Electronics Co., Ltd.

5. Signal Properties Summary

Table 5–1: Signal Properties Summary

Name	Alternate	Qty.	Dir.	Input Sync	Pullup[1]	Output Drive (ST/OD/SP)[2]
SCI (2)						
RXD0		1	I/O	N	Pull-up	ST
TXD0		1	I/O	N	Pull-up	ST
USB (2)						
DP		1	Analog			SP
DM		1	Analog			SP
I2C (2)						
SCL		1	I/O	N	Pull-up	OD
SDA		1	I/O	N	Pull-up	OD
SPI (7)						
MSS0		1	I/O	N	-	ST
MSDI0		1	I/O	N	Pull-up	ST
MSDO0		1	I/O	N	-	ST
MSS1		1	I/O	N	-	ST
MSDI1		1	I/O	N	Pull-up	ST
MSDO1		1	I/O	N	-	ST
MSCK		1	I/O	N	-	ST
PWM0 (4)						
PWM0[0]		1	I/O	N	Pull-up	ST
PWM0[1]		1	I/O	N	Pull-up	ST
PWM0[2]		1	I/O	N	Pull-up	ST
PWM0[3]		1	I/O	N	Pull-up	ST
PWM1 (2)						
PWM1[0]		1	I/O	N	Pull-up	ST
PWM1[1]		1	I/O	N	Pull-up	ST
PWM1[2]		1	I/O	N	Pull-up	ST
PWM1[3]		1	I/O	N	Pull-up	ST
ADC (7)						
AIN[7:3]	INT0[7:3]	5	Mixed	N	-	-
AIN[1:0]	-	2	Analog	N	-	-

©Levetop Electronics Co., Ltd.

Table 5–2: Signal Properties Summary (Continued)

Name	Alternate	Qty.	Dir.	Input Sync	Pullup[1]	Output Drive (ST/OD/SP)[2]
Edge Port 0 (1)						
INT0[1]	-	1	I/O	N	Pull-up	ST
Edge Port 0 (5)						
INT1[1]		1	I/O	N	Pull-up	ST
INT1[2]	32K_XO	1	I/O	N	Pull-up	ST
INT1[3]	32K_XI	1	I/O	N	Pull-up	ST
INT1[4]		1	I/O	N	Pull-up	ST
INT1[5]		1	I/O	N	Pull-up	ST
Debug Port (2)						
SWDIO	AIN[2]	1	I/O	N	Pull-up	ST
SWDCK	INT0[0]	1	I/O	N	Pull-up	ST
Clock (3)						
EXTAL	INT0[2]	1	I/O	N	-	SP
XTAL	INT1[7]	1	I/O	N	-	SP
CLKOUT	INT1[0]	1	I/O	N	Pull-up	ST
RESET (2)						
RESETB	-	1	-	Y	-	SP
RSTOUT	INT1[6]	1	I/O	N	Pull-up	ST
Power Supply						
VDDH	-	-	-	-	-	SP
VDDH1	-	-	-	-	-	SP
VDD3P3	-	-	-	-	-	SP
VDD1P5	-	-	-	-	-	SP
VSS	-	-	-	-	-	SP
AVDD	-	-	-	-	-	SP
AVSS	-	-	-	-	-	SP

Notes:

1. All Pull-ups are disconnected when the signal is programmed as an output.
2. Output Driver Type: ST=Standard, SP=Special, OD=Standard Driver with Open-Drain Pull down option selected.

©Levetop Electronics Co., Ltd.

6. Signal Descriptions

Table 6–1: Signal Descriptions

Pin Name	Pin Number		Pin Description
	LT32U02	LT32A02	
RXD0	7	5	**Receive Data** This is a Serial Communications Interface 0 Module Signal (SCI0). This signal is used for the SCI receiver data input and is also available for GPIO when not configured for receiver operation.
TXD0	6	4	**Transmit Data** This is a Serial Communications Interface 0 Module Signal (SCI0). This signal is used for the SCI transmitter data output and is also available for GPIO when not configured for transmitter operation.
DP	1	--	**USB Data Positive** These signals are used by the USB module.
DM	48	--	**USB Data Negative** These signals are used by the USB module.
SCL	44	30	**I2C Clock** This signal is used for the I2C clock line signal and is also available for GPIO when not configured for receiver operation.
SDA	45	31	**I2C Data** This signal is used for the I2C data line signal and is also available for GPIO when not configured for transmitter operation.
MSDO0	42	28	**Master 0 Out** This signal is the serial data output from the SPI0 in master mode.
MSDO1	41	--	**Master 1 Out** This signal is the serial data output from the SPI1 in master mode.
MSDI0	40	27	**Master 0 In** This signal is the serial data input to the SPI0 in master mode.
MSDI1	39	--	**Master 1 In** This signal is the serial data input to the SPI1 in master mode.

©Levetop Electronics Co., Ltd.

Table 6–1: Signal Descriptions (Continued)

Pin Name	Pin Number		Pin Description
	LT32U02	LT32A02	
MSCK	43	29	**Master Serial Clock** The serial clock synchronizes data transmissions between master and slave devices. MSCK is an output and is shared between with SPI0 and SPI1.
MSS0	38	26	**Slave Select 0** This output signal is the peripheral chip select signal in master mode.
MSS1	37	--	**Slave Select 1** This output signal is the peripheral chip select signal in master mode
INT0[7]	14	10	**Edge Port 0 Signals** These bidirectional signals function as either external interrupt sources or GPIO.
INT0[6]	15	11	
INT0[5]	16	12	
INT0[4]	17	13	
INT0[3]	21	15	
INT0[2]	4	2	
INT0[1]	22	--	
INT0[0]	32	22	
INT1[7]	5	3	**Edge Port 1 Signals** These bidirectional signals function as either external interrupt sources or GPIO.
INT1[6]	23	--	
INT1[5]	30	--	
INT1[4]	11	--	
INT1[3]	34	--	
INT1[2]	33	--	
INT1[1]	46	--	
INT1[0]	31	--	
PWM0[3:0]	25,24,10,9	17,16,8,7	**Pulse Width Modulator 0 Signals** These out signals function as either PMW0 output or GPIO.
PWM1[3:0]	29,28,27,26	21,20,19,18	**Pulse Width Modulator 1 Signals** These out signals function as either PMW1 output or GPIO.

©Levetop Electronics Co., Ltd.

Table 6–1: Signal Descriptions (Continued)

Pin Name	Pin Number		Pin Description
	LT32U02	LT32A02	
AIN[7]	14	10	**Analog-to-Digital Converter Signals** These analog signals function as ADC analog channels.
AIN[6]	15	11	
AIN[5]	16	12	
AIN[4]	17	13	
AIN[3]	21	15	
AIN[2]	20	14	
AIN[1]	19	--	
AIN[0]	18	--	
SWDCLK	32	22	**Test Clock** This input signal is the test clock used to synchronize the SWD logic.
SWDIO	20	14	**Test Data Input / Output** This input/output signal is the serial input/output for test instructions and data.
EXTAL	4	2	**Fast Oscillator Pad Input** The signal is the input of fast Oscillator Pad.
XTAL	5	3	**Fast Oscillator Pad output** The signal is the output fast Oscillator pad.
32K_XI	34	24	**32.768Khz Oscillator Pad input** The signal is the input of 32.768Khz Oscillator Pad.
32K_XO	33	23	**32.768Khz Oscillator Pad output** The signal is the output 32.768Khz Oscillator pad.
CLKOUT	31	--	**Clock Out** This output signal reflects the internal system clock.
RESETB	8	6	**Reset In** This active-low input signal is used as the external reset request. Reset places the CPU in supervisor mode with default settings for all register bits except some register bits only reset by POR. 0 = external reset assert 1 = external reset desert

©Levetop Electronics Co., Ltd.

Table 6–1: Signal Descriptions (Continued)

Pin Name	Pin Number		Pin Description
	LT32U02	LT32A02	
RSTOUT	23	--	**Reset Out** This active-low output signal is an indication that the internal reset controller has reset the chip. 0 = chip is at reset status 1 = chip is not reset status
VDDH	35	25	**Power** This signal supplies 2.5~5.5V positive power to the I/O pads and LDO.
VDDH1	36	--	**Power** This signal supplies 2.5~5.5V positive power to the SPI I/O pads. VDDH1 is short together with VDDH in QFN32 package.
VDD3P3	47	--	**Power** 3.3V LDO output signal and is used to supply the power of USB transceiver. 1uF ceramic bypass capacitor is required to externally connect between the pad and VSS.
VDD1P5	3	1	**Core Power** 1.5V LDO output signal and is used to supply the power of the core logic. 1uF ceramic bypass capacitor is required to externally connect between the pad and VSS.
VSS	2	32	**Ground** This signal supplies 2.5~5.5V negative supply (ground) to the I/O pads and LDO.
AVDD	12	9	**Analog Power** This signal supplies 2.5~5.5V positive power to analog module.
AVSS	13	--	**Analog Ground** This signal is the negative supply to the Analog module.

Notes:

1. The SPI signals are used by the SPI modules and may also be configured to be discrete I/O signals.
2. The VDD and VSS signals provide system power and ground to the chip. Multiple signals are provided for adequate current capability. All power supply signals must have adequate bypass capacitance for high-frequency noise suppression.

©Levetop Electronics Co., Ltd.

7. Embedded Interrupt Controller

This section describes the Embedded Interrupt Controller of C0.

7.1 Introduction

The interrupt controller collects requests from multiple interrupt sources and provides an interface to the CPU interrupt logic.

7.2 Features

Features of the interrupt controller module include:

- Interrupt sources configurable, up to 32
- 32 unique programmable priority levels for each interrupt source
- Independent enable/disable of pending interrupts based on priority level
- A fixed vector number for each interrupt source
- Support both level-sensitive and pulse interrupts
- Support PendTrap function
- Support Software reset

©Levetop Electronics Co., Ltd.

7.3 Memory Map and Registers

This subsection describes the memory map (see **Table 7–1**) and registers.

7.3.1 Memory Map (Base: 0xe000_0000)

EIC module base address (EIC_BASEADDR) is defined in C0 internal parameter. The default value is 0xE0000000. The EIC registers actual address is EIC_BASEADDR plus the offset address of each EIC registers. The core internal modules occupy 64K address area. The system should avoid mapping the other registers to the area from EIC_BASEADDR to EIC_BASEADDR+0x0000_ffff.

Table 7–1: Interrupt Controller Module Memory Map

Offset Address	Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	Access(1)
0x0000_0000	Interrupt control status register (ICSR)				S/U
0x0000_0004	Reserved				S/U
0x0000_0008	Reserved				S/U
0x0000_000c	Reserved				S/U
0x0000_0010	Interrupt Enable Register (IER)				S/U
0x0000_0014	Reserved				S/U
0x0000_0018	Interrupt Pending Set Register (IPSR)				S/U
0x0000_001c	Interrupt Pending Clear Register (IPCR)				S/U
0x0000_0020 through 0x0000_003c	Unimplemented(2)				—
Priority level select registers (PLSR0-PLSR31)					
0x0000_0040	PLSR3	PLSR2	PLSR1	PLSR0	S/U
0x0000_0044	PLSR7	PLSR6	PLSR5	PLSR4	S/U
0x0000_0048	PLSR11	PLSR10	PLSR9	PLSR8	S/U
0x0000_004c	PLSR15	PLSR14	PLSR13	PLSR12	S/U
0x0000_0050	PLSR19	PLSR18	PLSR17	PLSR16	S/U
0x0000_0054	PLSR23	PLSR22	PLSR21	PLSR20	S/U
0x0000_0058	PLSR27	PLSR26	PLSR25	PLSR24	S/U
0x0000_005c	PLSR31	PLSR30	PLSR29	PLSR28	S/U
0x0000_0060	System Priority level select register(SYSPLSR)				S/U
0x0000_0064 through 0x0000_007c	Unimplemented(2)				—

Notes:

1. In C0, there register can be accessed in any case.
2. Accesses to unimplemented address locations have no effect and result in a cycle termination transfer error.

©Levetop Electronics Co., Ltd.

7.3.2 Registers

This subsection contains a description of the interrupt controller module registers.

7.3.2.1 Interrupt Control Status Register

The 32-bit interrupt control register (ICSR) reflects the state of the interrupt controller

Address: EIC_BASEADDR+0x0000_0000

	31	30	29	28	27	26	25	24
R	SRTA	0	0	SetPTrap	ClrPTrap	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	0	VEC6	VEC5	VEC4	VEC3	VEC2	VEC2	VEC0
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

outputs to the CPU.

Figure 7–1: Interrupt Control Status Register (ICSR)

SRST — Software Reset Bit

The write-only bit is used to create a software reset request. Setting this bit will generate a pulse on SYSRESETREQ signal. Reads always return 0.

SetPTrap — Set PendTrap Bit

The read/write bit is used to create a pending software interrupt. The action is similar to execute “trap” instruction. However the pending software interrupt will not be entered until all the higher priority exceptions/interrupts exit. When the software interrupt entered, the bit will be cleared automatically. Reset also clears this bit.

On reads:

1 = the software interrupt is pending

0 = the software interrupt is not pending

On writes:

1 = set software interrupt to pending

0 = no effect

©Levetop Electronics Co., Ltd.

ClrPTrap — Clear PendTrap Bit

The read/write ClrDSI bit is used to cancel the pending software interrupt (PendTrap). Reset clears this bit.

On reads:

1 = the software interrupt is pending

0 = the software interrupt is not pending

On writes:

1 = cancel the pending software interrupt

0 = no effect

VEC [6:0] — Interrupt Vector Number Field

The read-only VEC [6:0] field contains the 7-bit interrupt vector number. Reset clears VEC [6:0].

7.3.2.2 Interrupt Enable Register

The read/write, 32-bit Interrupt Enable Register (IER) individually enables any current pending interrupts which are assigned to each priority level as a normal interrupt source. Enabling an interrupt source which has an asserted request causes that request to become pending and a request to the CPU is asserted if not already outstanding.

Address: EIC_BASEADDR+0x0000_0010

	31	30	29	28	27	26	25	24
R W	IE31	IE30	IE29	IE28	IE27	IE26	IE25	IE24
RESET	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
R W	IE23	IE22	IE21	IE20	IE19	IE18	IE17	IE16
RESET	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
R W	IE15	IE14	IE13	IE12	IE11	IE10	IE9	IE8
RESET	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
R W	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 7–2: Interrupt Enable Register (IER)

IE[31:0] — Interrupt Enable Field

The read/write IE[31:0] field enables interrupt requests from sources at the corresponding priority level as interrupt requests. Reset clears IE[31:0].

1 = interrupt request enabled

0 = interrupt request disabled

©Levetop Electronics Co., Ltd.

7.3.2.3 Interrupt Pend Set Register

Address: EIC_BASEADDR+0x0000_0018

	31	30	29	28	27	26	25	24
R W	SetPend31	SetPend30	SetPend29	SetPend28	SetPend27	SetPend26	SetPend25	SetPend24
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R W	SetPend23	SetPend22	SetPend21	SetPend20	SetPend19	SetPend18	SetPend17	SetPend16
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R W	SetPend15	SetPend14	SetPend13	SetPend12	SetPend11	SetPend10	SetPend9	SetPend8
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R W	SetPend7	SetPend6	SetPend5	SetPend4	SetPend3	SetPend2	SetPend1	SetPend0
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 7–3: Interrupt Pend Set Register (IPSR)

SetPend[31:0] — Interrupt Pend Set Field

The read/write SetPend[31:0] field set pend to associated interrupt and indicate whether the associated interrupt is pending . Reset clears SetPend[31:0].

On reads:

1 = the associated interrupt is pending

0 = the associated interrupt is not pending

On writes:

1 = change the state of associated interrupt to pending

0 = no effect

©Levetop Electronics Co., Ltd.

7.3.2.4 Interrupt Pend Clear Register

Address: EIC_BASEADDR+0x0000_001C

	31	30	29	28	27	26	25	24
R W	ClrPend31	ClrPend30	ClrPend29	ClrPend28	ClrPend27	ClrPend26	ClrPend25	ClrPend24
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R W	ClrPend23	ClrPend22	ClrPend21	ClrPend20	ClrPend19	ClrPend18	ClrPend17	ClrPend16
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R W	ClrPend15	ClrPend14	ClrPend13	ClrPend12	ClrPend11	ClrPend10	ClrPend9	ClrPend8
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R W	ClrPend7	ClrPend6	ClrPend5	ClrPend4	ClrPend3	ClrPend2	ClrPend1	ClrPend0
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 7–4: Interrupt Pend Clear Register (IPCR)

ClrPend[31:0] — Interrupt Pend Clr Field

The read/write ClrPend[31:0] field clear pend to associated interrupt and indicate whether the associated interrupt is pending . Reset clears ClrPend[31:0].

On reads:

1 = the associated interrupt is pending

0 = the associated interrupt is not pending

On writes:

1 = change the state of associated interrupt to not pending

0 = no effect

©Levetop Electronics Co., Ltd.

7.3.2.5 Priority Level Select Registers

The read/write 8-bit Priority Level Select Registers (PLSRx) are 32 read/write 8-bit priority level select registers PLSR0–PLSR31, one for each of the interrupt source. The PLSRx register assigns a priority level to interrupt source x.

Address: EIC_BASEADDR+0x0000_0040 through EIC_BASEADDR+0x0000_005c

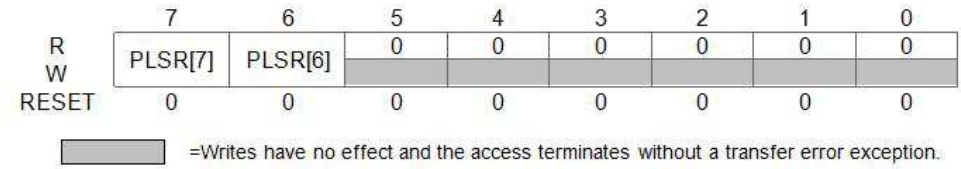


Figure 7–5: Priority Level Select Registers (PLSR0-PLSR31)

PLSRx[7:6] — Priority Level Select Field

IRQ0~31 has a default priority value 0~31.The lower the value, the higher the priority. That means IRQ0 priority > IRQ1 > ... > IRQ31 as default. However, user can set PLSRx[7:6] to adjust the interrupt priority. The actual value of priority level is the default value plus PLSRx[7:6] *64. For instance, if PLSR1[7:6] = 2, IRQ1's priority value is 1+2*64 = 129, then IRQ1's priority is lower than any IRQ with lower priority value.

Table 7–2: Priority Value Adjustment

PLSRx[7:6]	Pulsed Priority Value
00	0
01	64
10	128
11	192

©Levetop Electronics Co., Ltd.

7.3.2.6 System Priority Level Select Registers

Address: EIC_BASEADDR+0x0000_0060

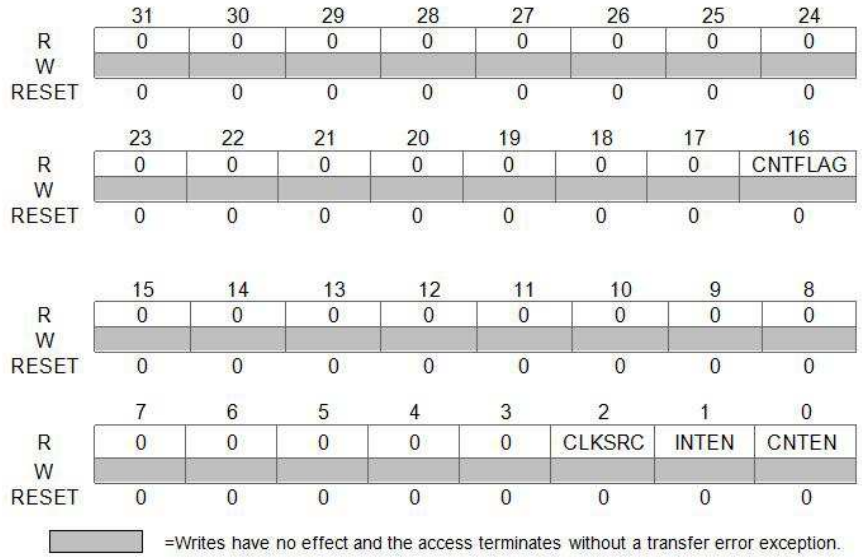


Figure 7–6: System Priority Level Select Registers (SYSPLSR)

EPTPRI[7:6] — EPT Priority Level Select Field

EPT interrupt's default priority is -2. That means EPT interrupt priority is higher than the other normal IRQs and system software interrupt (PendTrap) as default. The lower the value, the higher the priority. However, user can set PRI[7:6] to adjust the EPT interrupt priority. The actual value of priority level is the default value plus PRI[7:6] *64. For instance, if PRI[7:6] = 2, EPT interrupt priority value is -2+2*64 = 126.

Table 7–3: Priority Value Adjustment

PRI[7:6]	Adjusted Priority Value
00	0
01	64
10	128
11	192

SIPRI[7:6] — Software Interrupt Priority Level Select Field

Software interrupt (PendTrap) default priority is -1. That means EPT interrupt priority is higher than the other normal IRQs as default. The lower the value, the higher the priority. However, user can set SIPRI[7:6] to adjust the software interrupt priority. The actual value of priority level is the default value plus SIPRI[7:6] *64. For instance, if SIPRI[7:6] = 2, software interrupt priority value is -1+2*64 = 127.

Table 7–4: Priority Value Adjustment

PRI[7:6]	Adjusted Priority Value
00	0
01	64
10	128
11	192

©Levetop Electronics Co., Ltd.

7.4 Function Description

EIC supports both level-sensitive and pulse interrupts. Interrupt source number is from 1 to 32.

The interrupt becomes pending because one of the following reasons:

- the EIC detects that the interrupt signal is active and the corresponding interrupt is not active
- the EIC detects a rising edge on the interrupt signal

The pending interrupt remains pending until one of the following:

- The processor enters the ISR for the interrupt. This changes the state of the interrupt from pending to active. Then:
  - For a level-sensitive interrupt, when the processor returns from the ISR, the EIC samples the interrupt signal. If the signal is asserted, the state of the interrupt changes to pending, which might cause the processor to immediately re-enter the ISR. Otherwise, the state of the interrupt changes to inactive.
  - For a pulse interrupt, the EIC continues to monitor the interrupt signal, and if this is pulsed the state of the interrupt changes to pending and active. In this case, when the processor returns from the ISR the state of the interrupt changes to pending, which might cause the processor to immediately re-enter the ISR. If the interrupt signal is not pulsed while the processor is in the ISR, when the processor returns from the ISR the state of the interrupt changes to inactive.
- Software writes to the corresponding interrupt Pend Clear Register bit.

7.4.1 Interrupt Handling Without Confliction

If an interrupt is pulsed, the state of the interrupt changes to pending. Without confliction, the interrupt causes the processor to immediately enter the ISR. When the processor returns from the ISR, the state of the interrupt changes to inactive.

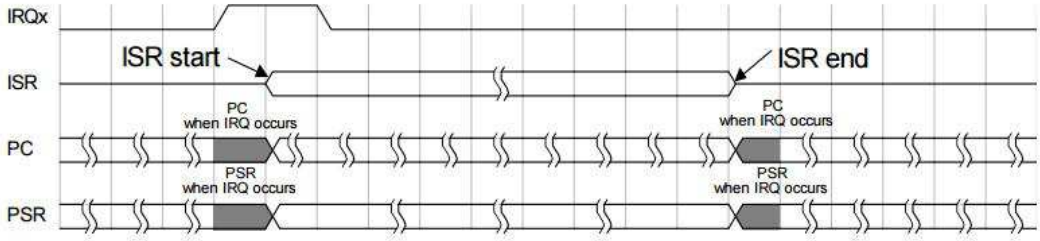


Figure 7–7: One Pulse Interrupt Without Confliction

For a level-sensitive interrupt, the state of the interrupt changes to pending if the signal is asserted. Without confliction, the interrupt causes the processor to immediately enter the ISR. When the processor returns from the ISR, EIC continues to samples the interrupt signal. If the signal is not cleared, the processor will re-enter the ISR. Otherwise, the state of the interrupt changes to inactive.

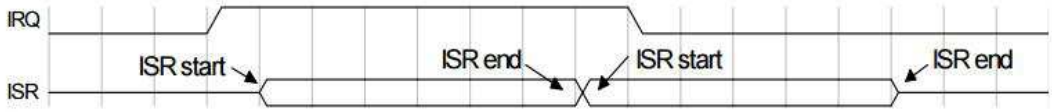


Figure 7–8: Level-sensitive Interrupt Without Confliction

©Levetop Electronics Co., Ltd.

7.4.2 Interrupt With Confliction

When two interrupt signals are asserted at the same time , the Interrupt Arbiter will judge which one has the greater priority. For instance, if the priority of IRQx is greater than IRQy , the processor will enter ISRx and IRQy becomes pending. After the processor returns from ISRx, the processor will enter ISRy immediately.

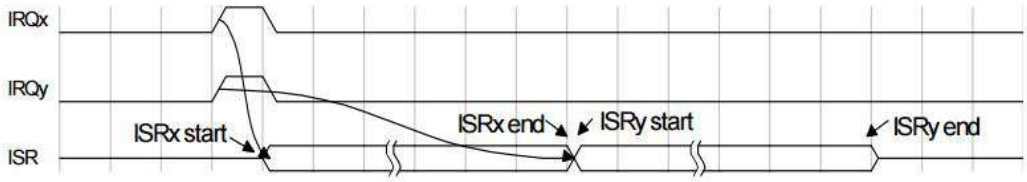


Figure 7–9: Two Interrupts Occur at The Same Time

.If an interrupt signal is asserted during another interrupt handling, there are two cases:

1. The asserted interrupt priority is lower than the handling interrupt priority. In this case, the asserted interrupt state is pending until the handling interrupt ends.
2. The asserted interrupt priority is higher than the handling interrupt priority. In this case, the higher priority interrupt handling will be nested in the lower priority interrupt routine.

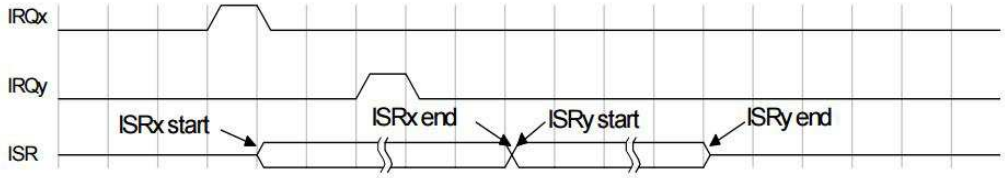


Figure 7–10: A Lower Priority Interrupt Asserted With Confliction

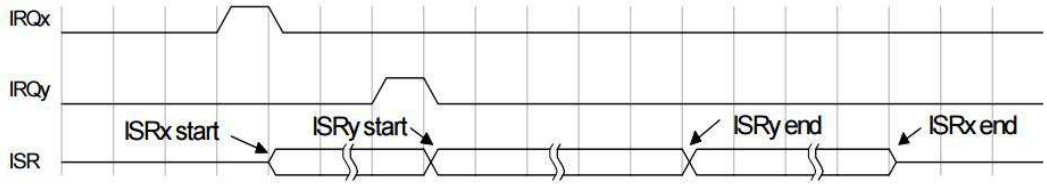


Figure 7–11: A Higher Priority Interrupt Asserted With Confliction

7.4.3 Pend Trap Function

SetPTrap/ClrPTrap bits in ICSR are used to create/cancel a “pending” software interrupt request while a higher priority interrupt is handling .As soon as the processor returns from the higher priority interrupt, the “pending” software interrupt will be accepted by the processor.

Usually, Pend Trap function is used for OS task passive switch.

©Levetop Electronics Co., Ltd.

7.5 Interrupts

The interrupt controller assigns a number to each interrupt source, as **4.5** shows.

Table 7–5: Interrupt Source Assignment

Source	Module	Flag	Source Description	Flag Clearing Mechanism
0	ADC			
1	SCM	SPI0F	SPI channel-0 transfer done	
		SPI1F	SPI channel-1 transfer done	
		WCOL0	SPI channel-0 Write FIFO Overrun	
		WCOL1	SPI channel-1 Write FIFO Overrun	
		RCOL0	SPI channel-0 Read FIFO Overrun	
		RCOL1	SPI channel-1 Read FIFO Overrun	
		WFEMPTY0	SPI channel-0 Write FIFO Empty	
		WFEMPTY1	SPI channel-1 Write FIFO Empty	
2	SCI0	TDRE/TFTS		
		TC/FTC		
		RDRF/RFTS		
		OR/FOR		
		IDLE		
		RTOS		
3	COMP0	CPRIF		
		CPFIF		
4	COMP1	CPRIF		
		CPFIF		
5	LDMAC	DONE[0]		Write DONE[0]=1
		DONE[1]		Write DONE[1]=1
		DONE[2]		Write DONE[2]=1
		DONE[3]		Write DONE[3]=1
		DONE[4]		Write DONE[4]=1
6	WDT0	IF		
7	PWM0	PIFR[0]		
		PIFR[1]		
		PIFR[2]		
		PIFR[3]		

©Levetop Electronics Co., Ltd.

Table 7–5: Interrupt Source Assignment (Continued)

Source	Module	Flag	Source Description	Flag Clearing Mechanism
8	PWM1	PIFR[0]		
		PIFR[1]		
		PIFR[2]		
		PIFR[3]		
9	PIT0	PIF	PIT Flag	Writing a 1 to it or writing to PMR
10	PIT1	PIF	PIT Flag	Writing a 1 to it or writing to PMR
11	PIT2	PIF	PIT Flag	Writing a 1 to it or writing to PMR
12	PIT3	PIF	PIT Flag	Writing a 1 to it or writing to PMR
13	WDT1	IF		
14	USB_DEV	USB_DEV Flag	USB_DEV Flag	
15	I2C	I2C Flag or PVDO	I2C Flag or PVD Flag	
16	EPORT0	EPF0	Edge port 0 flag 0	Write EPF0 = 1
17	EPORT0	EPF1	Edge port 0 flag 1	Write EPF1 = 1
18	EPORT0	EPF2	Edge port 0 flag 2	Write EPF2 = 1
19	EPORT0	EPF3	Edge port 0 flag 3	Write EPF3 = 1
20	EPORT0	EPF4	Edge port 0 flag 4	Write EPF4 = 1
21	EPORT0	EPF5	Edge port 0 flag 5	Write EPF5 = 1
22	EPORT0	EPF6	Edge port 0 flag 6	Write EPF6 = 1
23	EPORT0	EPF7	Edge port 0 flag 7	Write EPF7 = 1
24	EPORT1	EPF0	Edge port 1 flag 0	Write EPF0 = 1
25	EPORT1	EPF1	Edge port 1 flag 1	Write EPF1 = 1
26	EPORT1	EPF2	Edge port 1 flag 2	Write EPF2 = 1
27	EPORT1	EPF3	Edge port 1 flag 3	Write EPF3 = 1
28	EPORT1	EPF4	Edge port 1 flag 4	Write EPF4 = 1
29	EPORT1	EPF5	Edge port 1 flag 5	Write EPF5 = 1
30	EPORT1	EPF6	Edge port 1 flag 6	Write EPF6 = 1
31	EPORT1	EPF7	Edge port 1 flag 7	Write EPF7 = 1

©Levetop Electronics Co., Ltd.

8. C0 Introduction

This document describes the functionality of the C0 microprocessor, which is based on M*Core instruction set/architecture and designed for extremely low-power and cost-sensitive embedded control applications.

For the smaller size and power dissipation, C0 is built on a new 3-stage pipeline von Neumann architecture. C0 also integrates an EIC(embedded interrupt controller) to reduce system area.

The external bus interface protocol is AHB-lite. More configurable options are available in C0 design. Leveraging these configurable options, tradeoff among performance, functionality and cost are more flexible. The gate count of C0 varies from 12K to 20K with different configurations.

8.1 Features

The main features of the C0 are as follows:

- 32-bit load/store reduced instruction set computer (RISC) architecture with fixed 16-bit instruction length
- 16 entry 32-bit general-purpose register file
- Efficient 3-stage execution pipeline, hidden from application software
- Single-cycle instruction execution for many Instructions, three cycles for branches
- Support for byte/halfword/word memory accesses
- Embedded interrupt controller, support nested vector interrupts and low power mode wakeup
- Single-cycle 32-bit x 32-bit hardware integer multiplier array
- 3~13 cycles hardware integer divider array
- AHB-lite external bus

8.2 Microarchitecture Summary

The C0 utilizes a 3-stage pipeline for instruction execution. The instruction fetch, instruction decode/register file read, execute/writeback stages operate in an overlapped fashion, allowing single clock instruction execution for most instructions.

16 general-purpose registers are provided for source operands and instruction results. Register R15 is used as the link register to hold the return address for subroutine calls, and Register R0 is associated with the current stack pointer value by convention.

A dual entry 32-bit instruction buffer is provided to allow instruction prefetching to obtain two instructions per clock cycle from memory with a maximum of three buffered instructions, thus reducing or eliminating bus resource conflicts with data memory accesses. The unified bus structure is sufficient to sustain both instruction and data bandwidth requirements without resorting to expensive dual bus structures.

Memory load and store operations are provided for byte, halfword, and word (32-bit) data with automatic zero extension of byte and halfword load data. These instructions can be pipelined to allow effective single cycle throughput for short sequences. Data dependent operations can complete in two clock cycles. Load and store multiple register instructions allow low overhead context save and restore operations; these instructions can execute in (N+1) clock cycles, where N is the numbers of registers to transfer.

A single condition code/carry (C) bit is provided for condition testing and for use in implementing arithmetic and logical operations greater than 32-bits. Typically, the C bit is set only by explicit test/comparison operations, not as a side-effect of normal instruction operation. Exceptions to this rule occur for specialized operations where it is desirable to combine condition setting with actual computation.

©Levetop Electronics Co., Ltd.

8.3 Programming Model

The C0 programming model is defined separately for two privilege modes: supervisor and user. HPROT[1] bit is used to indicate the privilege modes.

Programs access registers based on the indicated mode. User programs can only access registers specific to the user mode; system software executing in the supervisor mode can access all registers, using the control registers to perform supervisory functions. User programs are thus restricted from accessing privileged information, and the operating system performs management and service tasks for the user programs by coordinating their activities.

All instructions execute in either mode. User program can also execute stop, doze, or wait instructions. The trap #n instructions provide controlled access to operating system services for user programs. To prevent a user program from entering the supervisor mode except in a controlled manner, instructions that can alter the S-bit in the program status register (PSR) are privileged.

When the S-bit in the PSR is set, the processor executes instructions in the supervisor mode. Bus cycles associated with an instruction indicate either supervisor or user access depending on the mode.

The processor utilizes the user programming model when it is in normal user mode processing. During exception processing, the processor changes from user to supervisor mode. Exception processing saves the current value of the PSR to stack memory and then sets the S bit in the PSR, forcing the processor into the supervisor mode. To return to the previous operating mode, a system routine may execute the rte (return from exception) instruction, causing the instruction pipeline to be flushed and refilled from the appropriate address space.

The registers depicted in the programming model (see **Figure 8–1**) provide operand storage and control. The user programming model consists of 16 general-purpose 32-bit registers, the 32-bit program counter (PC) and the Condition/Carry (C) bit. The C bit is implemented as bit 0 of the PSR. By convention, register R15 serves as the link register for subroutine calls, and register R0 is typically used as the current stack pointer.

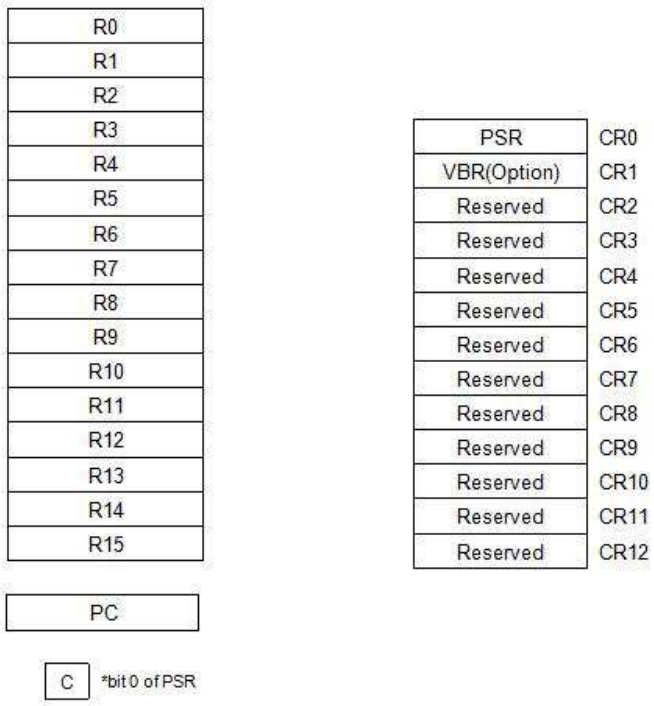


Figure 8–1: Programming Model

©Levetop Electronics Co., Ltd.

8.4 Data Format Summary

The operand data formats supported by the integer unit are standard two's complement data formats. The operand size for each instruction is either explicitly encoded in the instruction (load/store instructions) or implicitly defined by the instruction operation (index operations, byte extraction). Typically, instructions operate on all 32 bits of the source operand(s) and generate a 32-bit result.

Memory may be viewed from either a Big Endian or Little Endian byte ordering perspective depending on the processor configuration (see **Figure 8–2**. In Big Endian mode (the default operating mode), the most significant byte (byte 0) of word 0 is located at address 0. For Little Endian mode, the most significant bye of word 0 is located at address 3. Within registers, bits are numbered within a word starting with bit 31 as the most significant bit (see **Figure 8–3**). By convention, byte 0 of a register is the most significant byte regardless of Endian mode. This is only an issue when executing the **XTRB[0-3]** instructions.

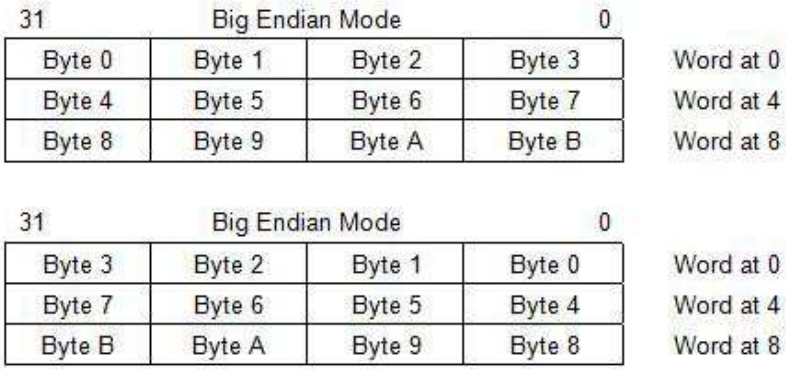


Figure 8–2: Data Organization in Memory

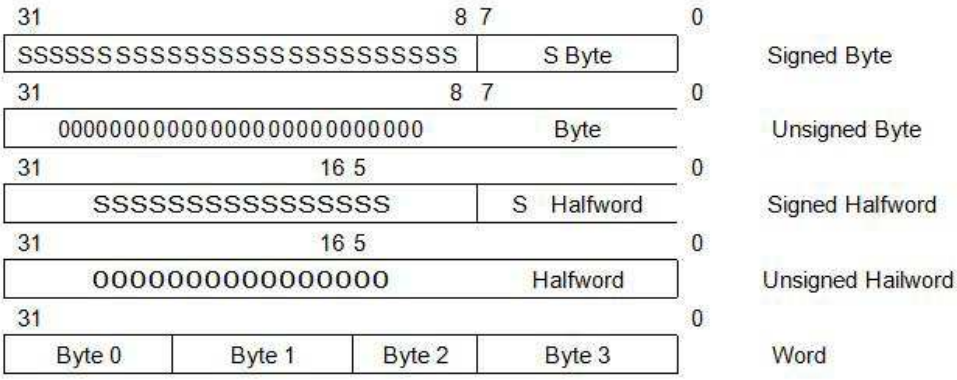


Figure 8–3: Data Organization in Registers

8.5 Operand Addressing Capabilities

C0 accesses all memory operands through load and store instructions, transferring data between the general-purpose registers (GPRs) and memory. Register + 4-bit scaled displacement addressing mode is used for the load and store instructions to address byte, halfword, or word (32 bit) data.

Load and store multiple instructions allow a subset of the 16 GPRs to be transferred to or from a base address pointed to by register R0 (the default stack pointer by convention).

Load and store register quadrant instructions use register indirect addressing to transfer a register quadrant to or from memory.

©Levetop Electronics Co., Ltd.

8.6 Instruction Set Overview

The instruction set is tailored to support high-level languages and is optimized for those instructions most commonly executed. A standard set of arithmetic and logical instructions is provided as well as instruction support for bit operations, byte extraction, data movement, control flow modification, and a small set of conditionally executed instructions which can be useful in eliminating short conditional branches.

Table 8–1 provides an alphabetized listing of the C0 instruction set.

Table 8–1: C0 Instruction Set

Mnemonic	Description
ABS	Absolute Value
ADDC	Add with C bit
ADDI	Add Immediate
ADDU	Add Unsigned
AND	Logical AND
ANDI	Logical AND Immediate
ANDN	AND NOT
ASR	Arithmetic Shift Right
ASRC	Arithmetic Shift Right, update C bit
ASRI	Arithmetic Shift Right Immediate
BCLRI	Clear Bit
BF	Branch on Condition False
BGENI	Bit Generate Immediate
BGENR	Bit Generate Register
BKPT	Breakpoint
BMASKI	Bit Mask Immediate
BR	Branch
BREV	Bit Reverse
BSETI	Bit Set Immediate
BSR	Branch to Subroutine
BT	Branch on Condition True
BTSTI	Bit Test Immediate
CLRF	Clear Register on Condition False
CLRT	Clear Register on Condition True
CMPHS	Compare Higher or Same
CMPLT	Compare Less-Than
CMPLTI	Compare Less-Than Immediate
DECF	Decrement on Condition False
DECGT	Decrement Register and Set Condition if Result Greater-than Zero
DECLT	Decrement Register and Set Condition if Result Less-than Zero
DECNE	Decrement Register and Set Condition if Result Not Equal to Zero
DECT	Decrement On Condition True
DIVS[1]	Divide Signed Integers
DIVU[1]	Divide Unsigned Integers
DOZE	Doze
FF1[1]	Find First One
INCF	Increment on Condition False
INCT	Increment On Condition True
IXH	Index Halfword
IXW	Index Word
JAVASW	Java interpreter switch
JMP	Jump
JMPI	Jump Indirect
JSR	Jump to Subroutine
JSRI	Jump to Subroutine Indirect

©Levetop Electronics Co., Ltd.

Table 8–1: C0 Instruction Set (Continued)

Mnemonic	Description
LD.[BHW]	Load
LDM	Load Multiple Registers
LDQ	Load Register Quadrant
LRW	Load Relative Word
LSL, LSR	Logical Shift Left and Right
LSLC, LSRC	Logical Shift Left and Right, update C bit
LSLI, LSRI	Logical Shift Left and Right by Immediate
MFCR	Move from Control Register
MOV	Move
MOVI	Move Immediate
MOVF	Move on Condition False
MOVT	Move on Condition True
MTCR	Move to Control Register
MULSH	Multiply signed Halfwords
MULT	Multiply
MVC	Move C bit to Register
MVCV	Move Inverted C bit to Register
NOT	Logical Complement
OR	Logical Inclusive-OR
ROTLI	Rotate Left by Immediate
RSUB	Reverse Subtract
RSUBI	Reverse Subtract Immediate
RTE	Return from Exception
RFI	Return from Interrupt
SEXTB	Sign-extend Byte
SEXTH	Sign-extend Halfword
ST.[BHW]	Store
STM	Store Multiple Registers
STQ	Store Register Quadrant
STOP	Stop
SUBC	Subtract with C bit
SUBU	Subtract
SUBI	Subtract Immediate
SYNC	Synchronize
TRAP	Trap
TST	Test Operands
TSTNBZ	Test for No Byte Equal Zero
WAIT	Wait
XOR	Exclusive OR
XSR	Extended Shift Right
XTRB0	Extract Byte 0
XTRB1	Extract Byte 1
XTRB2	Extract Byte 2
XTRB3	Extract Byte 3
ZEXTB	Zero-extend Byte
ZEXTH	Zero-extend Halfword

Note:

1. Not implemented in the current version.

©Levetop Electronics Co., Ltd.

9. Embedded Programmable Timer

9.1 Introduction

Embedded programmable timer (EPT) is a 24-bit timer that provides precise interrupts at regular intervals with minimal processor intervention. The timer can either count down from the reload value, or it can be a free-running down-counter.

EPT interrupt can trigger an exception (vector number = 24).

EPT module can be removed by clearing parameter “EPT” to 0 for reducing core gate count.

9.2 Memory Map and Registers

9.2.1 Memory Map (Base: 0xe000_1000)

EPT base address is defined as EIC_BASEADDR + 0x1000. The default based address is 0xE0001000. **Table 9–1** shows the offset address of EPT registers. EPT module occupies 4K address area.

Table 9–1: Programmable Timer Module Memory Map

Offset Address	Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	Access(1)
0x0000_0000	EPT Control and Status Register (EPTCSR)				S/U
0x0000_0004	EPT Reload Register (EPTRLD)				S/U
0x0000_0008	EPT Count Register (EPTCNT)				S/U
0x0000_000c	Reserved				S/U

Notes:

1. In C0, there register can be accessed in any case.

©Levetop Electronics Co., Ltd.

9.2.2 Registers

This subsection contains a description of the EPT module registers.

9.2.2.1 EPT Control Status Register

Address: EPT_BASEADDR+0x0000_0000

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	CNTFLAG
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	CLKSRC	INTEN	CNTEN
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 9–1: EPT Control Status Register (EPTCSR)

CNTFLAG — Count Down to 0 flag
The read-only bit indicates timer counted to 0. It will be reset by HRESETn.
1 = The timer counted down to 0.
0 = The timer is still counting down.

CLKSRC — Count clock source select
The read/write bit is used to select the count clock source. It will be reset by HRESETn.
1 = Core clock.
0 = External reference clock.

INTEN — EPT Interrupt Request Enable
The read/write bit is used to enable EPT's interrupt when timer counted down to 0. It will be reset by RESET.
1 = EPT exception request occurs when timer counted down to 0.
0 = EPT exception request will not occur when timer counted down to 0.

CNTEN — Counter Enable
The read/write bit is used to enable EPT's counter. It will be reset by RESET.
1 = Counter enabled
0 = Counter disabled

©Levetop Electronics Co., Ltd.

9.2.2.2 EPT Reload Register

Address: EPT_BASEADDR+0x0000_0004

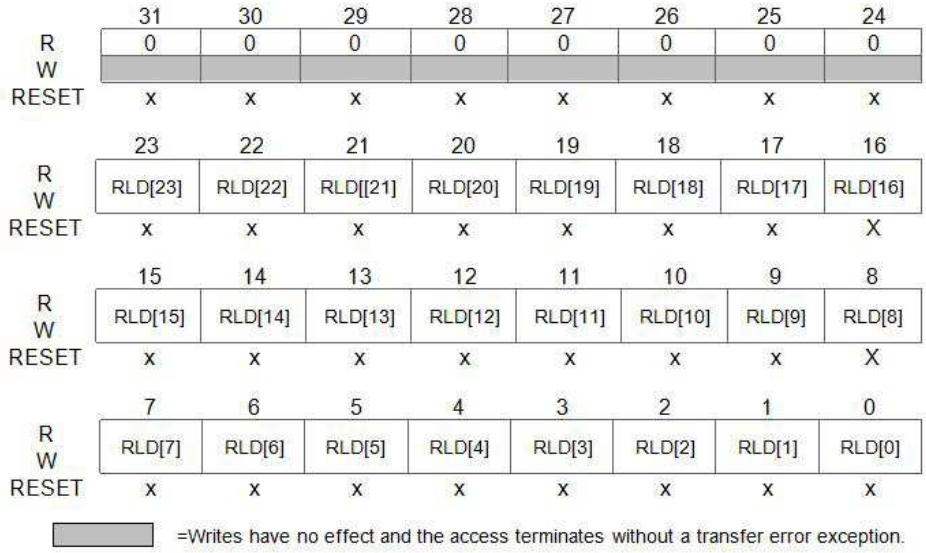


Figure 9–2: EPT Reload Register (EPTRLD)

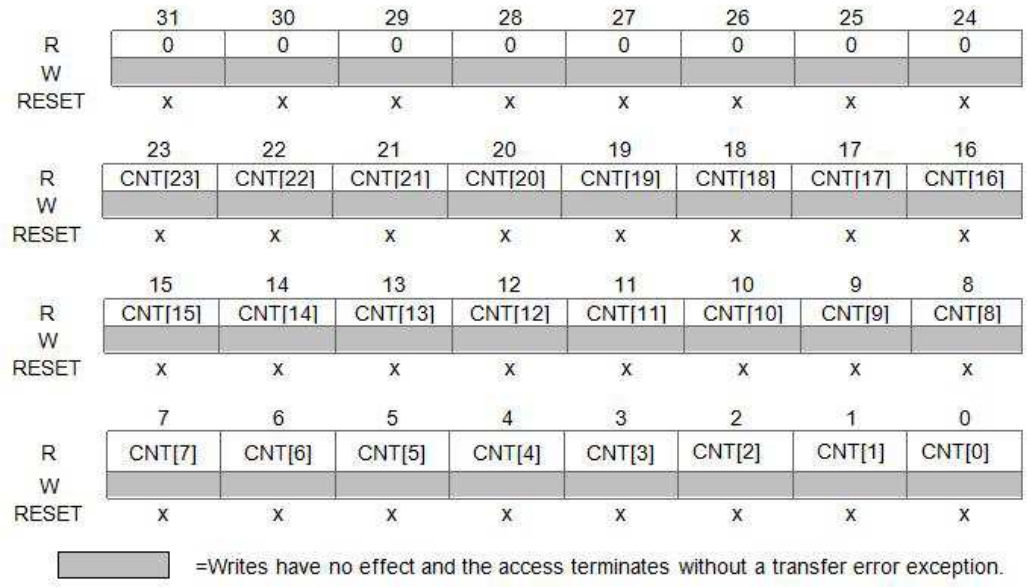
RLD[23:0] — Reload Value

The read/write RLD[23:0] field specifies the reload value when timer counted down to 0 . The register has no reset value. The RLD value can be any value in the range 0x00000001 ~ 0x00FFFFFF. Value 0 has no effect. To generate a period timer with N clock cycles, set RLD to N-1.

©Levetop Electronics Co., Ltd.

9.2.2.3 EPT Count Register

Address: EPT_BASEADDR+0x0000_0008



	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET	x	x	x	x	x	x	x	x

	23	22	21	20	19	18	17	16
R	CNT[23]	CNT[22]	CNT[21]	CNT[20]	CNT[19]	CNT[18]	CNT[17]	CNT[16]
W								
RESET	x	x	x	x	x	x	x	x

	15	14	13	12	11	10	9	8
R	CNT[15]	CNT[14]	CNT[13]	CNT[12]	CNT[11]	CNT[10]	CNT[9]	CNT[8]
W								
RESET	x	x	x	x	x	x	x	x

	7	6	5	4	3	2	1	0
R	CNT[7]	CNT[6]	CNT[5]	CNT[4]	CNT[3]	CNT[2]	CNT[1]	CNT[0]
W								
RESET	x	x	x	x	x	x	x	x

=Writes have no effect and the access terminates without a transfer error exception.

Figure 9–3: EPT Counter Register (EPTCNT)

CNT[23:0] — EPT Counter Value

The read-only register indicates the current count value of EPT timer. The register has not reset value.

Reads will return the current value of EPT counter. A write of any value to this register will clear the counter value to 0 and also clears the CNTFLAG to 0.

©Levetop Electronics Co., Ltd.

9.3 Function Description

When Enabled, EPT count down from the value set by RLD to zero, and wrap reloads the value in RLD on the next clock cycle, then down-counts by subsequent clock cycles, writing value of zero to RLD disables the counter on next wrap. When it counted to zero, the CLFAG bit will set to 1, then the ETP will trig the ETP interrupt if INTEN was enabled.

Reading CSR clears the CFLAG bit to 0. Write any value to CNT also clears the CFLAG bit to 0.

9.3.1 Count Timing

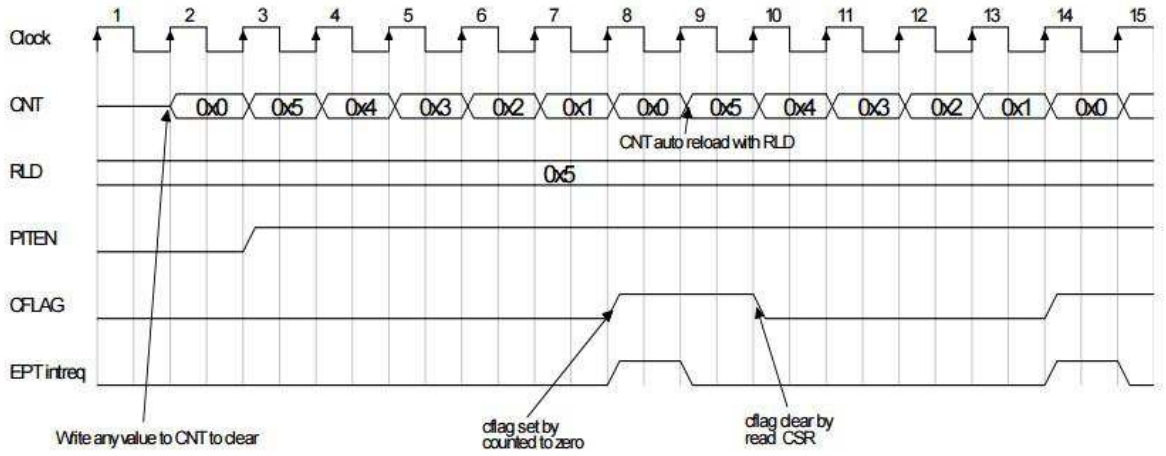


Figure 9–4: EPT Count Timing

©Levetop Electronics Co., Ltd.

10. Chip Configuration Module (CCM)

10.1 Introduction

The chip configuration module (CCM) controls the chip configuration.

10.2 Features

The CCM performs these operations.

- Wakeup function configuration
- LDO mode configuration
- IO function configuration

10.3 Memory Map and Registers

This subsection provides a description of the memory map and registers

10.3.1 Memory Map (Base: 0x4001_0000)

Table 10–1: CCM Memory Map

Offset Address	31:16	15:0	Access[1]
0x0000	WKUPC — Wakeup Configuration Register		S
0x0004	CRPDC — Chip Reduce Pin Drive Configuration Register		S
0x0008	CPSRC — Chip Pin Slow Rate Configuration Register		S
0x000c	CPPDC — Chip Pin Pull Down Configuration Register		S
0x0010	CIR — Chip Identification Register	CTR — Chip Test Register	S

Notes:

1. S = supervisor-only access. User mode accesses to supervisor only address locations have no effect and result in a cycle termination transfer error.

©Levetop Electronics Co., Ltd.

10.3.2 Register Descriptions

10.3.2.1 WKUPC — Wakeup Configuration Register

Address: 0x0000

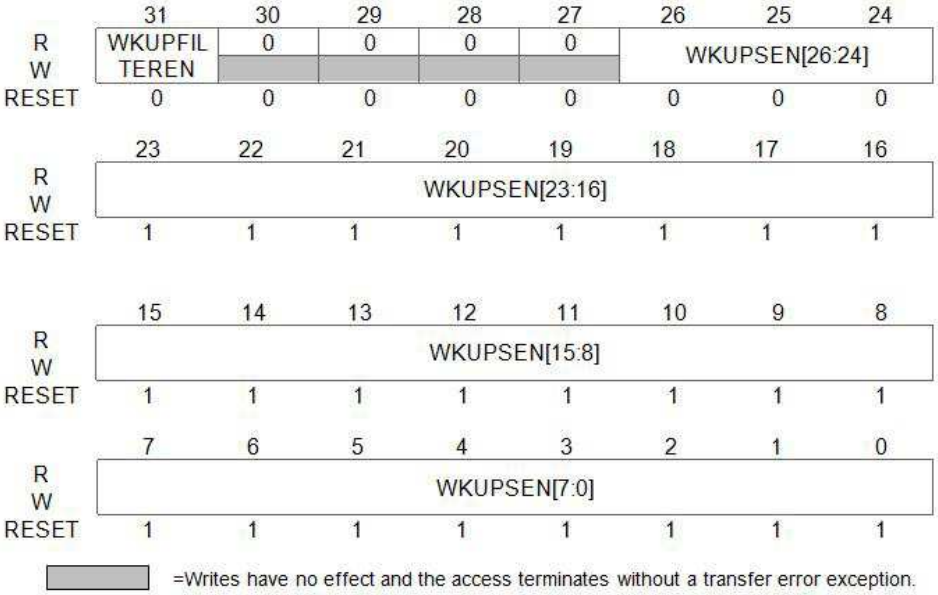


Figure 10–1: WKUPC — Wakeup Configuration Register

WKUPFILTEREN— Wakeup Source Filter Enable

If the WKUPFILTEREN is set, the wakeup source will pass a filter to be removed glitch and then to wakeup standby mode of the chip.

1 = Wakeup source filter enabled

0 = Wakeup source filter disabled

WKUPSEN— Wakeup Source Enable

This field control whether the corresponding source is used as a source to wakeup standby mode of the chip. If set, the corresponding source is used as wakeup source.

©Levetop Electronics Co., Ltd.

Table 10–2: WKUPSEN and The Corresponding Wakeup Source

WKUPSEN	Wakeup Source
WKUPSEN[26]	PVD interrupt
WKUPSEN[25]	USB resume
WKUPSEN[24]	COMP1 interrupt
WKUPSEN[23]	COMP0 interrupt
WKUPSEN[22]	WDT1 interrupt
WKUPSEN[21]	WDT1 reset
WKUPSEN[20]	i2c
WKUPSEN[19]	WDT0 interrupt
WKUPSEN[18]	JTAG POWERON REQEST
WKUPSEN[17]	RESETB pin
WKUPSEN[16]	WDT0 reset
WKUPSEN[15]	INT1[7]
WKUPSEN[14]	INT1[6]
WKUPSEN[13]	INT1[5]
WKUPSEN[12]	INT1[4]
WKUPSEN[11]	INT1[3]
WKUPSEN[10]	INT1[2]
WKUPSEN[9]	INT1[1]
WKUPSEN[8]	INT1[0]
WKUPSEN[7]	INT0[7]
WKUPSEN[6]	INT0[6]
WKUPSEN[5]	INT0[5]
WKUPSEN[4]	INT0[4]
WKUPSEN[3]	INT0[3]
WKUPSEN[2]	INT0[2]
WKUPSEN[1]	INT0[1]
WKUPSEN[0]	INT0[0]

©Levetop Electronics Co., Ltd.

10.3.2.2 CRPDC — Chip Reduce Pin Drive Configuration Register

Address: 0x0004

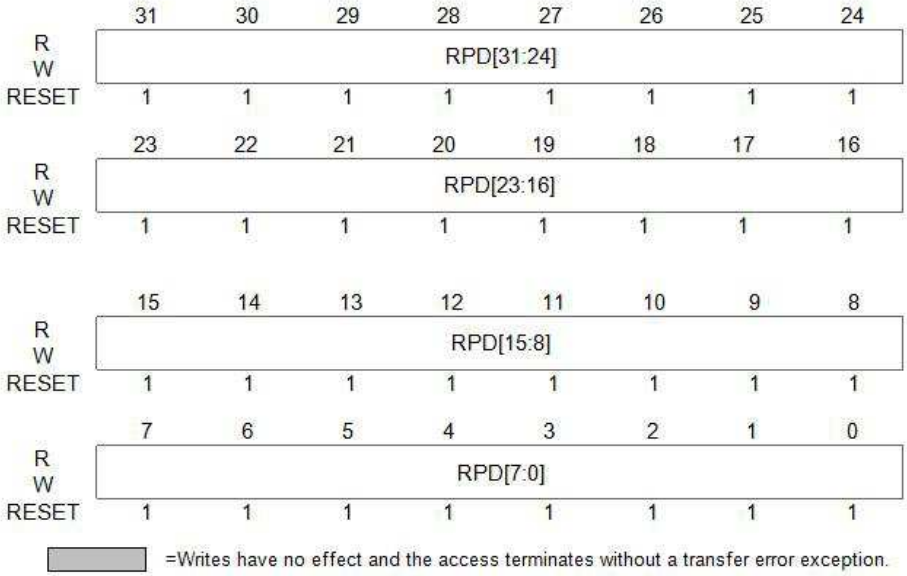


Figure 10–2: CPSC — Chip Reduce Pin Drive Configuration Register

RPD — Reduced Pin Drive Field

This read/write field controls the drive capability of the corresponding pin which shows in **Table 10–3**.

1 = Reduced pin drive
0 = Full pin drive

Table 10–3: Chip Reduce Pin Drive Configuration

Pin Name	Driver Strength Configuration Bit	Pin Driver Strength
MSS0, MSS1, SCK, MSDI0, MSDI1, MSDO0, MSDO1	RPD[31]	1: 6mA 0:12mA
RXD0,TXD0	RPD[30]	1: 6mA 0:12mA
Reversed	RPD[29]	1: 6mA 0:12mA
SCL, SDA	RPD[28]	1: 6mA 0:12mA

©Levetop Electronics Co., Ltd.

Table 10–3: Chip Reduce Pin Drive Configuration (Continued)

Pin Name	Driver Strength Configuration Bit	Pin Driver Strength
PWM0[3]	RPD[27]	0: 24mA 1:12mA
PWM0[2]	RPD[26]	0: 24mA 1:12mA
PWM0[1]	RPD[25]	0: 24mA 1:12mA
PWM0[0]	RPD[24]	0: 24mA 1:12mA
PWM1[3]	RPD[23]	0: 24mA 1:12mA
PWM1[2]	RPD[22]	0: 24mA 1:12mA
PWM1[1]	RPD[21]	0: 24mA 1:12mA
PWM1[0]	RPD[20]	0: 24mA 1:12mA
Reversed	RPD[19]	
Reversed	RPD[18]	
Reversed	RPD[17]	
Reversed	RPD[16]	
AIN[7]	RPD[15]	1: 6mA 0:12mA
AIN[6]	RPD[14]	1: 6mA 0:12mA
AIN[5]	RPD[13]	1: 6mA 0:12mA
AIN[4]	RPD[12]	1: 6mA 0:12mA
AIN[3]	RPD[11]	1: 6mA 0:12mA
EXTAL	RPD[10]	1: 6mA 0:12mA
INT0[1]	RPD[9]	1: 6mA 0:12mA
INT0[0]	RPD[8]	1: 6mA 0:12mA
XTAL	RPD[7]	1: 6mA 0:12mA
RSTOUT	RPD[6]	1: 6mA 0:12mA

©Levetop Electronics Co., Ltd.

Table 10–3: Chip Reduce Pin Drive Configuration (Continued)

Pin Name	Driver Strength Configuration Bit	Pin Driver Strength
INT1[5]	RPD[5]	1: 6mA 0:12mA
INT1[4]	RPD[4]	1: 6mA 0:12mA
INT1[3]	RPD[3]	1: 6mA 0:12mA
INT1[2]	RPD[2]	1: 6mA 0:12mA
INT1[1]	RPD[1]	1: 6mA 0:12mA
CLKOUT	RPD[0]	1: 6mA 0:12mA

10.3.2.3 CPSRC — Chip Pin Slow Rate Configuration Register

Address: 0x0008

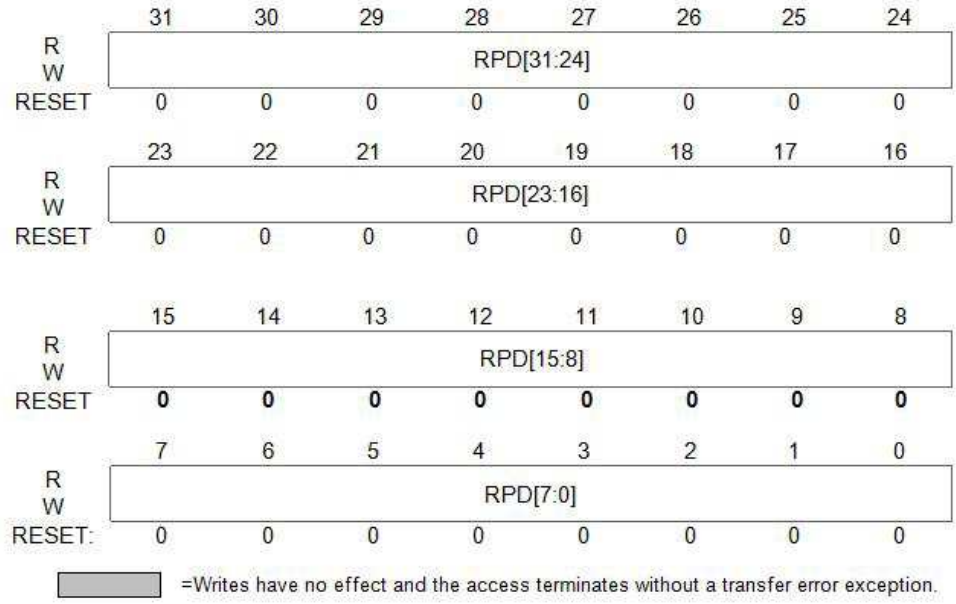


Figure 10–3: CPSRC — Chip Pin Slow Rate Configuration Register

PSRC — Pin Slow Rate Configuration Field
This read/write field controls the slow rate of the corresponding pin which shows in **Table 10–4**.

1 = Pin slow rate is limited
0 = Pin slow rate is normal

©Levetop Electronics Co., Ltd.

Table 10–4: Chip Pin Slow Rate Configuration

Pin Name	Slow Rate Configuration Bit	Slow Rate
MSS0, MSS1, SCK, MSDI0, MSDI1, MSDO0, MSDO1	PSRC[31]	0: Normal 1:Limited
RXD0, TXD0	PSRC[30]	0: Normal 1:Limited
Reversed	PSRC[29]	0: Normal 1:Limited
SCL, SDA	PSRC[28]	0: Normal 1:Limited
PWM0[3]	PSRC[27]	0: Normal 1:Limited
PWM0[2]	PSRC[26]	0: Normal 1:Limited
PWM0[1]	PSRC[25]	0: Normal 1:Limited
PWM0[0]	PSRC[24]	0: Normal 1:Limited
PWM1[3]	PSRC[23]	0: Normal 1:Limited
PWM1[2]	PSRC[22]	0: Normal 1:Limited
PWM1[1]	PSRC[21]	0: Normal 1:Limited
PWM1[0]	PSRC[20]	0: Normal 1:Limited
Reversed	PSRC[19]	0: Normal 1:Limited
Reversed	PSRC[18]	0: Normal 1:Limited
Reversed	PSRC[17]	0: Normal 1:Limited
Reversed	PSRC[16]	0: Normal 1:Limited
AIN[7]	PSRC[15]	0: Normal 1:Limited

©Levetop Electronics Co., Ltd.

Table 10–4 Chip Pin Slow Rate Configuration (Continued)

Pin Name	Slow Rate Configuration Bit	Slow Rate
AIN[6]	PSRC[14]	0: Normal 1:Limited
AIN[5]	PSRC[13]	0: Normal 1:Limited
AIN[4]	PSRC[12]	0: Normal 1:Limited
AIN[3]	PSRC[11]	0: Normal 1:Limited
EXTAL	PSRC[10]	0: Normal 1:Limited
INT0[1]	PSRC[9]	0: Normal 1:Limited
INT0[0]	PSRC[8]	0: Normal 1:Limited
XTAL	PSRC[7]	0: Normal 1:Limited
RSTOUT	PSRC[6]	0: Normal 1:Limited
INT1[5]	PSRC[5]	0: Normal 1:Limited
INT1[4]	PSRC[4]	0: Normal 1:Limited
INT1[3]	PSRC[3]	0: Normal 1:Limited
INT1[2]	PSRC[2]	0: Normal 1:Limited
INT1[1]	PSRC[1]	0: Normal 1:Limited
CLKOUT	PSRC[0]	0: Normal 1:Limited

©Levetop Electronics Co., Ltd.

10.3.2.4 CPPDC — Chip Pin Pull Down Configuration Register

Address: 0x000c

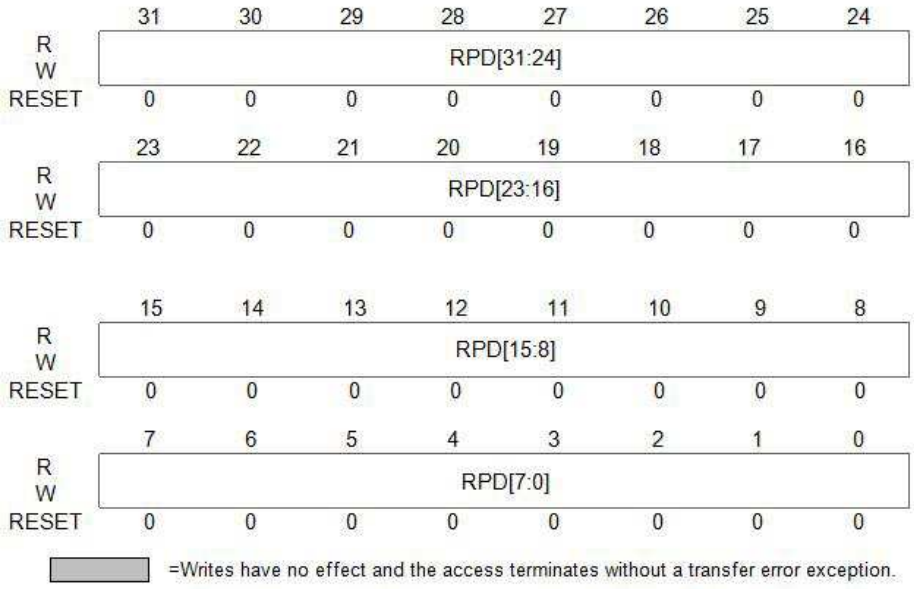


Figure 10–4: CPPDC — Chip Pin Pull Down Configuration Register

PPDC — Pin Pull Down Configuration Field

This read/write field controls the pull down function of the corresponding pin which shows in **Table 10–5.**

1 = Pull down function of the corresponding pin is enabled

0 = Pull down function of the corresponding pin is disabled

Table 10–5: Chip Pin Pull Down Configuration

Pin Name	Pull Down Configuration Bit	Pull Down Function
MSS0, MSS1, SCK, MSDI0, MSDI1, MSDO0, MSDO1	PPDC[31]	0: Disable 1:Enable
RXD0, TXD0	PPDC[30]	0: Disable 1:Enable
Reversed	PPDC[29]	0: Disable 1:Enable
SCL, SDA	PPDC[28]	0: Disable 1:Enable
PWM0[3]	PPDC[27]	0: Disable 1:Enable
PWM0[2]	PPDC[26]	0: Disable 1:Enable
PWM0[1]	PPDC[25]	0: Disable 1:Enable
PWM0[0]	PPDC[24]	0: Disable 1:Enable

©Levetop Electronics Co., Ltd.

Table 10–5: Chip Pin Pull Down Configuration (Continued)

Pin Name	Pull Down Configuration Bit	Pull Down Function
PWM1[3]	PPDC[23]	0: Disable 1:Enable
PWM1[2]	PPDC[22]	0: Disable 1:Enable
PWM1[1]	PPDC[21]	0: Disable 1:Enable
PWM1[0]	PPDC[20]	0: Disable 1:Enable
Reversed	PPDC[19]	0: Disable 1:Enable
Reversed	PPDC[18]	0: Disable 1:Enable
Reversed	PPDC[17]	0: Disable 1:Enable
Reversed	PPDC[16]	0: Disable 1:Enable
AIN[7]	PPDC[15]	0: Disable 1:Enable
AIN[6]	PPDC[14]	0: Disable 1:Enable
AIN[5]	PPDC[13]	0: Disable 1:Enable
AIN[4]	PPDC[12]	0: Disable 1:Enable
AIN[3]	PPDC[11]	0: Disable 1:Enable
EXTAL	PPDC[10]	0: Disable 1:Enable
INT0[1]	PPDC[9]	0: Disable 1:Enable
INT0[0]	PPDC[8]	0: Disable 1:Enable
XTAL	PPDC[7]	0: Disable 1:Enable
RSTOUT	PPDC[6]	0: Disable 1:Enable
INT1[5]	PPDC[5]	0: Disable 1:Enable
INT1[4]	PPDC[4]	0: Disable 1:Enable
INT1[3]	PPDC[3]	0: Disable 1:Enable
INT1[2]	PPDC[2]	0: Disable 1:Enable
INT1[1]	PPDC[1]	0: Disable 1:Enable
CLKOUT	PPDC[0]	0: Disable 1:Enable

©Levetop Electronics Co., Ltd.

10.3.2.5 CTR — Chip Test Register

The chip test register is reserved for factory testing.

Note:
To safeguard against unintentionally activating test logic, write $0000 to lock out test features. Setting any bits in the CTR register may lead to unpredictable results.

Address: 0x0010

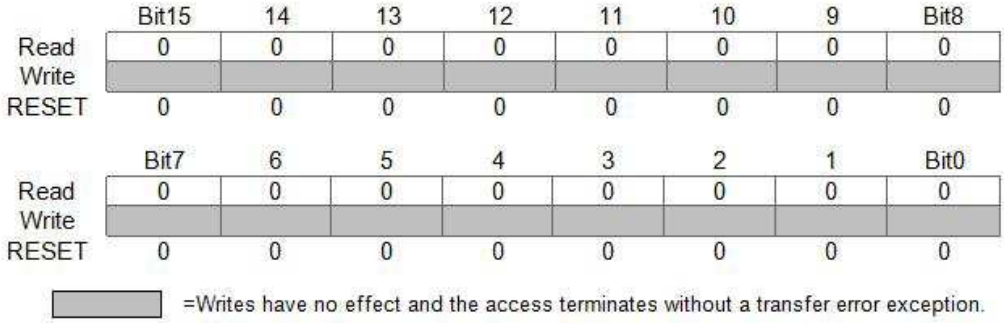


Figure 10–5: CTR — Chip Test Register

10.3.2.6 CIR — Chip Identification Register

The CIR register is a read-only register; writing to CIR has no effect. The chip's ID and Revision number can be read in this register.

Address: 0x0012

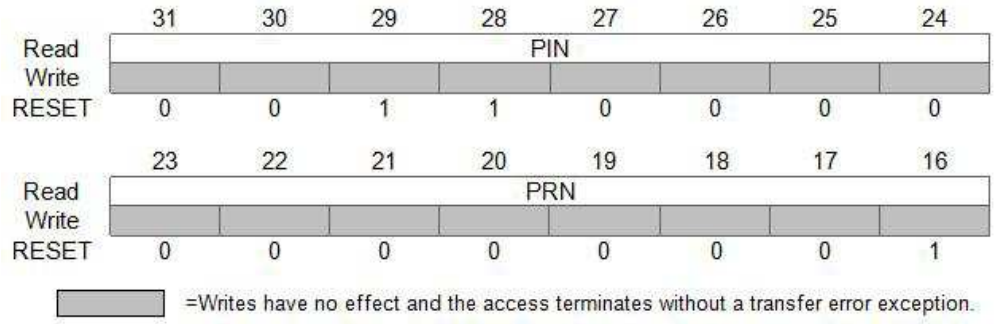


Figure 10–6: CIR — Chip Identification Register

PIN[7:0] — Part Identification Number Field
This read-only field contains a unique version identification number for the chip.

PRN[7:0] — Part Revision Number Field
This read-only field contains the full-layer mask revision number. This number is increased by one for each new full-layer mask set of this part. The revision numbers are assigned in chronological order.

©Levetop Electronics Co., Ltd.

11. Clock and Power Control Module

11.1 Overview

The clock module contains:

- FIRC: Internal high speed 144MHZ oscillator
- FXOSC: External Fast Speed crystal oscillator(12MHZ for USB Application and others can be range from 0 to 20MHZ)
- SIRC: Internal low speed 128Khz oscillator
- SXOSC: External Low Speed crystal oscillator(32768HZ)
- Status and control registers
- Clock and Power Control logic

11.2 Features

Features of the clock module include:

- Two system clock sources
  - Internal high speed 144MHz oscillator for cache memory and divided-by-two for system –
  - External Fast Speed crystal
- Individual clock divider for IPS, system and ADC clock
- Support for low-power mode
- Modules can be separately stopped by setting MSCR

11.3 Clock Structure

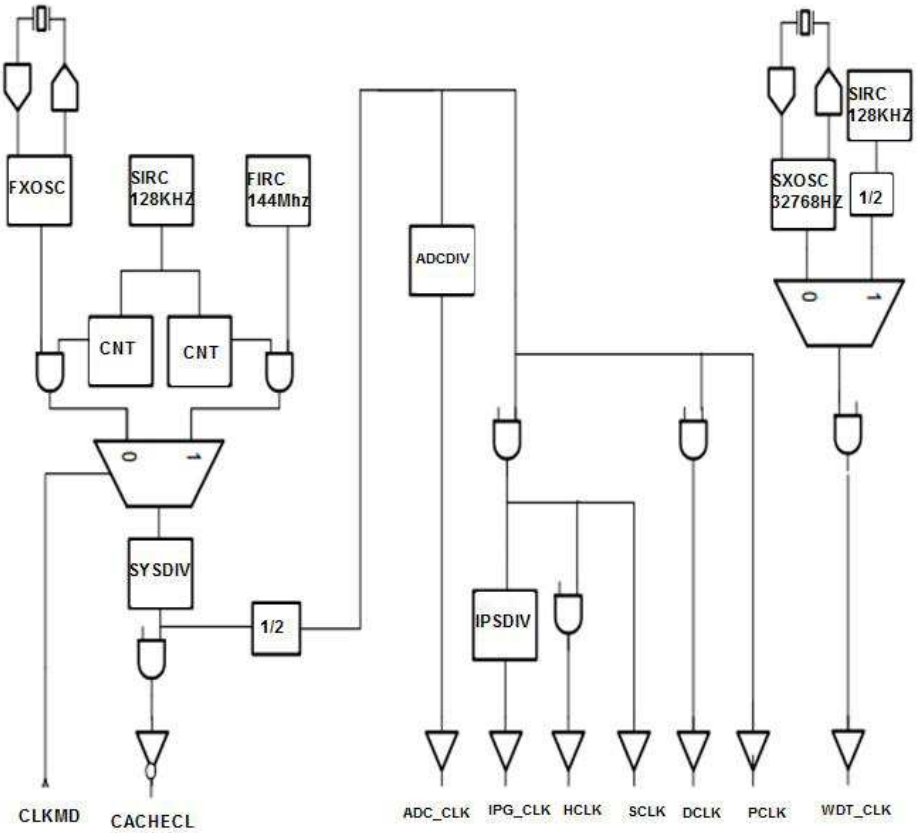


Figure 11–1: Clock Structure

©Levetop Electronics Co., Ltd.

11.4 Clock Source Select

System Clock Source can be internal high speed 144MHZ oscillator or external 12MHZ crystal/resonator. Clock source Select is depended on the CLKMD bit of CCR register in Option Byte module. If it is set then internal high speed 144MHZ oscillator is the system clock source, otherwise the system clock source is external 12MHZ crystal/resonator.

11.4.1 Low-Power Options

11.4.1.1 Wait and Doze Modes

In wait and doze modes, the system clocks to the peripherals and embedded-flash are enabled, the clocks to the CPU, ROM, and SRAM are stopped. Each module can disable the module clocks locally at the module level or by setting MSCR.

11.4.1.2 Stop Mode

In stop mode, all system clocks are disabled.

CAUTION:

Don’t program or erase EFLASH during stop mode.

©Levetop Electronics Co., Ltd.

11.5 Memory Map and Registers

The clock programming model consists of these registers:

- Synthesizer Control Register (SYNCR)
- Low Speed Oscillator Control Register (LOSCCR)
- Internal High Speed Oscillator Control and Status Register (IOSCCSR)
- Module stop control register (MSCR)

11.5.1 Module Memory Map (Base: 0x4003_0000)

Table 11–1: Clock Memory Map

Address	31:16	15:0	Access[1]
0x0000	Synthesizer Control Register (SYNCR)		S
0x0004	Low Speed Oscillator Control Register (LOSCCR)	Internal High Speed Oscillator Control and Status Register (IOSCCSR)	S
0x0008			S
0x000c	Module Stop Control Register (MSCR)		S

Note:

1. S = supervisor-only access. User mode accesses to supervisor only address locations have no effect and result in a cycle termination transfer error.

©Levetop Electronics Co., Ltd.

11.5.2 Register Description

This subsection provides a description of the clock module registers.

11.5.2.1 Synthesizer Control Register

The synthesizer control register (SYNCR) is read/write always.

Register Offset Address: 0x0000

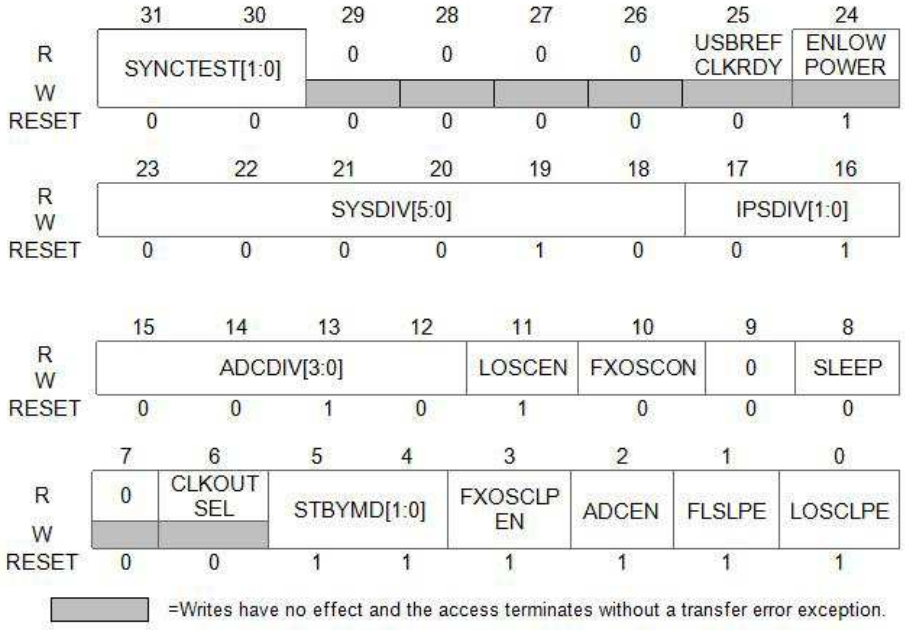


Figure 11–2: Synthesizer Control Register (SYNCR)

SYNCTEST[1:0] —SYNCR Write Access Sequence In

The writable bit of SYNCR register cannot be changed, unless the correct sequence writes in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of SYNCR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

SYSDIV[5:0] — System Clock Divider

This field sets the divide value for system clock. The default value is 6'b000000 (divide by one). The other divide value is referenced to

©Levetop Electronics Co., Ltd.

Table 11–2: System Clock Divider

SYSDIV[5:0]	Divide Value
000000	Divide-by-1
000001	Divide-by-2
000010	Divide-by-4
000011	Divide-by-6
000100	Divide-by-8
000101	Divide-by-10
…	…
…	…
111111	Divide-by-126

IPSDIV[1:0] — IPS Clock Divider

This field sets the divide value for IPS clock. The default value is 2'b01 (divide by two). The other divide value is referenced to **Table 11–3**.

Table 11–3: IPS Clock Divider

IPSDIV[1:0]	Divide Value
00	Divide-by-1
01	Divide-by-2
10	Divide-by-3
11	Divide-by-4

LOSCEN — Internal Low Speed 128KHZ Oscillator Enable Bit

1 = Internal Low Speed 128KHZ Oscillator is enabled

0 = Internal Low Speed 128KHZ Oscillator is disabled

FXOSCON — FXOSC Enable Bit When CKMD is set. When CKMD is set, FXOSC will be turn off if FXOSCON is clear otherwise FXOSC will be turned-on for USB reference clock source. When CKMD is clear, FXOSC will be turn-on and this bit is invalid.

1 = FXOSC will be turned on when CKMD is set

0 = FXOSC will be turned off when CKMD is set

SLEEP — Chip Sleep Mode Control Bit

Set the SLEEP bit, the chip will enter standby mode. The operation is the same as "stop" instruction.

©Levetop Electronics Co., Ltd.

ADCDIV[3:0] — ADC Clock Divider

This field sets the divide value for ADC clock. The default value is 4'b0000 (divide by one). The other divide value is referenced to **Table 11–4.**

Table 11–4: ADC Clock Divider

ADCDIV[3:0]	Divide Value
0000	Divide-by-1
0001	Divide-by-2
0010	Divide-by-3
0011	Divide-by-4
0100	Divide-by-5
0101	Divide-by-6
0110	Divide-by-7
0111	Divide-by-8
1000	Divide-by-9
1001	Divide-by-10
1010	Divide-by-11
1011	Divide-by-12
ADCDIV[3:0]	Divide Value
1100	Divide-by-13
1101	Divide-by-14
1110	Divide-by-15
1111	Divide-by-16

CLKOUTSEL — Clock Out Select Bit

Table 11–5: CLKOUTSEL Mode

CLKOUTSEL	CLKOUT
0	System clock
1	128KHZ clock

©Levetop Electronics Co., Ltd.

STBYMD[1:0] — Standby Operation Control Bits

STBYMD[1:0] control clock source, system clock operation and LDO State in stop mode as shown in **Table 11–6.**

Table 11–6: Standby Operation Control Bit in Standby Mode

	Operation During Standby Mode			
STBYMD	**ADC Clock**	**System Clocks**	**Clock Source**	**LDO**
00	Enable	Disabled	Enable	Normal
01	Disabled	Disabled	Enable	Normal
10	Disabled	Disabled	Disabled	Normal
11	Disabled	Disabled	Disabled	Standby

FXOSCLPEN — FXOSC Low Power Enable bit when CKMD is set. If this bit is set, FXOSC will be stopped during standby mode otherwise FXOSC will be still running for USB reference clock. This bit is valid only when CKMD is set and FXOSC will be turned off when STBYMD[1] is set during standby mode(by setting SLEEP bit or stop instruction).

1 = Low power mode of FXOSC is enabled when CKMD is set
0 = Low power mode of FXOSC is disabled when CKMD is set

FLSLPE —Flash Low Power Enable

If the FLSLPE is set, EFLASH will be enter standby mode during the chip enters standby mode.

1 = Low power mode of EFLASH is enabled
0 = Low power mode of EFLASH is disabled

LOSCLPE — Internal Low Speed 128KHZ Oscillator Low Power Enable

If the LOSCLPE is set, the internal low speed 128KHZ oscillator will be stop during the chip enters standby mode.

1 = Low power mode of Internal Low Speed 128KHZ Oscillator is enabled
0 = Low power mode of Internal Low Speed 128KHZ Oscillator is disabled

ADCEN — Analog-to-digital converter Clock Enable Bit

1 = ADC Clock enable
0 = ADC Clock disable

ENLOWPOWER — Enable Enter Low power mode status Bit

Before system enter standby mode, be sure that this bit is set, otherwise the low power mode won't be entered successfully. This bit is set after recovery from low power mode, and cleared by hardware when enter low power mode

1 = Enable Enter Low power mode
0 = Enter Low power mode is not allowed

USBREFCLKRDY — USB PLL reference oscillator (FXOSC) ready flag

0 = Not ready
1 = Ready

This is USB PLL reference oscillator (refer to FXOSC) ready flag which indicates when USB PLL reference oscillator is stable. This flag is cleared to "0" by hardware when FXOSC is powered on or turned on by setting FXOSCON (when CKMD is set) and then changes to a high level after FXOSC oscillator is stable. Therefore this flag will always be read as "1" by the application program after FXOSC is turn-on. The flag will be low when in the STANDBY mode, STBYMD is 2'b10 and FXOSCLPEN is set , but after a wake-up has occurred, the flag will change to a high level after 4096 SIRC clock cycles.

©Levetop Electronics Co., Ltd.

11.5.2.2 Low Speed Oscillator Control and Status Register

Register Offset Address: 0x0004

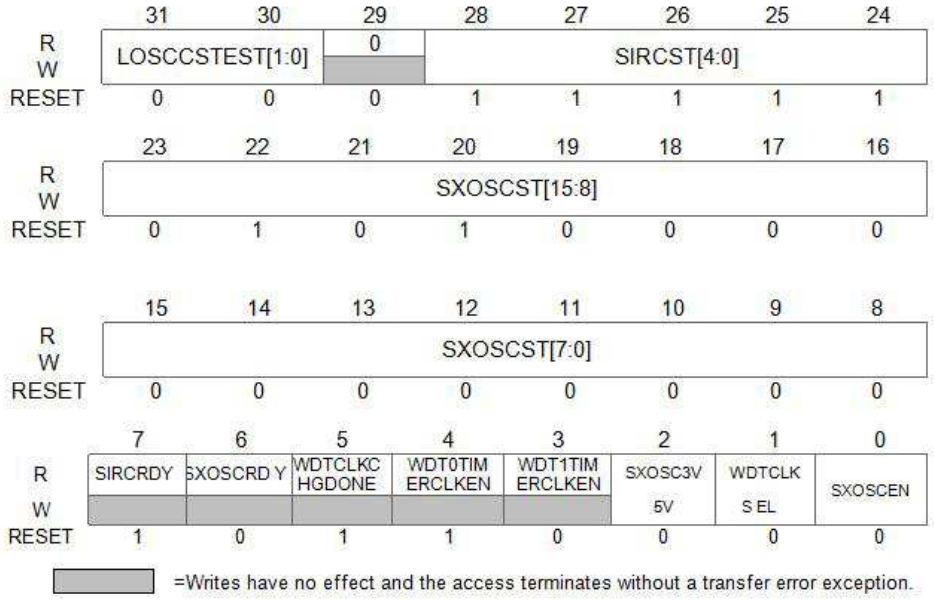


Figure 11–3: Low Speed Oscillator Control and Status Register (LOSCCSR)

LOSCCSTEST[1:0] —LOSCCSR Write Access Sequence In

The writable bit of IOSCCSR register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of LOSCCSR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

LOSCST[4:0] — Internal Low Speed Oscillator Stable Time Value

The internal low speed oscillator (SIRC) will wait SIRCST[4:0] cycles of 128KHZ(source of SIRC) oscillator and then ready for output clock after switch on.

SXOSCST[15:0] — External Low Speed Oscillator Stable Time Value

The external low speed oscillator (SXOSC) will wait SXOSCST[15:0] cycles of 32KHZ(source of SIRC) oscillator and then ready for output clock after switch on.

SXOSCEN — SXOSC crystal control bit

1 = SXOSC crystal is enabled

0 = SXOSC crystal is disabled

SXOSC Enabled	SXOSC Disabled
32K_XO	INT1[2]
32K_XI	INT1[3]

SXOSC3V5V — SXOSC 3.3V or 5V run mode control bit

If the main power of the chip is 5V power supply, then this bit should be set to 0, otherwise if the main power of the chip is 3.3V power supply, then this bit should be set to 1

©Levetop Electronics Co., Ltd.

WDTCLKSEL — WDT clock selection control bit
It is recommend that you should clear the bit first then turn off SXOSC when change WDT clock source from SXOSC to SIRC.
1 = WDT clock source is SXOSC (32.768HZ)
0 = WDT clock source is SIRC32Khz (which is generated from SIRC128Khz divided by 4)

SIRCRDY— Internal low speed oscillator (SIRC) ready flag.
1 = Internal low speed oscillator (SIRC) is ready
0 = Internal low speed oscillator (SIRC) is not ready

SXOSCRDY —External low speed oscillator (SXOSC) ready flag.
1 = External low speed oscillator (SXOSC) is ready
0 = External low speed oscillator (SXOSC) is not ready

WDTCLKCHGDONE —WDT clock switch done flag. This bit will be change to low when WDTCLKSEL is changed, when WDTCLK change done, then this bit will be set.
1 = WDT clock switch done and WDT can work normally
0 = WDT clock is switching and WDT cannot be work normally

WDT0TIMERCLKEN —WDT0 timer clock enable control bit.
When this bit is clear, the WDT0 timer counter is frozen, otherwise if this bit is set, WDT0 timer counter will be run freely.

WDT1TIMERCLKEN —WDT1 timer clock enable control bit.
When this bit is clear, the WDT1 timer counter is frozen, otherwise if this bit is set, WDT1 timer counter will be run freely.

©Levetop Electronics Co., Ltd.

11.5.2.3 Module Stop Control Register

The Module Stop Control Register (MSCR) is read/write always.

Register Offset Address: 0x000c

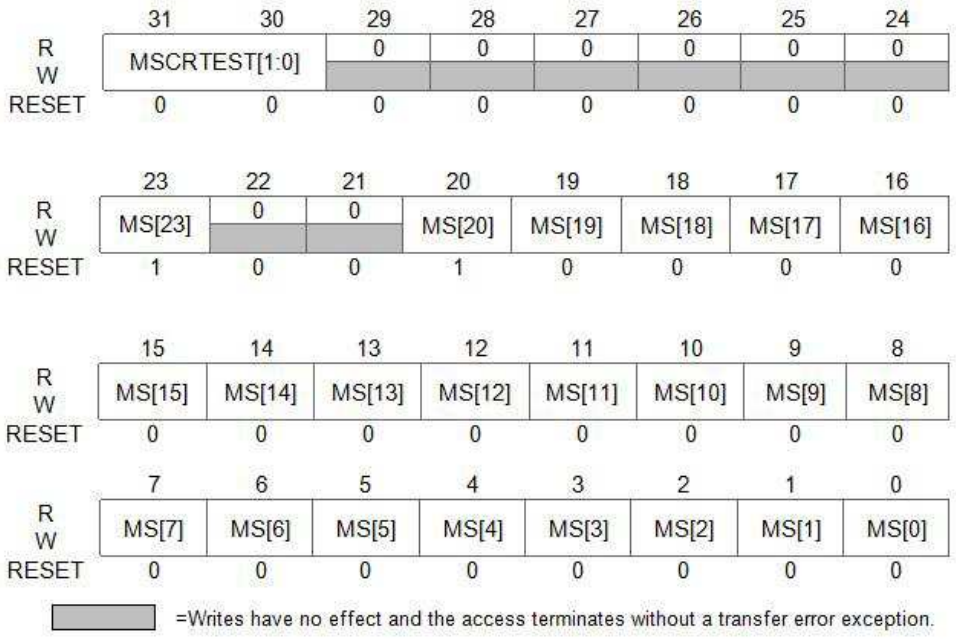


Figure 11–4: Module Stop Control Register (MSCR)

MSCRTEST[1:0] —MSCR Write Access Sequence In

The writable bit of MSCR register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of MSCR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

©Levetop Electronics Co., Ltd.

MS[20:0] — Module Stop Bits

The MS[20:0] bits disable the modules' clocks in the top level. (see **Table 11–7 MS[15:0] Bits Corresponding Modules**).

1 = Module Clock Disabled

0 = Module Clock Enabled

Table 11–7: MS[15:0] Bits Corresponding Modules

MS Bit	Corresponding Module
0	CPM
1	COMP0
2	COMP1
3	ADC
4	PIT0
5	PIT1
6	PIT2
7	PIT3
8	WDT1
9	SCM
10	PWM0
11	PWM1
12	EPORT0
13	EPORT1
14	LDMAC
15	EFM
16	RESET
17	WDT0
18	SCI0
19	CCM
20	I2C
23	USBC

©Levetop Electronics Co., Ltd.

11.5.2.4 EPT External Clock Source Enable Control Register

Register address: 0x0010

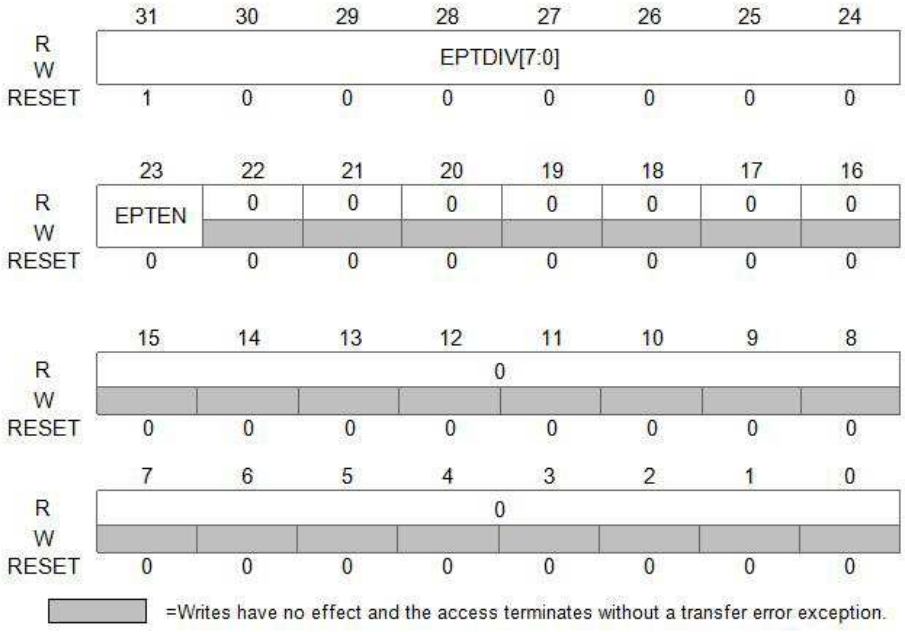


Figure 11–5: EPT External Clock Source Enable Control Register (ECSECR)

EPTDIV[5:0] — EPT Clock Divider

This field sets the divide value for EPT clock. The default value is 8'h80. The other divide value is referenced to **Table 11–8.**

Table 11–8: EPT Clock Divider

EPTDIV[8:0]	Divide Value
00000000	Divide-by-1
00000001	Divide-by-2
00000010	Divide-by-3
00000011	Divide-by-4
00000100	Divide-by-5
00000101	Divide-by-6
…	…
…	…
11111110	Divide-by-255
11111111	Divide-by-256

EPTEN — EPT Clock Enable Bit

If the EPTEN bit is set, EPT clock will be divided from system clock.

1 = EPT clock enable

0 = EPT clock disable

©Levetop Electronics Co., Ltd.

11.5.2.5 OSC Bist Test Configuration Register1

Register address: 0x0014

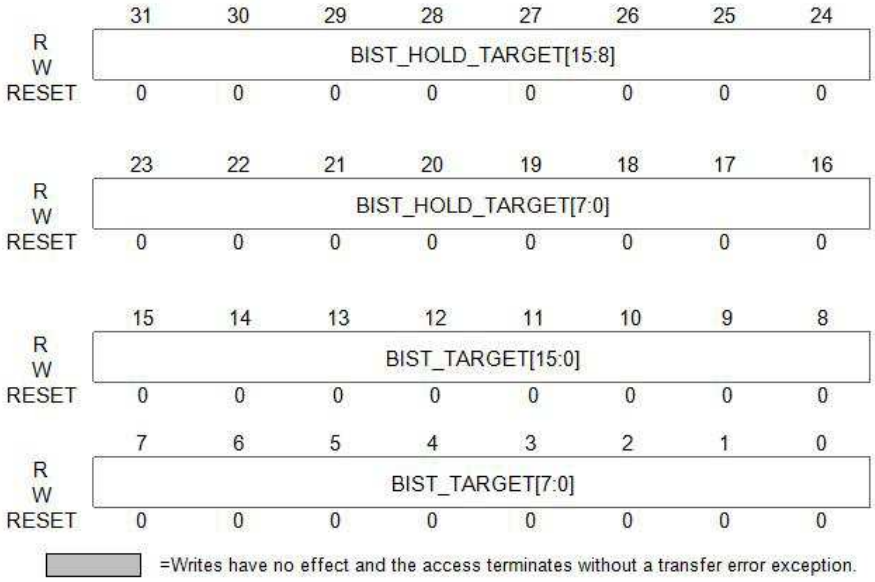


Figure 11–6: OSC Bist Test Configuration Register1 (OBTCR1)

BIST_HOLD_TARGET — BIST CLK HOLD COUNT TARGET VALUE
WAIT CLK UNDER TEST STABLE AFTER TRIM VALUE CHANGE

BIST_TARGET — BIST CLK COUNT TARGET VALUE

©Levetop Electronics Co., Ltd.

11.5.2.6 OSC Bist Test Configuration Register2

Register address: 0x0018

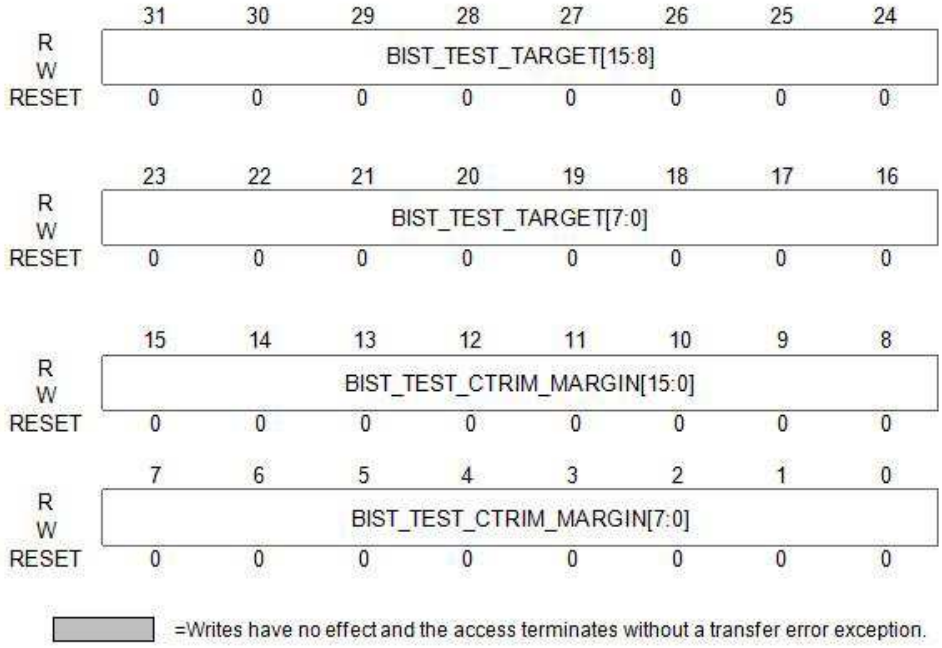


Figure 11–7: OSC Bist Test Configuration Register2 (OBTCR2)

BIST_TEST_TARGET — UNDER TEST CLK COUNT TARGET VALUE

BIST_TEST_CTRIM_MARGIN — UNDER TEST CLK COUNT ERROR MARGIN FOR COARSE TRIM

©Levetop Electronics Co., Ltd.

11.5.2.7 OSC Bist Test Control Register

Register address: 0x001c

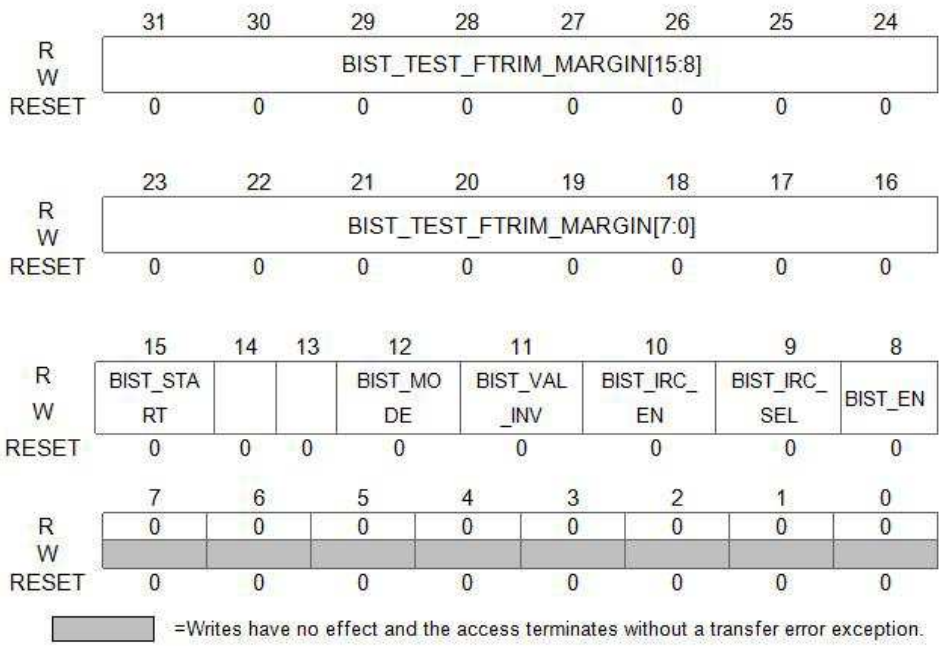


Figure 11–8: OSC Bist Test Control Register (OBTCR)

BIST_TEST_FTRIM_MARGIN — Under test CLK count error margin for fine trim.

BIST_MODE — BIST MODE
1 = trace mode
0 = trim mode

BIST_VAL_INV — BIST trim bit inverse
1 = inverse
0 = normal

BIST_IRC_SEL — FIRC or SIRC bist selection
1 = FIRC bist
0 = SIRC bist

BIST_EN — Under test CLK enable
1 = enable
0 = disable

©Levetop Electronics Co., Ltd.

11.5.2.8 OSC BIST Test Counter Register

Register address: 0x0020

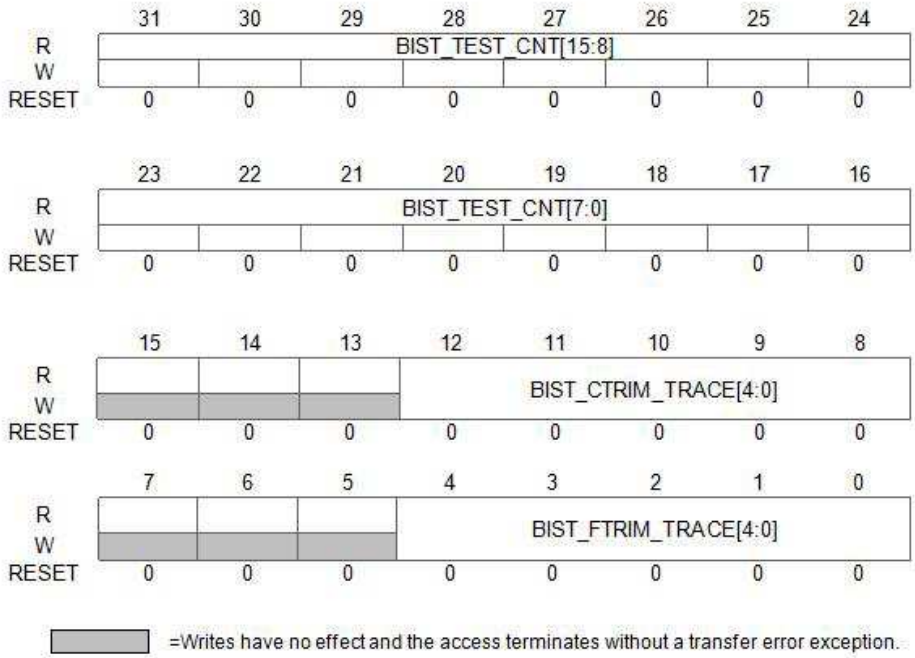


Figure 11–9: OSC BIST Test Counter Register (OBTCNTR)

BIST_CTRIM_TRACE — BIST CTRIM TRACE VALUE

BIST_FTRIM_TRACE — BIST FTRIM TRACE VALUE

BIST_TEST_CNT — BIST TEST COUNTER VALUE

©Levetop Electronics Co., Ltd.

11.5.2.9 OSC BIST Test Result Register

Register address: 0x0024

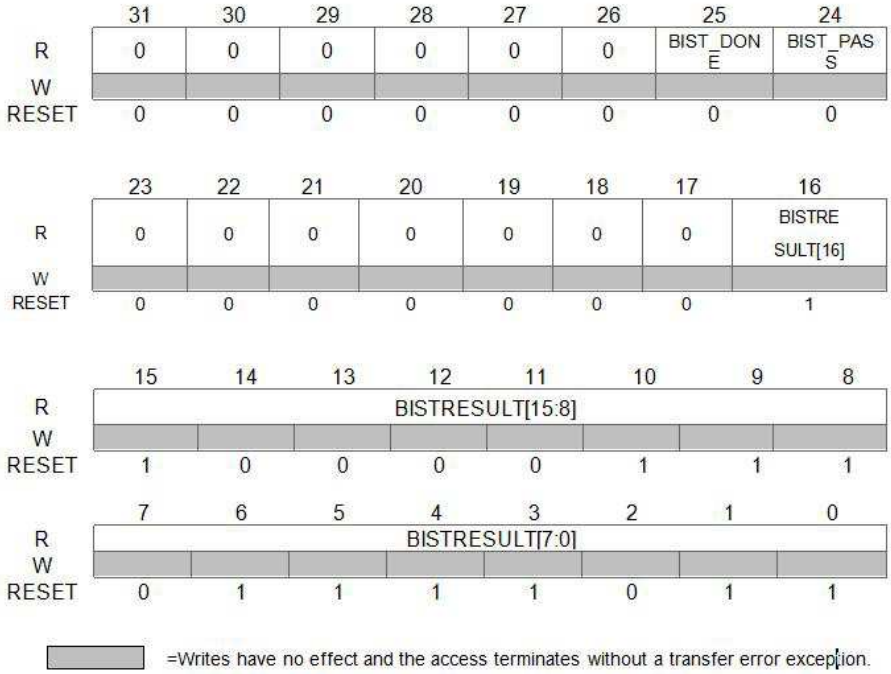


Figure 11–10: EPT External Clock Source Enable Control Register (ECSECR)

BIST_DONE — BIST DONE
1 = done
0 = not done

BISTRESULT — BIST TRIM RESULT

Valid when bist_done = 1 & bist_pass = 1

©Levetop Electronics Co., Ltd.

12. Reset Controller Module

12.1 Overview

The reset controller is provided to determine the cause of reset, assert the appropriate reset signals to the system, and then to keep a history of what caused the reset.

12.2 Features

Module features include:

- Four sources of reset:
  - Power on reset
  - External reset pin
  - Software reset
  - Watchdog Timer Reset
  - Programmable Voltage Detect Reset
- Software-readable status flags indicating the cause of the last reset

12.3 Block Diagram

Figure 12–1 illustrates the reset controller.

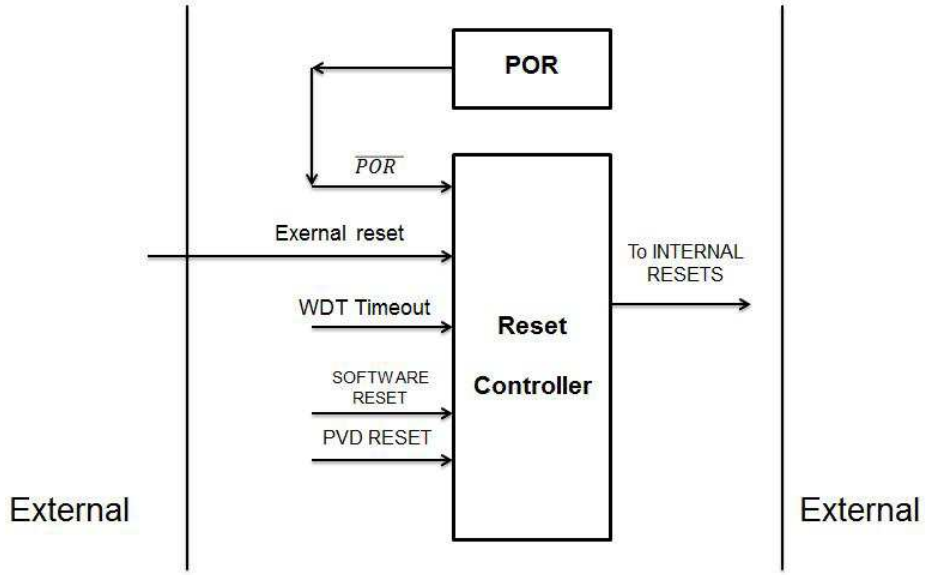


Figure 12–1: Reset Controller Block Diagram

©Levetop Electronics Co., Ltd.

12.4 Memory Map and Registers (Base: 0x4002_0000)

The reset controller programming model consists of these registers:

- Reset Control Register (RCR) Selects reset controller functions
- Reset Status Register (RSR) reflects the state of the last reset source

See **Table 12–1** for the address map and the following paragraphs for a description of the registers.

Table 12–1: Reset Controller Address Map

Address	Bit 7:0	Access [1]
0x0000	Reversed	S/U
0x0001	RTR—Reset Test Register	S/U
0x0002	RSR—Reset Status Register	S/U
0x0003	RCR—Reset Control Register	S/U

Note:

1. S/U = supervisor or user mode access.

12.4.1 Reset Test Register

The Reset Test Register (RTR) is only for factory testing.

Address: 0x0001

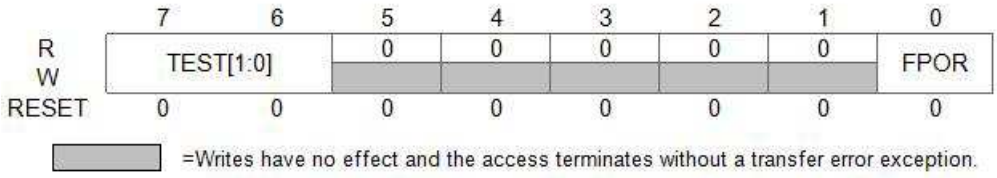


Figure 12–2: Reset Test Register (RTR)

FPOR — Force Power On Reset

Write 0x5b to the RTR register then set the bit will result in system power on reset. The reset will result in the chip trimming again.

©Levetop Electronics Co., Ltd.

12.4.2 Reset Status Register

The Reset Status Register (RSR) contains a status bit for every reset source. When reset is entered, the cause of the reset condition is latched along with a value of 0 for the other reset sources that were not pending at the time of the reset condition. These values are then reflected in RSR. One or more status bits may be set at the same time. The cause of any subsequent reset is also recorded in the register, overwriting status from the previous reset condition.

RSR can be read at any time. Writing to RSR has no effect.

Address: 0x0002

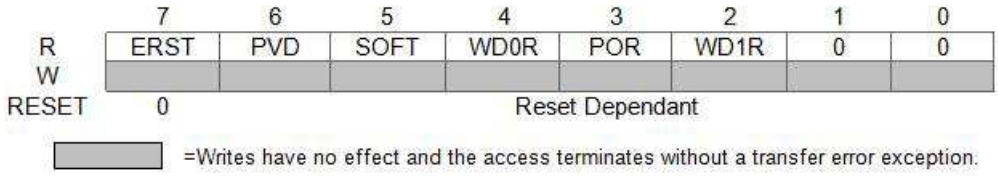


Figure 12–3: Reset Status Register (RSR)

ERST — External Reset
This bit indicates that the last reset state was caused by an external reset.
1 = Last reset state was caused by an external reset
0 = Last reset state was not caused by an external reset

PVD — Programmable Voltage Detect
This bit indicates that the last reset state was caused by a PVD reset.
1 = Last reset state was caused by an PVD reset
0 = Last reset state was not caused by an PVD reset

SOFT — Software Reset Flag
This bit indicates that the last reset state was caused by software.
1 = Last reset state was caused by software.
0 = Last reset state was not caused by software.

WD0R — Watchdog Timer 0 Reset Flag
This bit indicates that the last reset state was caused by a watchdog timer 0 timeout.
1 = Last reset state was caused by watchdog timer 0 timeout.
0 = Last reset state was not caused by watchdog timer 0 timeout.

POR — Power-On Reset Flag
This bit indicates that the last reset state was caused by power-on reset.
1 = Last reset state was caused by power-on reset.
0 = Last reset state was not caused by power-on reset

WD1R — Watchdog Timer 1 Reset Flag
This bit indicates that the last reset state was caused by a watchdog timer 1 timeout.
1 = Last reset state was caused by watchdog timer 1 timeout.
0 = Last reset state was not caused by watchdog timer 1 timeout

©Levetop Electronics Co., Ltd.

12.4.3 Reset Control Register

The Reset Control Register (RCR) allows software control for requesting a reset.

Address: 0x0003

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	FRCR
W								STOUT
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 12–4: Reset Control Register (RCR)

FRCRSTOUT — Force $\overline{\text{RSTOUT}}$ Pin

The FRCRSTOUT bit allows software to assert or negate the external $\overline{\text{RSTOUT}}$ pin.

1 = Assert $\overline{\text{RSTOUT}}$ pin

0 = Negate $\overline{\text{RSTOUT}}$ pin

©Levetop Electronics Co., Ltd.

12.5 Functional Description

12.5.1 Reset Sources

Table 12–2 defines the sources of reset and the signals driven by the reset controller.

Table 12–2: Reset Source Summary

Source	Type
POR	Asynchronous
ERST	Asynchronous
Watchdog timer	Asynchronous
Software	Synchronous
PVD	Asynchronous

To protect data integrity, a synchronous reset source is not acted upon by the reset control logic until the end of the current bus cycle. Reset is then asserted on the next rising edge of the system clock after the cycle is terminated. Whenever the reset control logic must synchronize reset to the end of the bus cycle, the internal bus monitor is automatically enabled.

Asynchronous reset sources usually indicate a catastrophic failure. Therefore, the reset control logic does not wait for the current bus cycle to complete. Reset is asserted immediately to the system.

12.5.1.1 Power-On Reset ($\overline{\text{POR}}$)

At power up, the reset controller asserts system reset. System reset continues to be asserted until $\overline{\text{POR}}$ has reached a minimum acceptable level.

12.5.1.2 Watchdog Timer Reset

A watchdog timer timeout causes timer reset request to be recognized and latched.

12.5.1.3 Software Reset

If the SYSRESTEQ bit in C0 Embedded Interrupt Controller (EIC) is set, the software reset will be generated. The reset controller asserts system reset for approximately 2048 cycles. Then the part exits reset and resume operation.

12.5.1.4 Programmable Voltage Detect Reset

When the PVDRE bit of the CCR register in Embedded Flash Module (EFM) is set, PVD will generate reset when the VDD exceeds the PVD threshold.

©Levetop Electronics Co., Ltd.

12.5.2 Reset Control Flow

The reset logic control flow is shown in **Figure 12–5**. All cycle counts given are approximate.

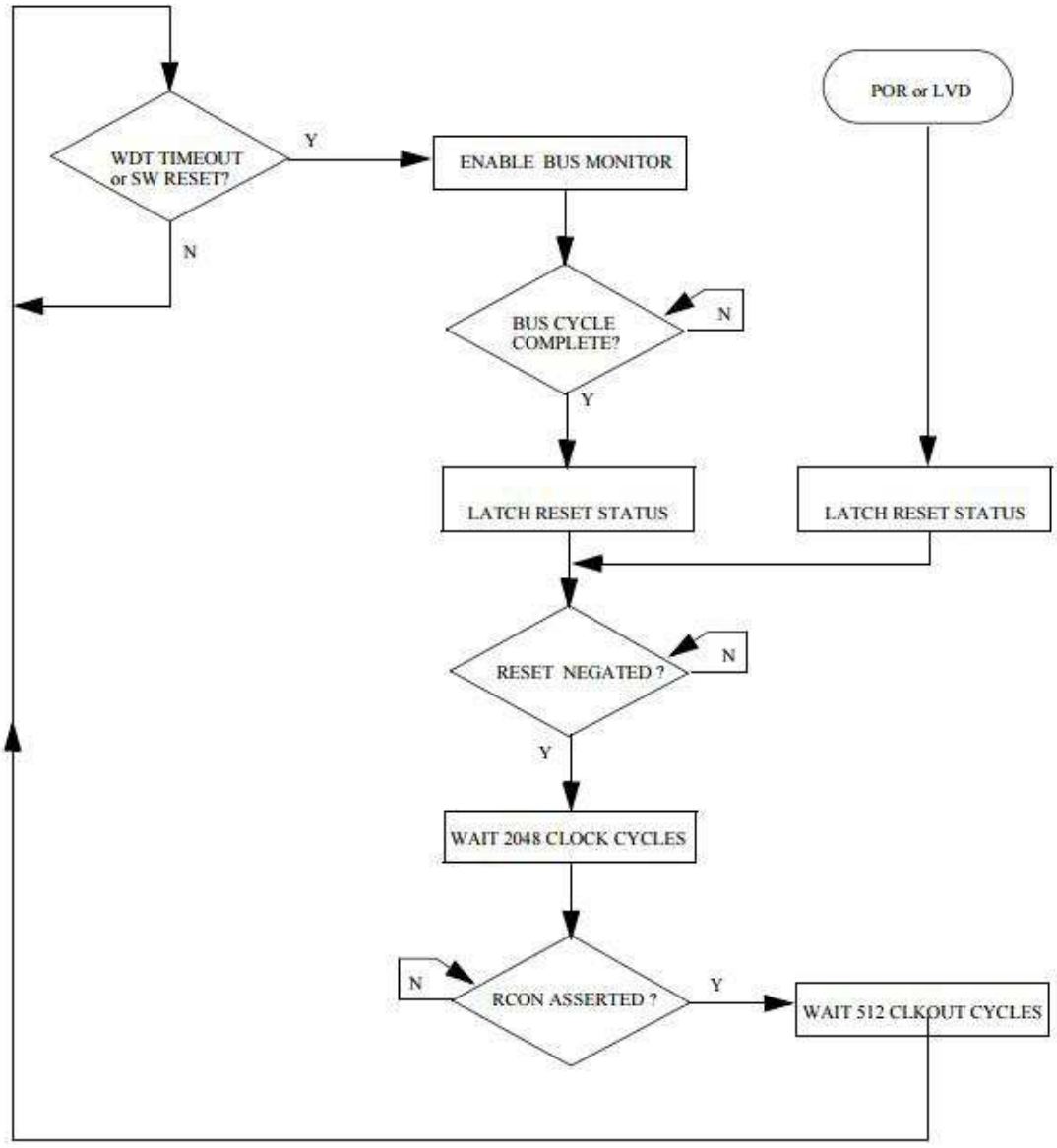


Figure 12–5: Reset Control Flow

©Levetop Electronics Co., Ltd.

13. Static Random Access Memory (SRAM)

13.1 Introduction

Features of the static random access memory (SRAM) include:

- On-chip 8-Kbyte SRAM
- Fixed address space
- Byte, half-word (16-bit), or word (32-bit) read/write accesses
- One clock per access (including bytes, half-words, and words)
- Supervisor or user mode access

13.2 Modes of Operation

Access to the SRAM is not restricted in any way. The array can be accessed in supervisor and user modes.

13.3 Low-Power Modes

In wait, doze and stop mode, clocks to the SRAM are disabled. No recovery time is required when exiting these modes.

13.4 Reset Operation

The SRAM contents are undefined immediately following a power-on reset. SRAM contents are unaffected by system reset. If a synchronous reset occurs during a read or writes access, then the access completes normally and any pipelined access in progress is stopped without corruption of the SRAM contents.

13.5 Interrupts

The SRAM module does not generate interrupt requests.

©Levetop Electronics Co., Ltd.

14. Cache (CACHE)

14.1 Features

The embedded flash contains an 1Kbytes, 2-way set-associative cache with a 16-byte line size. The cache improves system performance by providing low-latency data to the processor pipelines, which decouples processor performance from system memory performance. **Figure 14–1** shows a block diagram of the cache.

If a read operation address matches a valid cache tag entry, the access hits in the cache. The cache supplies the data to the processor. If the read access does not match a valid cache tag entry (misses in the cache), the cache will read data from the embedded flash then deliver it to the processor. The write operation will be performed to the embedded flash directly.

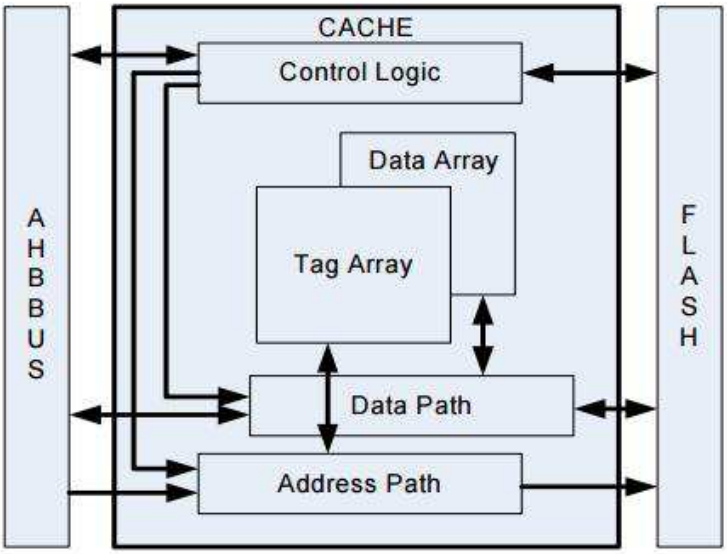


Figure 14–1: Embedded Flash Cache Block Diagram

14.2 Address Space Model

The embedded flash contains two address spaces as show in **Table 14–1.**

Table 14–1: Embedded Flash Address Map

Address Space	Address Range	Cachbility
Flash MAIN/RDN/NVR area	0x0000_0000 ~ 0x0001_97FF	Cached read and un-cached write
SRAM/Peripheral	0x0001_9800 ~ 0xFFFF_FFFF	Un-cached read and un-cached write

©Levetop Electronics Co., Ltd.

14.3 Cache Organization

The 2-way set associative cache is organized as two ways of 32 sets of storage. **Figure 14–2** illustrates the cache organization as well as the terminology used, along with the cache line format.

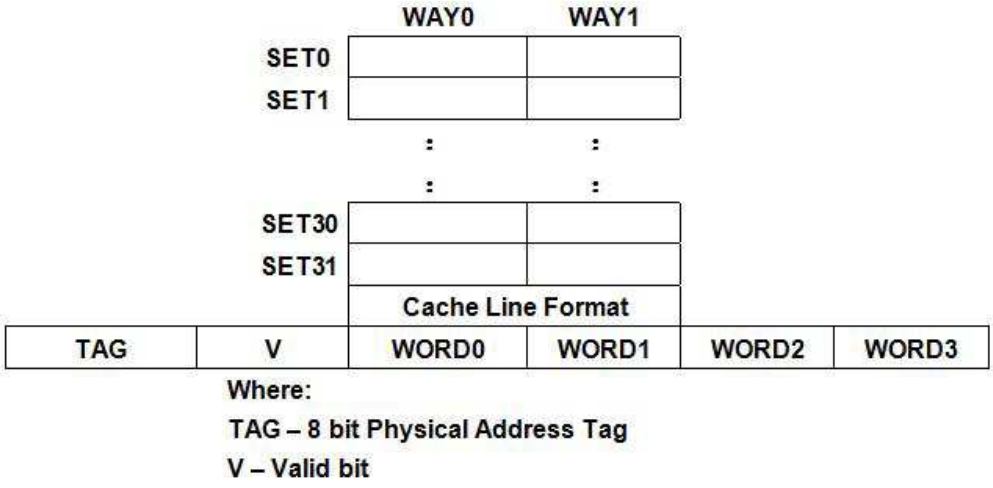


Figure 14–2: Cache Organization and Line Format

Address bits Addr[8:4] provide an index to select a set. Ways are selected according to the rules of set association.

Each line consists of a physical address tag, status bits, and four words of data. Address bits Addr[3:2] select the word within the line.

©Levetop Electronics Co., Ltd.

14.4 Cache Operation

The cache, once enabled, is searched for a tag match on all instruction fetches and data accesses from the CPU that correspond to a potentially cacheable area of the address space. If a match is found, the cached data is forwarded on a read access to the instruction fetch unit or the execution unit (data access).

When a read miss occurs, the cache will fetch a four-word cache line beginning with the requested word (critical word first). The line is fetched and placed into the appropriate cache block and the critical word is forwarded to the CPU. Subsequent data words may be streamed to the CPU if they have been requested.

The cache always fills an entire line, thereby providing validity on a line-by-line basis. A cache line is always in one of two states:

- Invalid
- Valid

For invalid lines, the V bit is clear, causing the cache line to be ignored during lookups. Valid lines have their V bit set, indicating the line contains valid data consistent with memory.

The cache should be explicitly invalidated by flushing tag memory after hardware reset. Reset does not invalidate the cache lines. Following initial power-up, the cache contents will be undefined.

To determine if the address is already allocated in the cache:

1. The cache set index, the address bit Addr[8:4] are used to select one cache set. A set is defined as the grouping of two lines (one from each way) corresponding to the same index into the cache array.
2. The higher order physical address bits (Addr[16:9]) are used as a tag reference or used to update the cache line tag field.
3. The two tags from the selected cache set are compared with the tag reference. If any one of the two tags matches the tag reference and the tag status is valid, a cache hit has occurred.
4. Addr[3:2] are used to select one of the four words in each line. A cache hit indicates that the selected data word (W[0:3]) in that cache line contain valid data (for a read access).

On a cache read miss, the cache controller uses a pseudo-round-robin replacement algorithm to determine which cache line will be selected to be replaced.

The replacement pointer is initialized to point to way0 on a reset.

©Levetop Electronics Co., Ltd.

14.4.1 Cache Line Tag Format

Each cache line tag entry contains the physical address tag, and a valid bit. The format of a tag entry is shown in **Figure 14–3**.

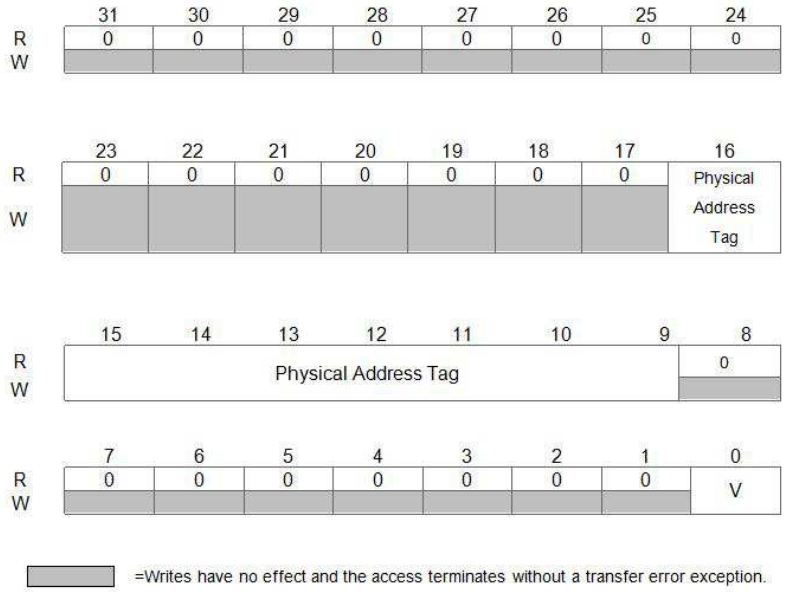


Figure 14–3: Tag Format

Bits[16:9] - Physical Address Tag.
The physical address corresponding to the data contained in this line

V- Valid bit.
0 = This bit signifies that the cache line is invalid and a tag match should not occur.
1 = This bit signifies that the cache line is valid.

©Levetop Electronics Co., Ltd.

14.4.2 Cache Control

Control of the cache is provided by bits in the cache control register (CACR).Control bits are provided to enable/disable the cache.

14.4.2.1 Cache Control Register Format

The format of the cache control register is shown in **Figure 14–4**. This register must be accessed with 32-bit access only. Byte and half-word access results are undefined

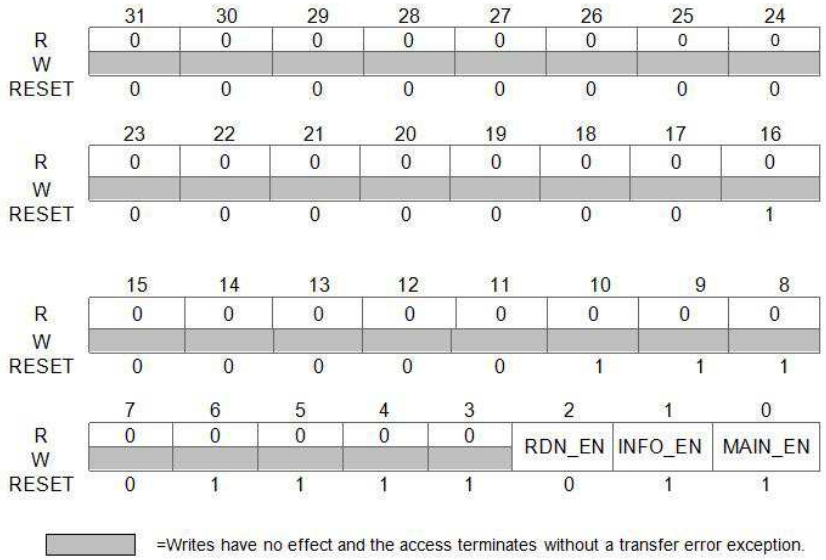


Figure 14–4: Cache Control Register(CACR)

MAIN_EN - Flash Main Area Cache Enable.

0 = The Flash Main Area Cache is disabled and will not provide data or accept data from the CPU. On reset, this bit is clear.

1 = The Flash Main Area Cache is enabled.

INFO_EN - Flash Information Area Cache Enable

0 = The Flash Information Area Cache is disabled and will not provide data or accept data from the CPU. On reset, this bit is clear.

1 = The Flash Information Area Cache is enabled.

RDN_EN - Flash RDN Area Cache Enable.

0 = The Flash RDN Area Cache is disabled and will not provide data or accept data from the CPU. On reset, this bit is clear.

1 = The Flash RDN Area Cache is enabled.

©Levetop Electronics Co., Ltd.

14.4.3 Cache Access Operations

The following paragraphs describe the cache search and cache fill operations.

14.4.3.1 Cache Search

A cache search consists of the following:

1. Address bits Addr[8:4] are used to index into the cache tag array, the tag and status bits of the cache line are read.
2. Address bits (Addr[16:9]) are compared with the stored physical tag value. If the tag matches the access address, and the V bit for the line is set, a cache hit has occurred. If the tag does not match the compare value, or the V bit is clear, a cache miss will occurred.
3. Addr[3:2] are used to select one of four words to be read or updated.

14.4.3.2 Cache Fill

In case of a cache read miss, the data corresponding to the requested access is not in the cache and must be fetched from memory. The following steps occur:

1. As soon as it is received, the requested data is forwarded to the CPU and is written into the selected cache line.
2. As subsequent words are received, they are forwarded to the CPU if a request for them has been made. After all words in the line have been successfully fetched, the tag V bit is set.

14.5 Cache Management

The CACR is used to enable and configure the cache. A hardware reset clears the CACR (MAIN_EN/INFO_EN/RDN_EN) bits, disabling the cache, and removing all configuration information. However, reset does not affect the tags, state information, and data within the cache arrays. Invalidate Tag operation should be performed before enabling it.

The address space directly addresses the cache's tag array. Accesses may be performed to invalidate a line.

The state of the cache enable bit in the CACR does not affect these operations

Within the cache control space, addresses are used to specify the operation as well as to specify the cache set and way indexes.

©Levetop Electronics Co., Ltd.

14.6 Cache Memory Map

Table 14–2 shows the offsets of the cache control register, tag and data array entries from the cache base address value for the CPU. The base address for the cache address map is 0xFFF80000. The cache memory map occupies the region 0xFFF81000 ~0xFFF82FFF. Accessing to the region 0xFFF83000~0xFFFFFFFF result in undefined data being returned on reads, and ignored on writes.

Table 14–2 shows the various register and entry offsets from the base address. Access to the cache tag entries, as well as to the CACR must be performed as 32-bit accesses. Byte and halfword accesses are not supported.

Table 14–2: Cache Address Map

Address	Use
0x0000	Cache Control Register(CACR)
…	Undefined
0x1000~0x13FF	Cache Tag Space
0x1000	Tag set 0, way 0
0x1010	Tag set 1, way 0
0x1020	Tag set 2, way 0
…	…
0x11F0	Tag set 31, way 0
0x1200	Tag set 0, way 1
0x1210	Tag set 1, way 1
0x1220	Tag set 2, way 1
…	…
0x13F0	Tag set 31, way1
0x2000~0x23FF	Cache Data Space
0x2000	Data set 0, way 0, Word 0
0x2004	Data set 0, way 0, Word 1
0x2008	Data set 0, way 0, Word 2
0x200C	Data set 0, way 0, Word 3
0x2010	Data set 1, way 0, Word 0
0x2014	Data set 1, way 0, Word 1
0x2018	Data set 1, way 0, Word 2
0x201C	Data set 1, way 0, Word 3
…	…
0x21F0	Data set 31, way 0, Word 0
0x21F4	Data set 31, way 0, Word 1
0x21F8	Data set 31, way 0, Word 2

©Levetop Electronics Co., Ltd.

Table 14–2: Cache Address Map (Continued)

Address	Use
0x21FC	Data set 31, way 0, Word 3
0x2200	Data set 0, way 0, Word 0
0x2204	Data set 0, way 1, Word 1
0x2208	Data set 0, way 1, Word 2
0x220C	Data set 0, way 1, Word 3
0x2210	Data set 1, way 1, Word 0
0x2214	Data set 1, way 1, Word 1
0x2218	Data set 1, way 1, Word 2
0x221C	Data set 1, way 1, Word 3
…	Undefined
0x23F0	Data set 31, way 1, Word 0
0x23F4	Data set 31, way 1, Word 1
0x23F8	Data set 31, way 1, Word 2
0x23FC	Data set 31, way 1, Word 3

©Levetop Electronics Co., Ltd.

15. Embedded Flash Module

15.1 Introduction

The Embedded FLASH(EFM) is a CMOS sector erase, half-word(16-bit) program(single-byte program is not permit).It is organized as 64K Bytes main array with twelve Non-Volatile Register sectors (NVR) (512 Bytes each), and eight redundant sectors (RDN) (512 bytes each).The sector erase operation erases all bytes within a sector(512bytes).

The Flash IP device writes (program or erase) with a 1.5 Volt typical power supply. Program and erase operations are performed under CPU control through a command driven interface to an internal state machine. It is not permitted to read from a FLASH physical block while the block is being programmed or erased.

15.2 Features

Features of the EFM include:

- 64-K bytes(Main Array)+6K bytes(Twelve NVR Sectors)+4K bytes(Eight RDN Sectors) of FLASH memory
- 24 MHz single cycle reads of bytes, aligned halfwords (16 bits) and aligned words (32 bits)
- Automated program and erase operation
- Optional interrupt on command completion
- Flexible scheme for protection against accidental program or erase operations
- Security for single-chip applications
- Single power supply (system VDD) used for all module operations

15.3 Modes of Operation

The EFM has two operating modes:

1. FLASH User Mode — In this mode, the EFM is used for non-volatile program and data storage. FLASH program and erase operations are controlled by user software.
2. FLASH STB Mode — This is used at the factory only to test the EFM. Refer to for a description of FLASH user mode operations.

©Levetop Electronics Co., Ltd.

15.4 Block Diagram

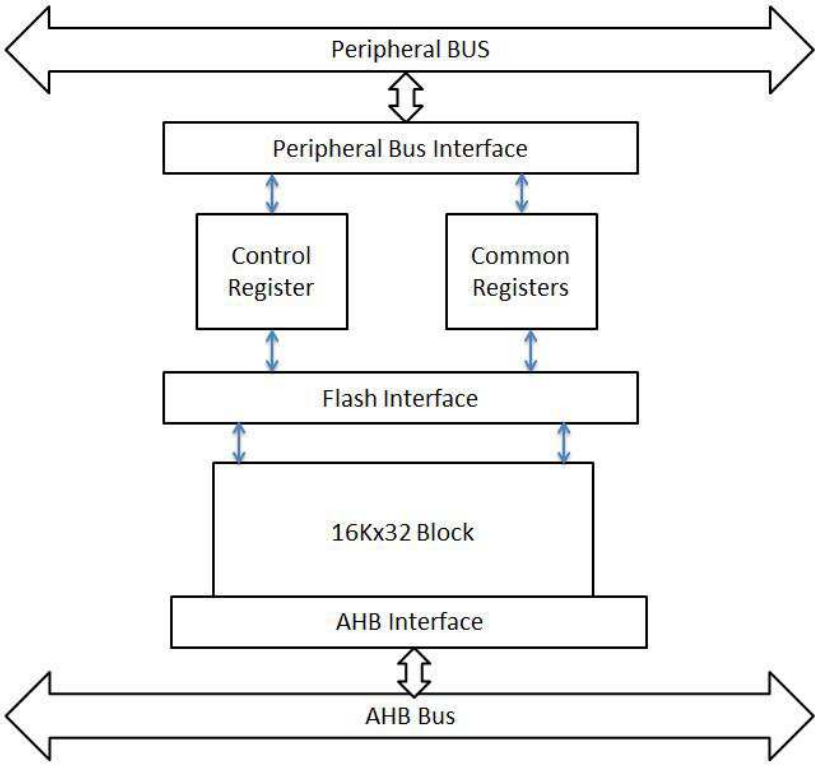


Figure 15–1: EFM Block Diagram

©Levetop Electronics Co., Ltd.

15.5 Module Memory Map (Base: 0x4012_0000)

The flash main array is mapped starting at address 0x0000_0000 and the flash NVR array is mapped starting at address 0x0000_0000.

Figure 15–2 shows the flash main and NVR arrays' address mapping.

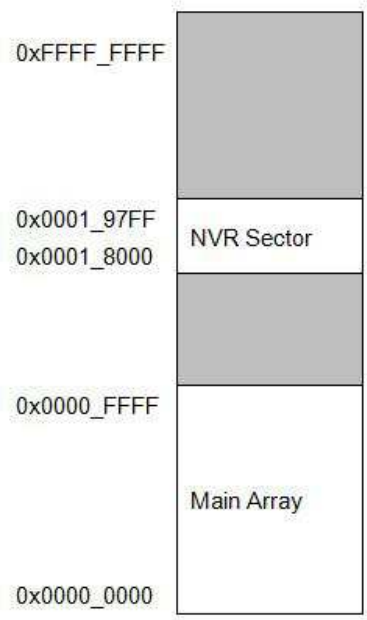


Figure 15–2: EFM Array Memory Map

The EFM module also contains a set of control and status registers. The memory map for these registers is shown in **Table 15–1**.

Table 15–1: EFM Register Memory Map

Address Offset	Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	Access[1,2]
0x0000	EFMCR				S
0x0004	EFMSEC0				S
0x0008	EFMSEC1				S
0x000c	EFMSEC2				S
0x0010	EFMTIM0				S
0x0014	EFMTIM1				S
0x0018	EFMCMD	Reserved[3]	EFMSTAT	Reserved	S

Notes:

1. S = CPU supervisor mode access only.
2. User mode accesses to supervisor-only address locations have no effect and result in a cycle termination transfer error.
3. Writes to reserved address locations have no effect and reads return 0s.

©Levetop Electronics Co., Ltd.

15.6 Register Descriptions

15.6.1 EFM Configuration Register (EFMCR)

The EFM Configuration Register (EFMCR) is unbanked and is used to configure and control the operation of the EFM array and bus interface unit (BIU)..

Address Offset: 0x0000 to 0x0003

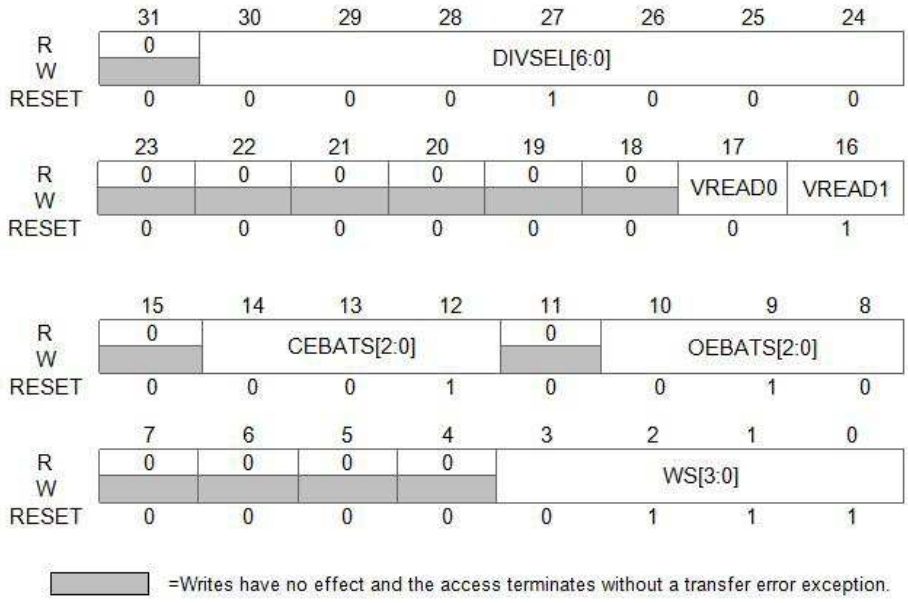


Figure 15–3: EFM Module Configuration Register (EFMCR)

DIVSEL[5:0] — Clock Divider Selects

The DIVSEL[5:0] effectively divides the EFM input clock down to a low frequency around 1MHz. EFM divided clock frequency = ipg_clk frequency /(DIVSEL[5:0] +1). DIVSEL[5:0] == 0 means the EFM divided clock frequency == ipg_clk frequency.ipg_clk cannot lower than 1.5MHz,for the margin of Tpgs minimum and maximum value is 0.7us.

VREAD0 — Enable VREAD0 flash function

1 = Enable VREAD0

0 = Disable VREAD0

VREAD1 — Enable VREAD1 flash function

1 = Enable VREAD1

0 = Disable VREAD1

CEBATS[2:0]— CEb Signal Assert Timing Select

These bits select the assertion timing of CEb when the operation is read or write the configuration registers. See functional description for details.

OEBATS[2:0]— OEb Signal Assert Timing Select

These bits select the assertion timing of OEb when the operation is read. See functional description for detail

WS[3:0] — Wait State Bits

The RWS field determines the number of wait states when the flash address matched.

©Levetop Electronics Co., Ltd.

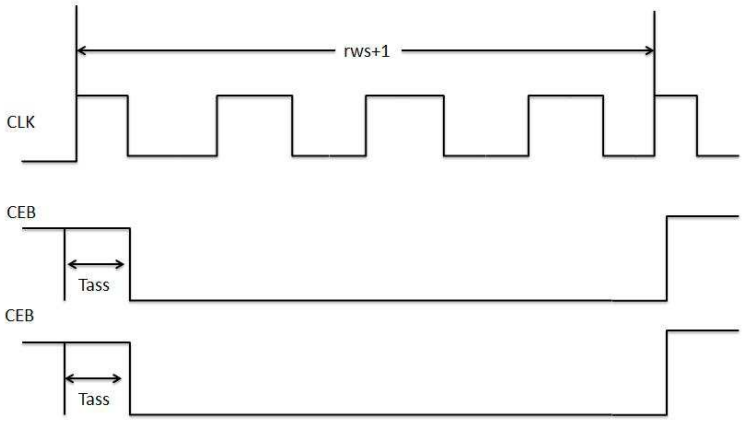


Figure 15–4: CEb/OEb Assert Timing

Table 15–2: Assert and Negate Timing (Unit: system cycle)

OEBATS[2:0]/CEBATS[2:0]	Assert Time (T_{ass})
000	0
001	1/2
010	2/2
011	3/2
100	4/2
101	5/2
110	6/2
111	7/2

©Levetop Electronics Co., Ltd.

15.6.2 EFM Security Read Register0 (EFMSEC0)

The EFM Security Read Register0 (EFMSEC0) is used to store from the first to the last RDN repair enable or disable information.

Address Offset: 0x0004 to 0x0007

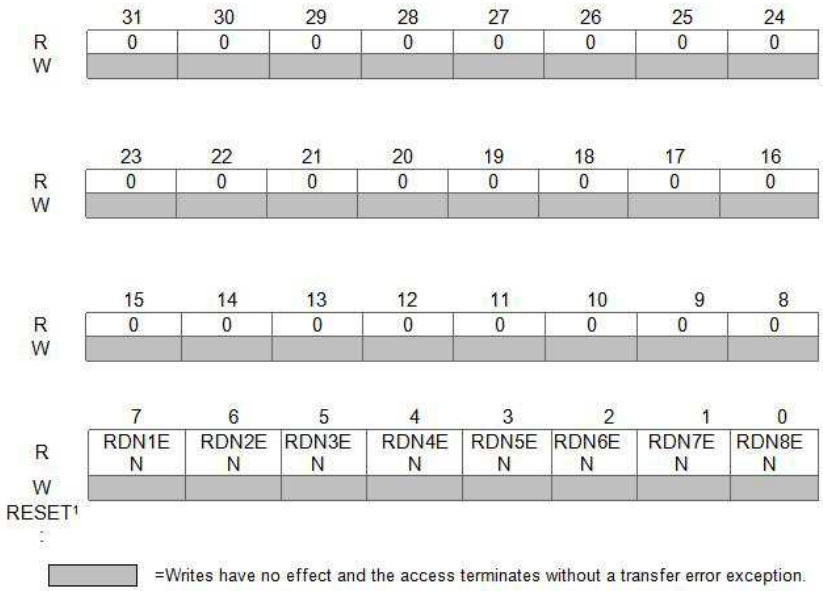


Figure 15–5: EFMSEC0

Note:

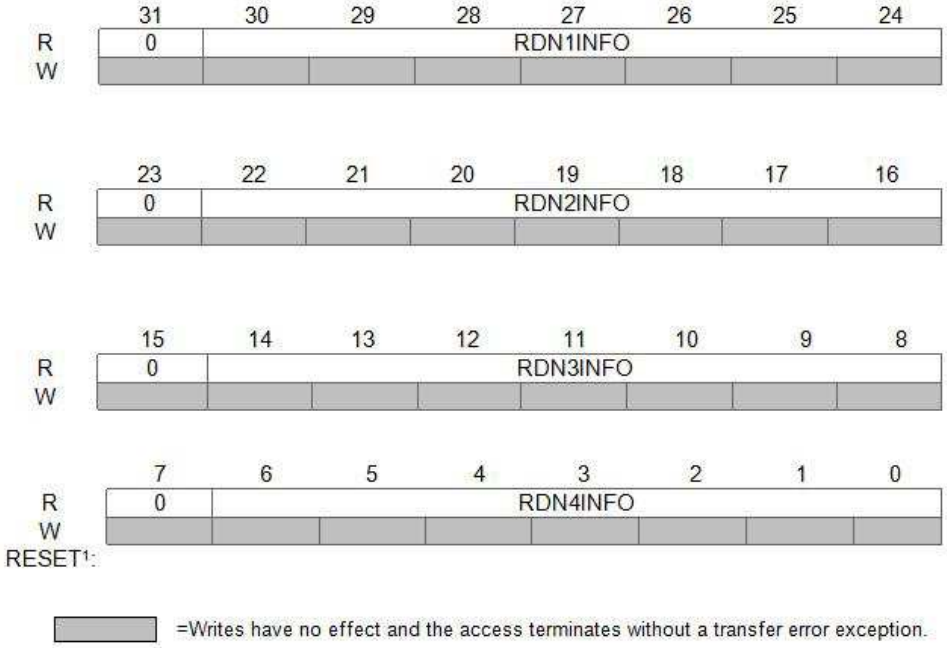
1. reset value is decided by the RDN information in NVR12 array

©Levetop Electronics Co., Ltd.

15.6.3 EFM Security Read Register1 (EFMSEC1)

The EFM Security Read Register1 (EFMSEC1) is used to store the RDN information.

Address Offset: 0x0008 to 0x000b



Note:

1. reset value is decided by the RDN information in NVR12 array

Figure 15–6: EFMSEC1

RDNxINFO[6:0] — RDNx Repair Information

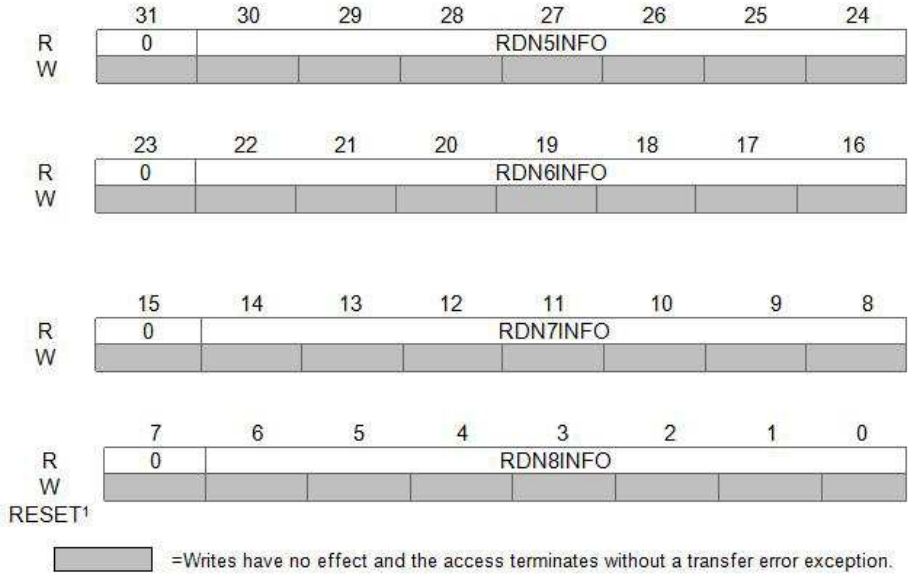
This 7-bit information indicates the bad sector address which will be replaced with RDNx sector.

©Levetop Electronics Co., Ltd.

15.6.4 EFM Security Read Register2 (EFMSEC2)

The EFM Security Read Register2 (EFMSEC2) is used to RDN information.

Address Offset: 0x000c to 0x000f



Note:

1. reset value is decided by the RDN information in NVR12 array

Figure 15–7: EFMSEC2

RDNxINFO[6:0] — RDNx Repair Information

This 7-bit information indicates the bad sector address which will be replaced with RDNx sector.

©Levetop Electronics Co., Ltd.

15.6.5 EFM Timing Register0 (EFMTIM0)

The EFM Timing Register0 (EFMTIM0) is used to configure the timing parameter which used for timed events in program and erase algorithms.

Address Offset: 0x0010 to 0x0013

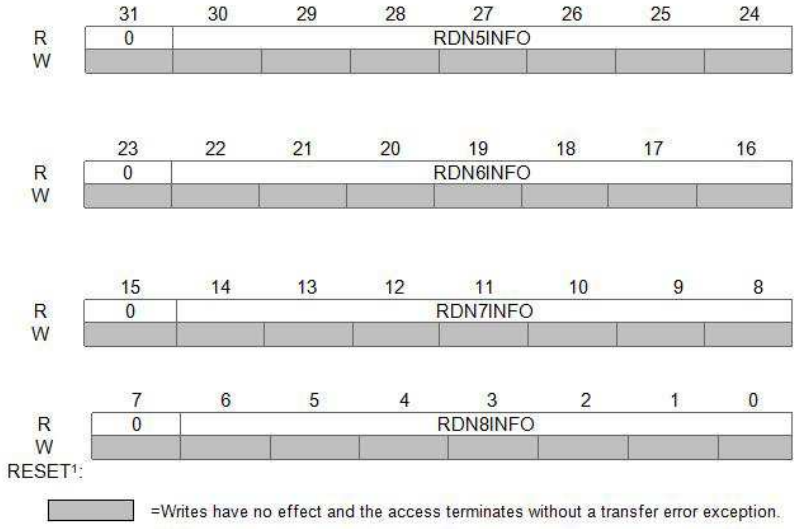


Figure 15–8: EFMTIM0

TPGS[8:0] — Tpgs Cycle Counter Number

Configure the wait cycles to satisfy WEb low to PROG low setup time. The WEb low to PROG2 high setup time should between 2.5 ~3.2us.TPGS counts cycles which are based on the ipg_clk. So the TPGS can be set like this:

TPGS = (2.5~3.2us)/ipg_clk period.

TERASE_SMALL[12:0] — Sector/Block Erase Terase Cycle Counter Number

Configure the wait cycles to satisfy Sector erase time. The sector erase time should between 4~5ms.So the TERASE_SMALL can be set like this:

TERASE_SMALL = (4~5ms)/The EFM divide clock period.

©Levetop Electronics Co., Ltd.

15.6.6 EFM Timing Register1 (EFMTIM1)

The EFM Timing Register1 (EFMTIM1) is used to configure the timing parameter which used for timed events in program and erase algorithms.

Address Offset: 0x0014 to 0x0017

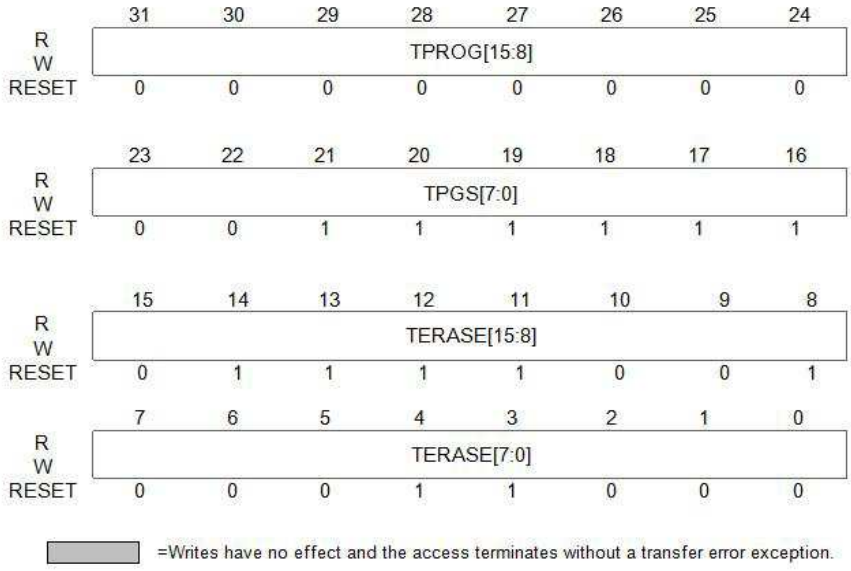


Figure 15–9: EFMTIM1

TPROG[15:0] — Tprog Cycle Counter Number

Configure the wait cycles to satisfy byte program time. The program time should between 6~7.5us.TPROG counts cycles which are based on the ipg_clk. So the TPROG can be set like this:

TPROG = (6~7.5us)/ipg_clk period.

TERASE[15:0] — Terase Cycle Counter Number

Configure the wait cycles to satisfy Chip erase time. The minimum chip erase time is 30ms,and the maximum chip erase time is 40ms.so the TERASE can be set like this:

TERASE= (30ms~40ms)/The EFM divide clock period.

©Levetop Electronics Co., Ltd.

15.6.7 EFM Command Register (EFMCMD)

Address Offset: 0x00018

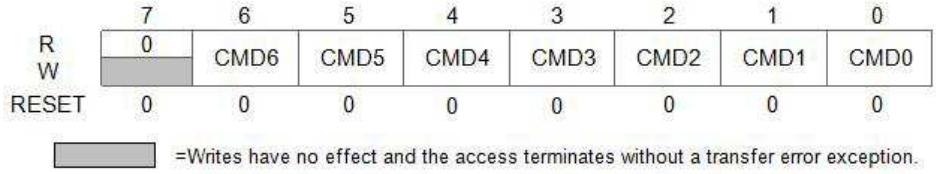


Figure 15–10: EFMCMD

Table 15–3: Flash Valid Command

EFMCMD	Meaning	Description
$20	Program	Program a 16/32-bit word
$40	Sector erase	Erase a sector(512byte) of Flash.
$41	Chip erase	Erase a block(128Kbytes main array) of Flash

15.6.8 EFM Status Register (EFMSTAT)

The EFM Status Register (EFMSTAT) is banked reports FLASH state machine command status and array access errors status.

Address Offset: 0x001a

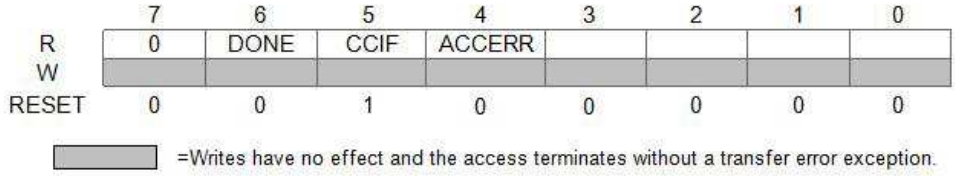


Figure 15–11: EFM Status Register(EFMSTAT)

CCIF — Command Complete Indicate Flag

The CCIF flag indicates that no commands are pending for the FLASH physical blocks. CCIF is set and cleared automatically upon start and completion of a command. Writing to CCIF has no effect.

1 = All commands are completed

0 = Command in progress

DONE — Command Complete Interrupt Flag

The DONE flag indicates that no commands are pending for the FLASH physical blocks. DONE is set automatically upon completion of a command. DONE is cleared by writing it to 1. The DONE bit can trigger an interrupt request if the DONEIE bit is set in EFMCR

1 = All commands are completed

0 = No effect

ACCERR — Access Error Flag

The ACCERR flag indicates an illegal access to the EFM array or registers caused by a bad program or erase sequence. ACCERR is cleared by writing it to 1. Writing a 0 to ACCERR has no effect. See for details on what sets the ACCERR flag.

1 = Access error has occurred

0 = No failure

©Levetop Electronics Co., Ltd.

15.7 Functional Description

15.7.1 Program and Erase Operations

15.7.1.1 Setting the EFMTIM0/1 Register

Prior to issuing any program or erase commands, EFMTIM0/1 must be written to set the corresponding configurations which are needed during program or erase operation.

15.7.1.2 Program, Erase and Verify Sequences

This three-step command write sequence must be strictly followed. No intermediate writes to the EFM module are permitted between these three steps. The command write sequence is:

1. Read CCIF bit. If CCIF bit is set, the address, data, and command are ready for a new command sequence to begin.
2. Write the half-word/word to be programmed to its location in the EFM array. The address and data will be stored in internal buffers. All address bits are valid for program commands. The value of the data written for verify and erase commands is ignored. For chip erase or verify, the address can be any location in the EFM array. For sector erase, address bits [8:0] are ignored.
3. Write the valid command to EFMCMD and launch the command. CCIF will be cleared automatically when the command is launched.

When command execution is complete, the FLASH state machine will set the CCIF and DONE flag, indicating that the address, data, and command are ready for a new command sequence to begin.

DONE flag should be cleared before commencing another command write sequence.

The FLASH state machine will flag errors in command write sequences by means of the ACCERR flags in the EFMSTAT register. The ACCERR flags must be cleared before commencing another command write sequence.

15.7.1.3 Flash Illegal Operations

The ACCERR flag will be set during a command write sequence if any of the illegal operations below are performed.

1. Writing or reading the EFM array when the command sequence is busy(CCIF = 0) will has no effect and set ACCERR;
2. Writing a new command to EFMCMD when the command sequence is busy (CCIF = 0) will has no effect and set ACCERR. The command register will not be updated;
3. Program or sector erase NVR when not permitted1 will has no effect and set ACCERR;
4. Program or sector erase NVR when the input password not correct if password is used will has no effect and set ACCERR;
5. Trying to program or sector erase NVR12(factory seclor) will has no effect and set ACCERR
6. Single-byte program will have no effect and set ACCERR.

©Levetop Electronics Co., Ltd.

16. Option Byte

16.1 Register Memory Map (Base: 0x4012_0000)

Table 16–1: Register Memory Map

Address Offset	Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	Access[1,2]
0x001c	CCR		PVDC		S
0x0020	IOSCST		EOSCST		S
0x0024	PVDFEVR		RFEVR		S
0x002c	FCR		Reserved[3]		S
0x0030	IOSTC				S
0x0034	LDOTC	VREFTCR	ADCCDISR		S
0x0038	MPUCONFR				S
0x003c	GLFTC		Reserved		S

Notes:

1. S = CPU supervisor mode access only.
2. User mode accesses to supervisor-only address locations have no effect and result in a cycle termination transfer error.
3. Writes to reserved address locations have no effect and reads return 0s.

16.1.1 Register Descriptions

16.1.1.1 PVDC— Programmable Voltage Detector Configuration Register

Address: 0x001c

	15	14	13	12	11	10	9	8
Read	PVDF	PVDORRE	PVDTEST[1:0]		PVDOE	PVDRE	PVDIE	PVDE
Write								
RESET	0	0	0	0	1	Note1	0	1

	7	6	5	4	3	2	1	0
Read	0	0	0	0	0	PVDC[2:0]		
Write								
RESET	0	0	0	1	1	0	0	0

Figure 16–1: PVDC— Programmable Voltage Detector Configuration Register

Note:

1. Determined by the value of Customer Information Area

©Levetop Electronics Co., Ltd.

PVDTEST[1:0] — Write Access Enable Sequence Input

The PVDC register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the PVDC register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

PVDF— Programmable Voltage Detector Flag

The PVDF indicates VDD is lower than PVD threshold. POR can clear it.

1 = VDD33 is lower than PVD threshold

0 = VDD33 is not lower than PVD threshold

PVDOE— Programmable Voltage Detector Output Enable

1 = PVD Output Enable

0 = PVD Output Disable

PVDE— Programmable Voltage Detector Enable

The PVDE Configure if PVD is enabled. POR can clear it

1 = Enable PVD

0 = Disable PVD

PVDPORRE— Program Voltage Detector (PVD) Power-On Reset Enable

The PVDPORRE shows whether Program Voltage Detector Power-On reset is enable.

1 = PVD Power-On reset enabled

0 = PVD Power-On reset disabled

PVDRE— Program Voltage Detector (PVD) Reset Enable

The PVDRE shows whether Program Voltage Detector reset is enable. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on. Software can change it after power on reset.

1 = PVD reset enabled

0 = PVD reset disabled

PVDIE— Programmable Voltage Detector Interrupt Enable

The PVDRE configure if VCC is lower than PVD threshold can generate a interrupt

1 = VCC is lower than PVD threshold generates a interrupt

0 = VCC is lower than PVD threshold not generates a interrupt

PVDC[2:0] — Programmable Voltage Detector Configuration Value.

Table 16–2: PVDC Description

PVDC	Detect Voltage
3'b000	2.16V
3'b001	2.32V
3'b010	2.48V
3'b011	2.64V
3'b100	3.92V
3'b101	4.08V
3'b110	4.24V
3'b111	4.40V

The default value is loaded Value from the customer information area if enabled. Software can change it after power on reset.

©Levetop Electronics Co., Ltd.

16.1.1.2 CCR — Customer Configuration Register

Address: 0x001e

	31	30	29	28	27	26	25	24
R	CIAPPDIS	CIAPEDIS	EXTALDIS	JTAGDIS	RSTOUT DIS	CLKOUT DIS	CLKMD	0
W								
RESET	Note1	Note1	Note1	Note1	Note1	Note1	Note1	0

	23	22	21	20	19	18	17	16
R	EXTALBYP	STBDLDIS	ADCC SPOR	PVDCE	IOSCSTE	EOSCSTE	CCRTEST[1:0]	
W								
RESET	0	Note1	0	Note1	Note1	Note1	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 16–2: CCR — Customer Configuration Register

Note:
1. Determined by the value of Customer Information Area

CCRTEST[1:0] — Write Access Enable Sequence Input

The writable bit of CCR register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of CCR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

CIAPPDIS— Customer Information Area Program Disable Bit

The CIAPPDIS shows whether the program of Customer Information Area Page is disabled .The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

1 = Customer Information Area Page Program disabled
0 = Customer Information Area Page Program enabled

CIAPEDIS— Customer Information Area Page Erase Disable Bit

The CIAPEDIS shows whether the page erase of Customer Information Area Page is disabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

1 = Customer Information Area Page Erase disabled
0 = Customer Information Area Page Erase enabled

©Levetop Electronics Co., Ltd.

EXTALDIS— External crystal Disable Bit

The EXTALDIS shows whether the External crystal is disabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is set, otherwise if the content in corresponding customer information address is matched, this bit is clear after power-on. Software can change it after power on reset.

1 = External crystal function disabled

0 = External crystal function enabled

External crystal Enabled	External crystal Disabled
EXTAL	INT0[2]
XTAL	INT1[7]

EXTALBYP— External crystal Bypass Control Bit

The EXTALBYP shows whether the External crystal is bypass mode. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on. Software can change it after power on reset.

1 = External crystal is in bypass mode, and XTAL work as GPIO, and EXTAL work as ext_clock input

0 = External crystal bypass mode is disabled

External crystal Enabled	External crystal Bypass
XTAL	INT1[7]

SWDDIS— SWD Debug Interface Disable Bit

The SWDDIS shows whether the SWD Debug Interface is disabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

1 = SWD function disabled

0 = SWD function enabled

SWD Enabled	SWD Disabled
SWDCLK	INT0[0]
SWDIO	AIN[2]

CLKOUTDIS— CLKOUT Disable Bit

The CLKOUTDIS shows whether the CLKOUT pin function is disabled and INT1[6] is enabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is set, otherwise if the content in corresponding customer information address is matched, this bit is clear after power-on. Software can change it after power on reset.

1 = CLKOUT Pin function disabled

0 = CLKOUT Pin function enabled

Clkout Enabled	Clkout Disabled
CLKOUT	INT1[0]

©Levetop Electronics Co., Ltd.

RSTOUTDIS— RSTOUT Disable Bit

The RSTOUTDIS shows whether the RSTOUT pin function is disabled and INT1[6] is enabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is set, otherwise if the content in corresponding customer information address is matched, this bit is clear after power-on. Software can change it after power on reset.

1 = RSTOUT Pin function disabled

0 = RSTOUT Pin function enabled

Rstout Enabled	Rstout Disabled
RSTOUT	INT1[6]

CLKMD— Clock Mode Control Bit

The CLKMD reflects the clock source of the chip. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is set, otherwise if the content in corresponding customer information address is matched, this bit is clear after power-on.

Table 16–3: CLKMD Description

CLKMD	Description
1'b1	FIRC144MHZ
1'b0	FOXC

STBDLDIS— STB Download Disable Bit

The STBDLDIS shows flash serial download bus is disabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on

1 = STB Download function disabled

0 = STB Download function enabled

ADCCSPOR— ADC Channel Setting Bit after power-on

The ADCSPOR shows whether the ADC channel default setting bit is loaded customer information area or not power on. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

PVDCE— Program Voltage Detector (PVD) Configuration Enable

The PVDCE shows whether Program Voltage Detector configuration is loaded from Customer Information Area after power up. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on

1 = PVD configuration loaded from Customer Information Area after POR

0 = PVD configuration not loaded from Customer Information Area after POR

IOSCSTE— Internal High Speed Oscillator Stable Time Configuration Enable

The IOSCSTE shows whether Internal High Speed Oscillator Stable Time configuration is loaded from Customer Information Area after power up. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

1 = Internal High Speed Oscillator Stable Time configuration loaded from Customer Information Area after POR

0 = Internal High Speed Oscillator Stable Time configuration not loaded from Customer Information Area after POR

©Levetop Electronics Co., Ltd.

EOSCSTE— External High Speed Oscillator Stable Time Configuration Enable

The EOSCSTE shows whether External High Speed Oscillator Stable Time configuration is loaded from Customer Information Area after power up. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on.

1 = Internal High Speed Oscillator Stable Time configuration loaded from Customer Information Area after POR

0 = Internal High Speed Oscillator Stable Time configuration not loaded from Customer Information Area after POR

CIAKEYE— Customer Information Area Key Enable Bit

The CIAKEYE reflects whether the key of Customer Information Area is enabled. The default value is loaded Value from the customer information area. If the corresponding customer information area is erased status, this bit is clear, otherwise if the content in corresponding customer information address is matched, this bit is set after power-on

1 = Customer Information Area Key is set

0 = Customer Information Area Key is not set

16.1.1.3 EOSCST— External Oscillator Stable Time Configuration Register

Address: 0x0020

	15	14	13	12	11	10	9	8
R W	EOSCTEST[1:0]		0	0	0	0	EOSCST[9:8]	
RESET	0	0	0	0	0	0	Note1	Note1

	7	6	5	4	3	2	1	0
R W	EOSCST[7:0]							
RESET	Note1	Note1	Note1	Note1	Note1	Note1	Note1	Note1

=Writes have no effect and the access terminates without a transfer error exception.

Figure 16–3: EOSCST— External Oscillator Stable Time Configuration Register

Note:

Determined by the value of Customer Information Area

EOSCTEST[1:0] — Write Access Enable Sequence Input

The writable bit of EOSCST register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of EOSCST register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

EOSCST[9:0] — External Oscillator Stable Time Value.

The value is loaded from customer information area during power-on if enabled, otherwise is 10'h3ff.Software can change it after power on reset. After enabled, the external oscillator will wait EOSCST[9:0] cycles of 128KHZ oscillator and then output clock.

©Levetop Electronics Co., Ltd.

16.1.1.4 IOSCST— Internal High Speed Osc. Stable Time Configuration Register

Address: 0x0022

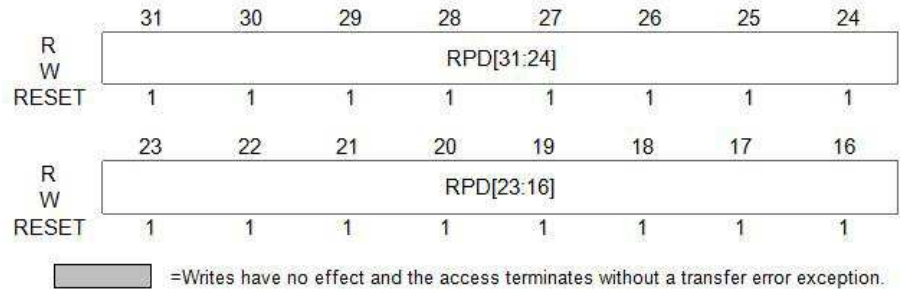


Figure 16–4: IOSCST— Internal High Speed Oscillator Stable Time Configuration Register

IOSCTEST[1:0] — Write Access Enable Sequence Input

The IOSCST register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the IOSCST register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

IOSCST[9:0] — Internal High Speed Oscillator Stable Time Value.

The value is loaded from customer information area during power-on if enabled, otherwise is 10'h3ff.Software can change it after power on reset. After enabled, the internal high speed oscillator will wait IOSCST[9:0] cycles of 128KHZ oscillator and then output clock.

16.1.1.5 RFEVR— RESET Pin Filter Enable and Value Register

Address: 0x0024

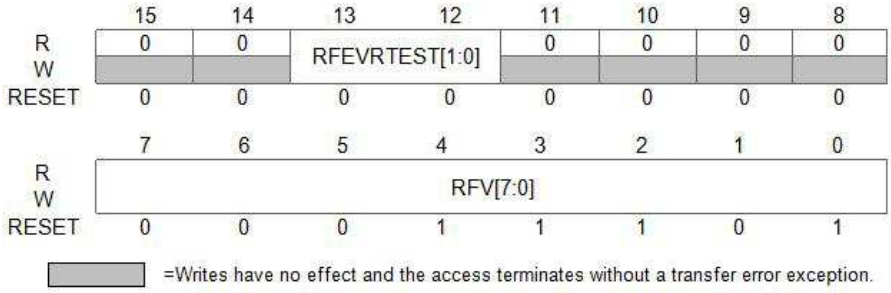


Figure 16–5: RFEVR— RESET Pin Filter Enable and Value Register

PVDFTEST[1:0] — Write Access Enable Sequence Input

The writable bit of RFEVR register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of RFEVR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

RFV[7:0]— RESET Pin Filter Value

The default value of RFV[7:0] is loaded Value from the customer information area if enabled, otherwise is 8'h. Software can change it after power on reset.

©Levetop Electronics Co., Ltd.

16.1.1.6 PVDFEVR— Programmable Voltage Detector Filter Control Register

Address: 0x0026

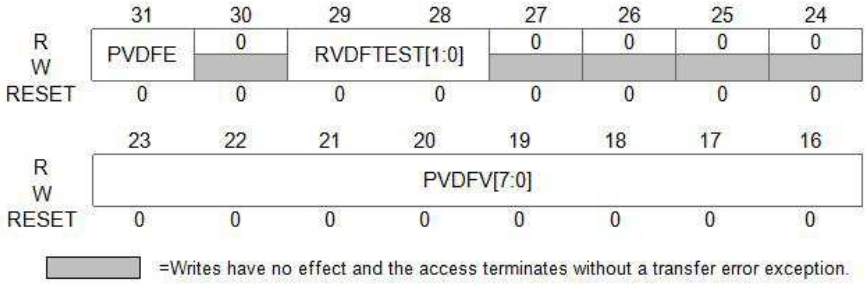


Figure 16–6: PVDFEVR— Programmable Voltage Detector Filter Enable and Value Register

PVDFTEST[1:0] — Write Access Enable Sequence Input

The writable bit of PVDFEVR register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of PVDFEVR register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

PVDFE— Programmable Voltage Detector Filter Enable.

The PVDFE shows whether Program Voltage Detector Filter is enabled.

1 = PVD filter is enabled

0 = PVD filter is not enabled

PVDFV[7:0]— Programmable Voltage Detector Filter Value

If PVDFE is set, PVDFV[7:0] is loaded Value from the customer information area. Software can change it after power on reset.

©Levetop Electronics Co., Ltd.

16.1.1.7 FCR — Factory Configuration Register

Address: 0x002e

	31	30	29	28	27	26	25	24
R	FIAPPDIS	FIAPEDIS	0	TMDIS	IOSCTE	LDO1P5TE	VREFTE	LDO3P3TE
W								
RESET:	Note1	Note1	0	0	Note1	Note1	Note1	Note1

	23	22	21	20	19	18	17	16
R	MPUPDIS	MPUEDIS	0	0	0	0	0	BIGENDIAN
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 16–7: FCR — Factory Configuration Register

Note: Determined by flash trim.

FIAPPDIS— Factory Information Area Program Disable Bit

The FIAPPDIS shows the program of Factory Information Area Pages is disabled. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Factory Information Area Page Program disabled

0 = Factory Information Area Page Program enabled

FIAPEDIS— Factory Information Area Page Erase Disable Bit

The FIAPEDIS shows the page erase of Factory Information Area Pages is disabled. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Factory Information Area Page Erase disabled

0 = Factory Information Area Page Erase enabled

TMDIS— Test Mode Disable Bit

The TMDIS shows test mode except STB is disabled. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Test mode disabled

0 = Test mode enabled

IOSCTE — Internal High Speed Oscillator Trimming Enable

The IOSCSTE shows whether Internal High Speed Oscillator is trimmed. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Internal High Speed Oscillator is trimmed

0 = Internal High Speed Oscillator is not trimmed

©Levetop Electronics Co., Ltd.

LDO1P5TE — LDO1P5 Trimming Enable

The LDO1P5TE shows whether LDO1P5 is trimmed. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = LDO1P5 is trimmed

0 = LDO1P5 is not trimmed

VREFTE— VREF Module Trimming Configuration Register

The VREFTE shows whether VREF Module is trimmed. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Internal VREF Module Trimming Configuration loaded from factory Information Area after POR

0= Internal VREF Module Trimming Configuration not loaded from factory Information Area after POR

LDO3P3TE — LDO3P3 Trimming Enable

The LDO3P3TE shows whether LDO3P3 is trimmed. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = LDO3P3 is trimmed

0 = LDO3P3 is not trimmed

MPUPDIS— MPU Configuration Information Area Program Disable Bit

The MPUPDIS shows the program of MPU Configuration Information Area Pages is disabled. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = MPU Configuration Information Area Page Program disabled

0 = MPU Configuration Information Area Page Program enabled

MPUEDIS— MPU Configuration Information Area Page Erase Disable Bit

The MPUEDIS shows the page erase of MPU Configuration Information Area Pages is disabled. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on

1 = MPU Configuration Information Area Page Erase disabled

0 = MPU Configuration Information Area Page Erase enabled

ENDIAN— Big Endian Enable Bit

The ENDIAN shows the data organization of CPU. The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = CPU Big-endian enabled

0 = CPU Little-endian enabled

©Levetop Electronics Co., Ltd.

16.1.1.8 IOSCTC— Internal High Speed Oscillator Trimming Config. Register

Address: 0x0030

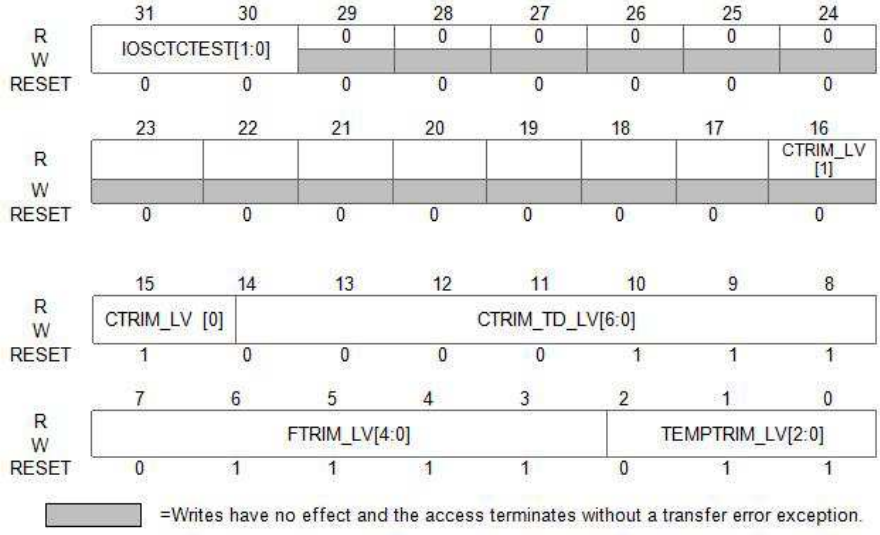


Figure 16–8: IOSCTC—Internal High Speed Oscillator Trimming Configuration Register

IOSCTCTEST[1:0] —IOSCTC Write Access Sequence In

The writable bit of IOSCTC register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of IOSCTC register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

CTRIM_LV[1:0] — Coarse Trimming Value for high speed oscillator, and is loaded from factory information area during power-on if enabled. Software can change it after power on reset.

CTRIM_TD_LV[4:0] — Coarse Trimming Value for high speed oscillator, and is loaded from factory information area during power-on if enabled. Software can change it after power on reset.

FTRIM_LV[6:0] — Fine Trimming Value for high speed oscillator, and is loaded from factory information area during power-on if enabled. Software can change it after power on reset.

TEMPTRIM_LV[2:0]— Temperature correction for high speed oscillator, and is loaded from factory information area during power-on if enabled. Software can change it after power on reset.

©Levetop Electronics Co., Ltd.

16.1.1.9 ADCCDISR— ADC Channel Disable Configuration Register

Address: 0x0034

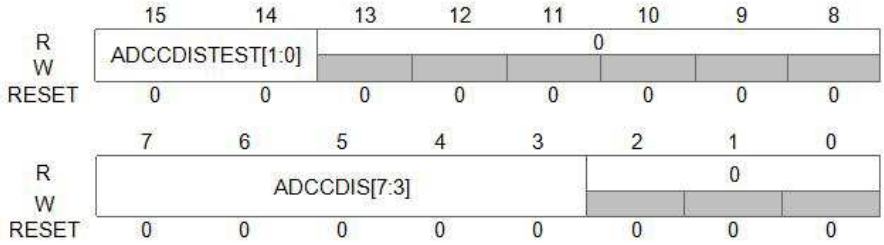


Figure 16–9: GLFTCR— Glitch Filter Trimming Configuration Register

ADCCDISTEST[1:0] — Write Access Enable Sequence Input

The writable bit of ADCCDISTEST register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of ADCCDISTEST register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

ADCCDIS[7:3]— ADC Channel Disable Configuration

If ADCSPOR is set in CCR register, ADCCDIS[7:3] are loaded Value from the custom information area. Software can change it after power on reset. If the corresponding bit is set, then the ADC channel is disabled, and the ADC channel is used as GPIO.

Table 16–4: ADC Channel Disable Configuration

ADC Channel Disabled	ADC Channel Enabled
INT0[3]	AIN[3]
INT0[4]	AIN[4]
INT0[5]	AIN[5]
INT0[6]	AIN[6]
INT0[7]	AIN[7]

16.1.1.10 VREFTCR— VREF Trimming Configuration Register

Address: 0x0036

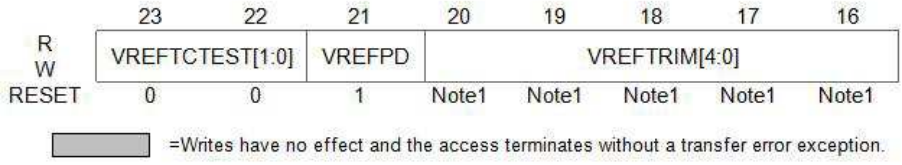


Figure 16–10: VREFTCR— VREF Trimming Configuration Register

©Levetop Electronics Co., Ltd.

Note:

1. Determined by the value of Factory Information Area

VREFTCTEST[1:0] — WSFTCR Write Access Sequence In

The writable bit of VREFTCTEST register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of VREFTCTEST register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

VREFPD — Internal 1.2Voltage Reference Source Power Down

1 = Internal Reference Voltage Source Power Down

0 = Internal Reference Voltage Source Power On

VREFTRIM[4:0] — VREF Trimming value

If VREFTE in FCR is set, VREFTRIM[4:0] is the loaded Value from the factory information area, otherwise the value is 5'h0.. Software can change it after power on reset.

16.1.1.11 LDO1P5TC— LDO1P5 Trimming Configuration Register

Address: 0x0037

	31	30	29	28	27	26	25	24
R W	LDO1P5TEST[1:0]		LDO1P5TC[5:0]					
RESET	0	0	Note1	Note1	Note1	Note1	Note1	Note1

=Writes have no effect and the access terminates without a transfer error exception.

Figure 16–11: LDO1P5TC— LDO1P5 Trimming Configuration Register

Note: Determined by the value of Factory Information Area

LDO1P5TEST[1:0] — LDO1P5TC Write Access Sequence In

The writable bit of LDO1P5TC register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of LDO1P5TC register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

LDO1P5TC[5:0] — LDO1P5 Trimming Value

IfLDO1P5TE is set, LDO1P5TC[5:0] is the loaded Value from the factory information area, otherwise the value is 6'h0. Software can change it after power on reset.

©Levetop Electronics Co., Ltd.

16.1.1.12 MPUCONFR— Memory Protect Unit Trimming Config. Register

Address: 0x0038

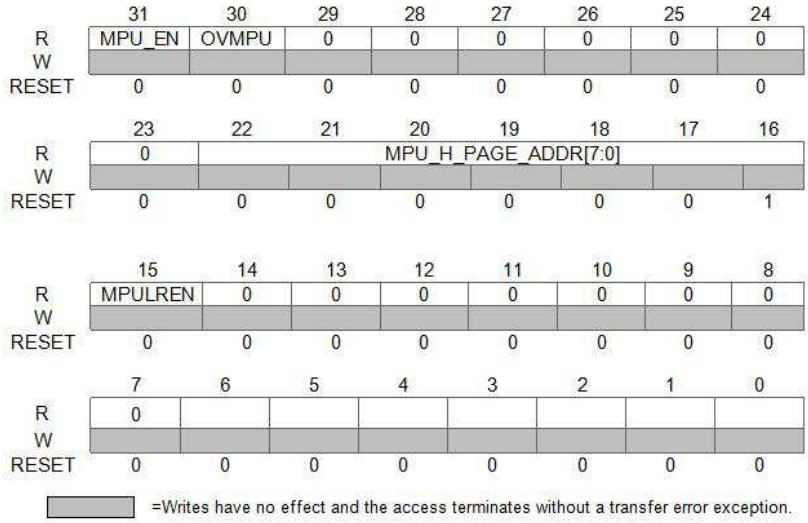


Figure 16–12: MPUCONFR—Memory Project Unit Configuration Register

MPU_EN — MPU enable status bit

The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = MPU is enabled

0 = MPU is disabled

OVMPU — MPU configuration area overwrite status bit

The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is set, otherwise if the content in corresponding factory information address is matched, this bit is clear after power-on.

1 = MPU configuration is overwrite by FIA MPU configuration

0 = MPU configuration is not overwrite by FIA MPU configuration

MPULREN — MPU area local read enabled status bit.

The default value is loaded Value from the factory information area. If the corresponding custofactorymer information area is erased status, this bit is clear, otherwise if the content in corresponding factory information address is matched, this bit is set after power-on.

1 = Memory protected area is readable by the local read

0 = Memory protected area is not readable by the local read

MPU_H_PAGE_ADDR[6:0] — Memory protected area higher page boundary,.

The default value is loaded Value from the factory information area. The protected area size is (MPU_H_PAGE_ADDR[6:0] - MPU_L_PAGE_ADDR[6:0]) * 512Bytes

MPU_L_PAGE_ADDR[6:0] — Memory protected area lower page boundary.

The default value is loaded Value from the factory information area. The protected area size is (MPU_H_PAGE_ADDR[6:0] - MPU_L_PAGE_ADDR[6:0]) * 512Bytes

©Levetop Electronics Co., Ltd.

16.1.1.13 LDO3P3TC— LDO3P3Trimming Configuration Register

Address: 0x003e

	31	30	29	28	27	26	25	24
R	LDO3P3TEST[1:0]		0	0	0	0	0	0
W								
RESET	Note1		0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	LDO3P3PD	LDO3P3BYP	0	0	0	0	LDO3P3TC[1:0]	
W								
RESET	0	0	0	0	0	0	Note1	Note1

Figure 16–13: LDO3P3TC— LDO3P3 Trimming Configuration Register

LDO3P3TEST[1:0] — LDO3P3TC Write Access Sequence In

The writable bit of LDO3P3TC register cannot be changed, unless the correct sequence write in. The right sequence is: 2'b01->2'b10->2'b11.After write these two bits following this sequence, these two bits' value == 2'b11, then the writable bit of LDO3P3TC register can be changed at will. Only writes 2'b00 can clear these two bits when the value equals to 2'b11.Writes other value has no effect and returns 2'b11.

LDO3P3TC[1:0] — LDO3P3 Trimming Value

IfLDO3P3TE is set, LDO3P3TC[1:0] is the loaded Value from the factory information area, otherwise the value is 2'b00. Software can change it after power on reset

LDO3P3PD — LDO3P3 Power Down Control bit

1 = LDO3P3 will be power-down

0 = LDO3P3 won't be power-down.

LDO3P3BYP — LDO3P3 Bypass Mode Control bit

1 = LDO3P3 will be work in bypass mode

0 = LDO3P3 will be work in normal mode

©Levetop Electronics Co., Ltd.

17. LDMA Controller

17.1 Introduction

The LDMA controller is used to directly transfer data between memory and peripheral by request-acknowledge. The LDMA module provides three channels that allow byte, halfword and word transfers.

17.2 Features

The LDMA controller features include:

- Five independently programmable LDMA controller channels
- Fixed priority from channel 0 to channel 4
- Programmable memory base address
- Programmable byte counter
- Programmable endian configuration
- Programmable transfer size
- Multiple peripheral select
- Support hardware request-acknowledge

17.3 Low-Power Mode Operation

The LDMA controller is not affected by any low-power modes. CPU can stop LDMAC by setting the corresponding module stop bit in Clock Module.

17.4 Block Diagram

The LDMAC can transfer data from Memory to peripheral or from peripheral to Memory. The peripheral can be SPI, ADC.

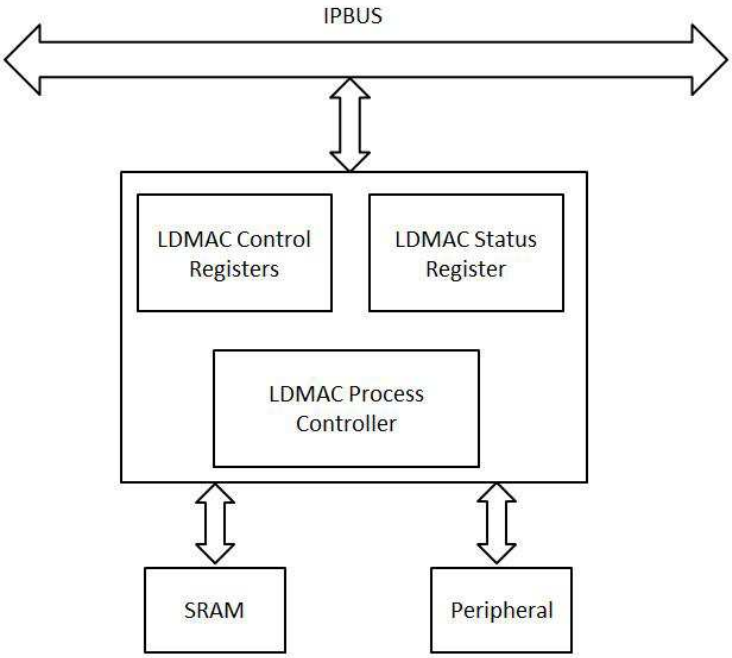


Figure 17–1: LDMAC Block Diagram

©Levetop Electronics Co., Ltd.

17.5 Module Memory Map (Base: 0x4000_0000)

Table 17–1 shows the LDMAC register memory map.

Table 17–1: Register Memory Map

Address Offset	Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	Access[1,2]
0x0000	LDMA Status Register (LDMASR)				S/U
0x0004	LDMA Memory Base Address Register of Channel0(LDMAMBAR0)				S/U
0x0008	LDMA Byte Count Register of Channel0(LDMABCR0)				S/U
0x000c	LDMA Control Register of Channel0(LDMACR0)				S/U
0x0010	LDMA Memory Base Address Register of Channel1(LDMAMBAR1)				S/U
0x0014	LDMA Byte Count Register of Channel1(LDMABCR1)				S/U
0x0018	LDMA Control Register of Channel1(LDMACR1)				S/U
0x001c	LDMA Memory Base Address Register of Channel2(LDMAMBAR2)				S/U
0x0020	LDMA Byte Count Register of Channel2(LDMABCR2)				S/U
0x0024	LDMA Control Register of Channel2(LDMACR2)				S/U
0x0028	LDMA Memory Base Address Register of Channel3(LDMAMBAR3)				S/U
0x002c	LDMA Byte Count Register of Channel3(LDMABCR3)				S/U
0x0030	LDMA Control Register of Channel3(LDMACR3)				S/U
0x0034	LDMA Memory Base Address Register of Channel4(LDMAMBAR4)				S/U
0x0038	LDMA Byte Count Register of Channel4(LDMABCR4)				S/U
0x003c	LDMA Control Register of Channel4(LDMACR4)				S/U

Notes:

1. S = CPU supervisor mode access only.
2. User mode accesses to supervisor-only address locations have no effect and result in a cycle termination transfer error.

©Levetop Electronics Co., Ltd.

17.6 Register Descriptions

17.6.1 LDMA Status Register (LDMASR)

Address Offset: 0x0000 through 0x0003

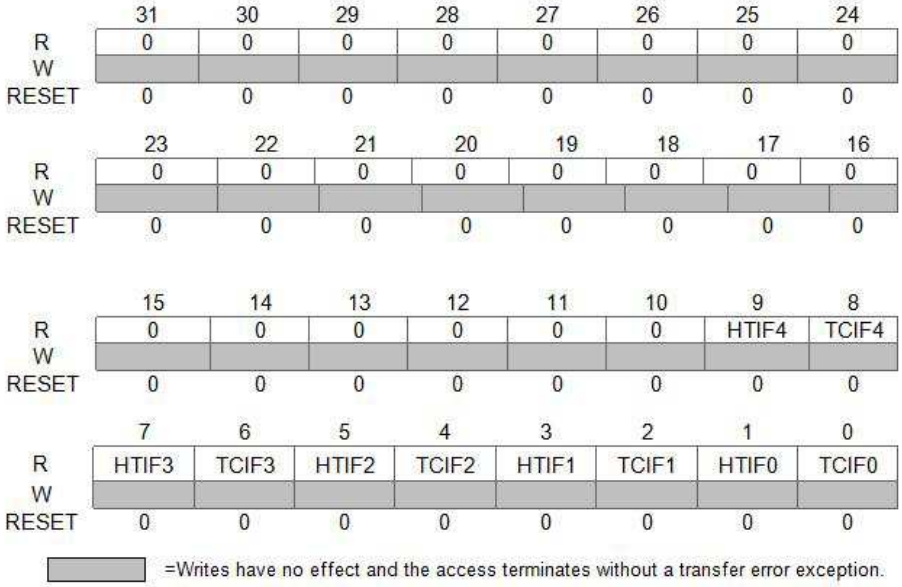


Figure 17–2: LDMA Status Register (LDMASR)

HTIFx — Channelx Half Transfer Flag
This bit is set by hardware when half of the bytes are transferred. Write 1 clear this bit.
1 = Channelx Transfer half of the bytes.
0 = Channelx Transfer not reach half of the bytes.

Note: HTIFx set when transfer bytes reach BCRx/2 and also be decided by PTSx.
When PTSx='b00/'b11, BCRx>'b100, HTIFx is valid.
PTSx='b10, BCRx>'b10, HTIFx is valid.
PTSx='b01, BCRx>'b1, HTIFx is valid.

TCIFx — Channelx Transfer Complete Flag
This bit is set by hardware when all of the bytes are transferred. Write 1 clear this bit.
1 = Channelx Transfer Complete.
0 = Channelx Transfer not Complete.

©Levetop Electronics Co., Ltd.

17.6.2 LDMA Memory Base Address Register (LDMAMBARx)

MBARx keep the initially programmed value during transfer operation .The MBARx will not be loaded to memory address counter of channelx until channelx start transfer. In circular mode, after the last transfer, the MBARx will reload to memory address counter automatically.

Address Offset: 0x0004+n*0xc, where n=0 to 4

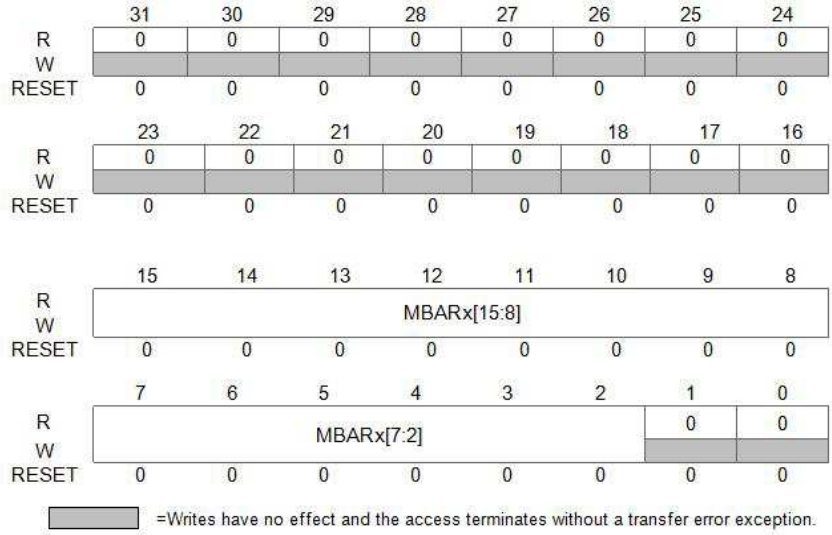


Figure 17–3: LDMA Memory Base Address Register of Channelx (LDMAMBARx)

17.6.3 LDMA Byte Count Register (LDMABCRx)

BCRx register bits contain the number of bytes yet to be transferred of channelx. When channelx transfer start, BCRx will be loaded into the internal byte counter. Internal byte counter decrements on the successful completion of a transfer and decrements by 1.2 or 4 for byte, half-word or word transfer respectively. BCRx signal the internal byte counter value when channelx is transferring.

Once the channelx transfer complete, the internal byte counter can either stay at zero or reloaded with the BCRx value previous programmed automatically (in circular mode).

Address Offset: 0x0008+n*0xc, where n=0 to 2

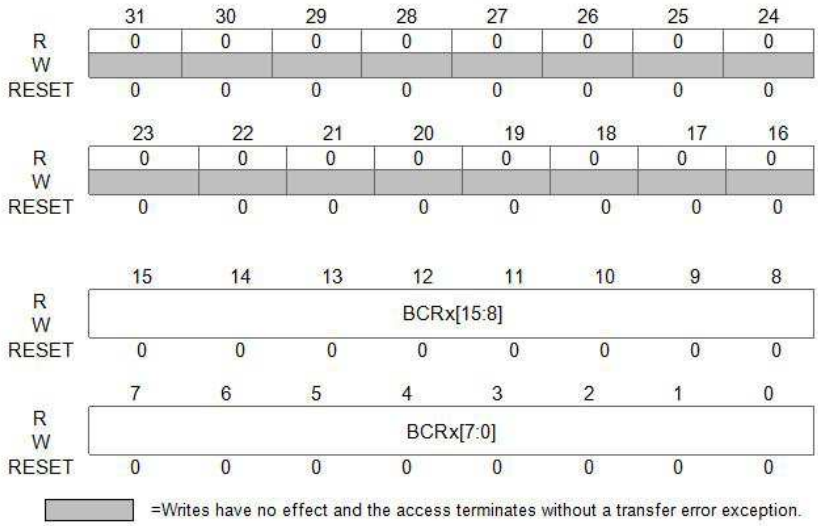


Figure 17–4: LDMA Byte Count Register of Channelx (LDMABCRx)

©Levetop Electronics Co., Ltd.

17.6.4 LDMA Control Register (LDMACRx)

Address Offset: 0x000c+n*0xc, where n=0 to 3

	31	30	29	28	27	26	25	24
R	BUSYx	CEx	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	HTIEx	TCIEx
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	PSx[1:0]		EHRx
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	CIRCx	MINCx	MDECx	TDx	PTSx[1:0]		LBFEx	STARTx
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 17–5: LDMA Control Register of Channelx(LDMACRx)

STARTx — Channelx Start Bit

The STARTx bit signals the channelx is available .When clear STARTx bit, the channelx is disabled.

1 = The channelx is available

0 = The channelx is not available

©Levetop Electronics Co., Ltd.

Note:
Before set STARTx, please config LDMACRx other bits first. Please set STARTx at last to start LDMA transfer.

LBFEx — Left Byte Transfer First Enable
1 = Left byte transfer first
0 = Right byte transfer first

PTSx[1:0] — Peripheral Transfer Size

Table 17–2: Peripheral Transfer Size

PTS[1:0]	Transfer Size
00	Word
01	Byte
10	Halfword
11	Reserved

TDx— Transfer Direction
The TDx bit determines the direction of transfer
1 = Transfer from Peripheral to Memory.
0 = Transfer from Memory to Peripheral.

MINCx — Memory Address Increment

MDECx — Memory Address Decrement
The MINCx and MDECx bits determine whether the memory address increment or decrement after a successful transfer.

Table 17–3: MINCx/MDECx Description

MINCx	MDECx	Description
0	0	No Change to Memory Address
0	1	Memory Address Decrement
1	0	Memory Address Increment
1	1	No Change to Memory Address

CIRCx — Circular Mode Enable
1 = Circular mode enable
0 = Circular mode disable

EHRx — Enable Hardware Trigger Mode
The EHRx bit enable the channelx by hardware trigger. If EHRx=0, as soon as the channel is started, it can serve any DMA request from the peripheral connected on the channel. If EHRx=1, it can't serve any DMA request from the peripheral until the channelx hardware send trigger signal to LDMA.
1 = Enable Hardware trigger mode
0 = Disable Hardware trigger mode

©Levetop Electronics Co., Ltd.

Table 17–4: Hardware Trigger Assignment of The Channels

Channel Number	Hardware Trigger
0	
1	
2	PIT3
3	PIT1
3	PIT2

PSx[1:0] — Peripheral Select
The PS[1:0] bits determines which peripheral to be selected.

Note: PS[1:0] should be set to 2'b00

Table 17–5: Peripheral Assignment of The Channels

Channel Number	PS[1:0]	Peripheral
0	0	ADC Read
1	0	SCM Channel0 Read
2	0	SCM Channel0 Write
3	0	SCM Channel1 Read
4	0	SCM Channel1 Write

TCIE — Transfer Complete Interrupt Enable Bit
This TCIE bit enable the corresponding channel TCIFx flag of LDMASR to generate interrupt request.
1 = The corresponding TCIF interrupt request enabled
0 = The corresponding TCIF interrupt request disabled

HTIE — Half Memory Reach Interrupt Enable Bit
This HMRIE bit enable the corresponding channel HTIFx flag of LDMASR to generate interrupt request.
1 = The corresponding HTIF interrupt request enabled
0 = The corresponding HTIF interrupt request disabled

CEx — Configuration Error Status
The CE bit set when the configuration errors happen. When PTSx=2'b00 and BCRx[1:0]!=2'b00,then CEx set to 1.When PTSx=2'b10 and BCRx[0]=1'b1,then CEx set to 1.This bit set automatically.
1 = Configuration error happen.
0 = Configuration error not happen.

Note:
Please check CEx after configuring the LDMACRx, if CEx=1, please reconfigure the LDMACRx with right value.

BUSYx — Channelx transfer busy
The busy bit signals the channelx transfer busy. This bit will be set and clear automatically.
1 = Channelx transfer busy.
0 = Channelx is idle.

©Levetop Electronics Co., Ltd.

17.7 Function Description

17.7.1 Circular Mode

Circular mode is available to handle circular buffers and continuous data flows (e.g. ADC scan mode). This feature can be enabled using the CIRC bit in the LDMACRx register. When circular mode is activated, the number of data to be transferred is automatically reloaded with the initial value programmed during the channel configuration phase, and the DMA requests continue to be served.

17.7.2 Channel Configuration Procedure

The following sequence should be followed to configure a LDMA channelx (where x is the channel number).

1. Set the memory address in the LDMAMARx register. The data will be written to or read from this memory after the peripheral event.
2. Configure the total number of data to be transferred in the LDMABCRx register. After each peripheral event, this value will be decremented.
3. Configure the LDMACRx register except STARTx bit. And Check CEx bit in LDMACRx, if CEx=1, please reconfigure the LDMACRx register.
4. Activate the channelx by setting the STARTx bit in the LDMACRx register

If EHRx=0, as soon as the channel is started, it can serve any DMA request from the peripheral connected on the channel. If EHRx=1, it can't serve any DMA request from the peripheral until the channelx hardware send trig signal to LDMA.

Once half of the bytes are transferred, the half-transfer flag (HTIF) is set and an interrupt is generated if the Half-Transfer Interrupt Enable bit (HTIE) is set. In no-circular mode, at the end of the transfer, the Transfer Complete Flag (TCIF) is set and an interrupt is generated if the Transfer Complete Interrupt Enable bit (TCIE) is set.

©Levetop Electronics Co., Ltd.

18. Programmable Interrupt Timer Modules (PIT)

18.1 Introduction

The programmable interrupt timer (PIT) is a 16-bit timer that provides precise interrupts at regular intervals with minimal processor intervention. The timer can either count down from the value written in the modulus latch, or it can be a free-running down-counter.

18.2 Block Diagram

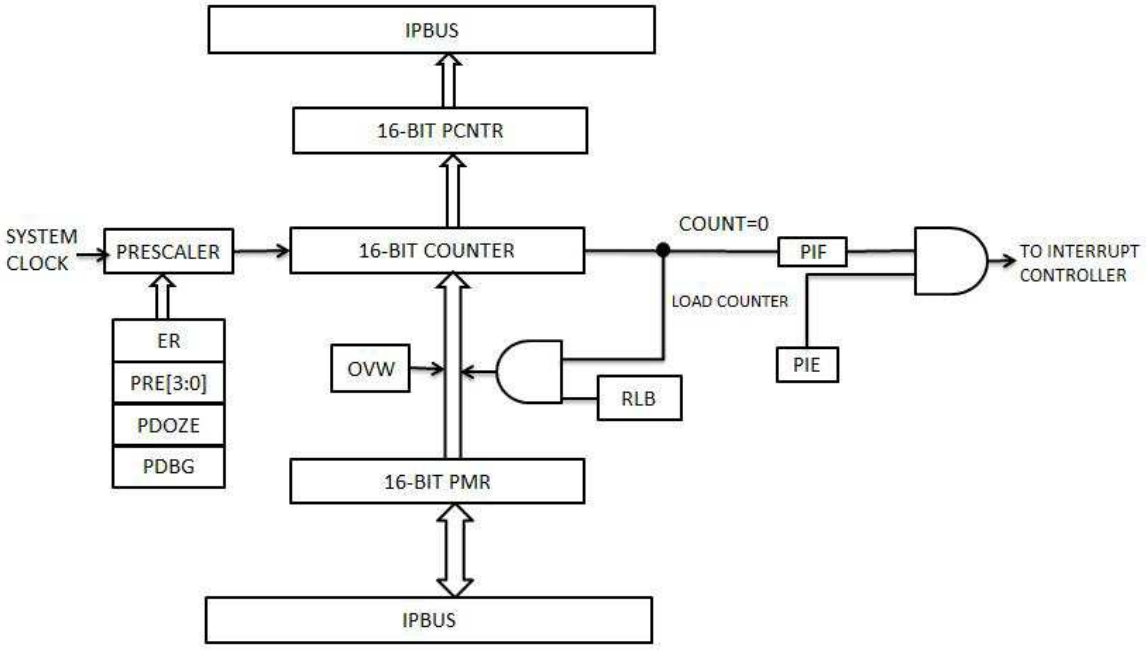


Figure 18–1: PIT Block Diagram

©Levetop Electronics Co., Ltd.

18.3 Modes of Operation

This subsection describes the three low-power modes and the debug mode.

18.3.1 Wait Mode

In wait mode, the PIT module continues to operate normally and can be configured to exit the low-power mode by generating an interrupt request.

18.3.2 Doze Mode

In doze mode with the PDOZE bit set in the PIT Control and Status Register (PCSR), PIT module operation stops. In doze mode with the PDOZE bit clear, doze mode does not affect PIT operation. When doze mode is exited, PIT operation continues from the state it was in before entering doze mode.

18.3.3 Stop Mode

In stop mode, the system clock is absent, and PIT module operation stops.

18.3.4 Debug Mode

In debug mode with the PDBG bit set in PCSR, PIT module operation stops. In debug mode with the PDBG bit clear, debug mode does not affect PIT operation. When debug mode is exited, PIT operation continues from the state it was in before entering debug mode, but any updates made in debug mode remain.

18.4 Signals

The PIT module has no off-chip signals.

©Levetop Electronics Co., Ltd.

18.5 Memory Map and Registers

This subsection describes the memory map and register structure for PIT.

18.5.1 Memory Map

(Base: 0x4004_0000, 0x4005_0000, 0x4006_0000, 0x4007_0000)
Refer to **Table 18–1** for a description of the memory map.
This device has four programmable interrupt timers.

Table 18–1: Programmable Interrupt Timer Module Memory Map

PITx Address	Bits 15-8	Bits 7-0	Access(1)
0x0	PIT Modulus Register (PMR)		S
0x2	PIT Control and Status Register (PCSR)		S
0x4	Unimplemented(2)		—
0x6	PIT Count Register (PCNTR)		S/U

18.5.2 Registers

The PIT programming model consists of these registers:

- The PIT Control and Status Register (PCSR) configures the timer's operation. See 15.5.2.1 PIT Modulus Register.
- The PIT Modulus Register (PMR) determines the timer modulus reload value. See 15.5.2.3 PIT Count Register.
- The PIT Count Register (PCNTR) provides visibility to the counter value. See 15.5.2.3 PIT Count Register.

18.5.2.1 PIT Modulus Register

The 16-bit read/write PIT Modulus Register (PMR) contains the timer modulus value for loading into the PIT counter when the count reaches 0x0000 and the RLD bit is set

When the OVW bit is set, PMR is transparent, and the value written to PMR is immediately loaded into the PIT counter. The prescaler counter is reset anytime a new value is loaded into the PIT counter and also during reset. Reading the PMR returns the value written in the modulus latch. Reset initializes PMR to 0xFFFF.

Address: 0x0000

	15	14	13	12	11	10	9	8
R W	PM15]	PM14	PM13	PM12	PM11	PM10	PM9	PM8
RESET	1	1	1	1	1	1	1	1

	7	6	5	4	3	2	1	0
R W	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
RESET	1	1	1	1	1	1	1	1

Figure 18–2: PIT Modulus Register (PMR)

©Levetop Electronics Co., Ltd.

18.5.2.2 PIT Control and Status Register

Address: 0x0002

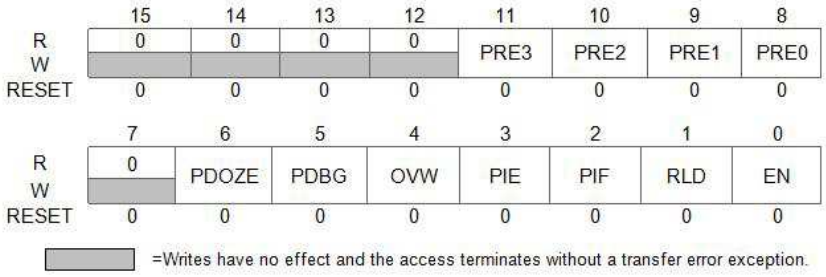


Figure 18–3: PIT Control and Status Register (PCSR)

PRE[3:0] — Prescaler Bits

The read/write PRE[3:0] bits select the system clock divisor to generate the PIT clock as **Table 18–2** shows.

To accurately predict the timing of the next count, change the PRE[3:0] bits only when the enable bit (EN) is clear. Changing the PRE[3:0] resets the prescaler counter. System reset and the loading of a new value into the counter also reset the prescaler counter. Setting the EN bit and writing to PRE[3:0] can be done in this same write cycle. Clearing the EN bit stops the prescaler counter.

Table 18–2: Prescaler Select Encoding

PRE[3:0]	System Clock Divisor
0000	1
0001	2
0010	4
0011	8
0100	16
0101	32
0110	64
0111	128
1000	256
1001	512
1010	1,024
1011	2,048
1100	4,096
1101	8,192
1110	16,384
1111	32,768

©Levetop Electronics Co., Ltd.

PDOZE — Doze Mode Bit

The read/write PDOZE bit controls the function of the PIT in doze mode. Reset clears PDOZE.

1 = PIT function stopped in doze mode

0 = PIT function not affected in doze mode

When doze mode is exited, timer operation continues from the state it was in before entering doze mode.

PDBG — Debug Mode Bit

The read/write PDBG bit controls the function of the PIT in debug mode. Reset clears PDBG.

1 = PIT function stopped in debug mode

0 = PIT function not affected in debug mode

During debug mode, register read and write accesses function normally. When debug mode is exited, timer operation continues from the state it was in before entering debug mode, but any updates made in debug mode remain.

Note:

Changing the PDBG bit from 1 to 0 during debug mode starts the PIT timer. Likewise, changing the PDBG bit from 0 to 1 during debug mode stops the PIT timer.

OVW — Overwrite Bit

The read/write OVW bit enables writing to PMR to immediately overwrite the value in the PIT counter

1 = Writing PMR immediately replaces value in PIT counter.

0 = Value in PMR replaces value in PIT counter when count reaches 0x0000.

PIE — PIT Interrupt Enable Bit

The read/write PIE bit enables the PIF flag to generate interrupt requests.

1 = PIF interrupt requests enabled

0 = PIF interrupt requests disabled

PIF — PIT Interrupt Flag

The read/write PIF flag is set when the PIT counter reaches 0x0000. Clear PIF by writing a 1 to it or by writing to PMR. Writing 0 has no effect. Reset clears PIF

1 = PIT count has reached 0x0000.

0 = PIT count has not reached 0x0000.

RLD — Reload Bit

The read/write RLD bit enables loading the value of PMR into the PIT counter when the count reaches 0x0000.

1 = Counter reloaded from PMR on count of 0x0000

0 = Counter rolls over to 0xFFFF on count of 0x0000

EN — PIT Enable Bit

The read/write EN bit enables PIT operation. When the PIT is disabled, the counter and prescaler are held in a stopped state.

1 = PIT enabled

0 = PIT disabled

©Levetop Electronics Co., Ltd.

18.5.2.3 PIT Count Register

The 16-bit, read-only PIT Control Register (PCNTR) contains the counter value. Reading the 16-bit counter with two 8-bit reads is not guaranteed to be coherent. Writing to PCNTR has no effect, and write cycles are terminated normally.

Address: 0x0006

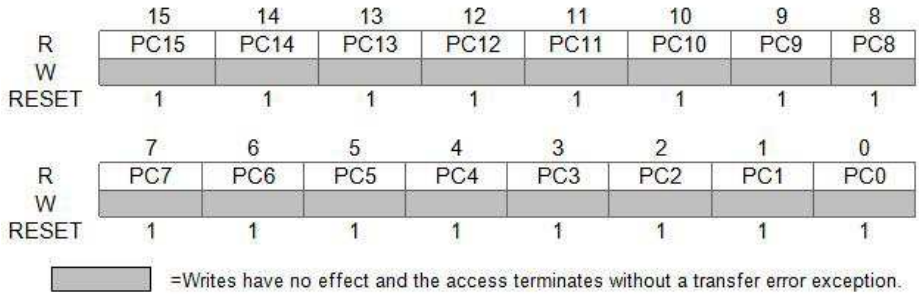


Figure 18–4: PIT Count Register (PCNTR)

©Levetop Electronics Co., Ltd.

18.6 Functional Description

This subsection describes the PIT functional operation.

18.6.1 Set-and-Forget Timer Operation

This mode of operation is selected when the RLD bit in the PCSR register is set.

When the PIT counter reaches a count of 0x0000, the PIF flag is set in PCSR. The value in the modulus latch is loaded into the counter, and the counter begins decrementing toward 0x0000. If the PIE bit is set in PCSR, the PIF flag issues an interrupt request to the CPU.

When the OVW bit is set in PCSR, the counter can be directly initialized by writing to PMR without having to wait for the count to reach 0x0000.

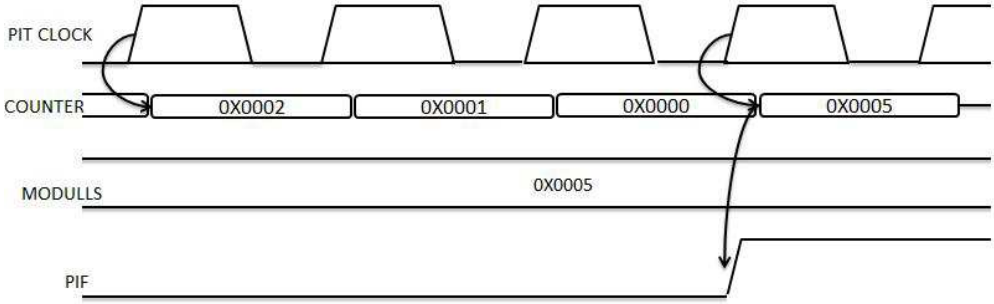


Figure 18–5: Counter Reloading from the Modulus Latch

18.6.2 Free-Running Timer Operation

This mode of operation is selected when the RLD bit in PCSR is clear. In this mode, the counter rolls over from 0x0000 to 0xFFFF without reloading from the modulus latch and continues to decrement.

When the counter reaches a count of 0x0000, the PIF flag is set in PCSR. If the PIE bit is set in PCSR, the PIF flag issues an interrupt request to the CPU.

When the OVW bit is set in PCSR, the counter can be directly initialized by writing to PMR without having to wait for the count to reach 0x0000.

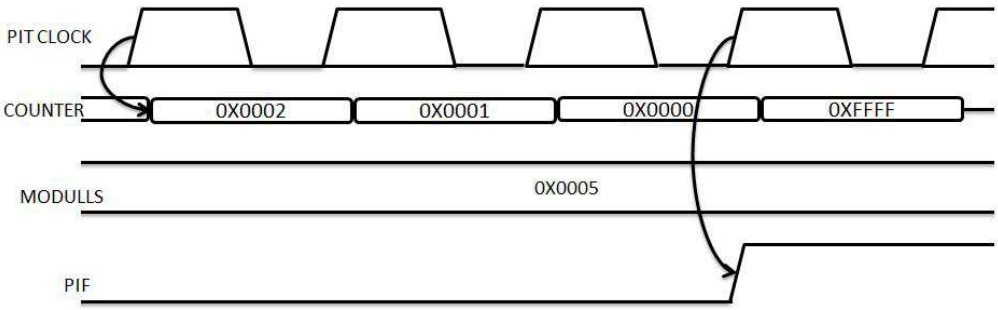


Figure 18–6: Counter in Free-Running Mode

©Levetop Electronics Co., Ltd.

18.6.3 Timeout Specifications

The 16-bit PIT counter and prescaler supports different timeout periods. The prescaler divides the system clock as selected by the PRE[3:0] bits in PCSR. The PM[15:0] bits in PMR select the timeout period.

Timeout period = PRE[3:0] ×(PM[15:0] + 1) clocks

18.7 Interrupt Operation

Table 18–3 lists the interrupt requests generated by the PIT.

Table 18–3: PIT Interrupt Requests

Interrupt Request	Flag	Enable Bit
Timeout	PIF	PIE

The PIF flag is set when the PIT counter reaches 0x0000. The PIE bit enables the PIF flag to generate interrupt requests. Clear PIF by writing a 1 to it or by writing to the PMR.

©Levetop Electronics Co., Ltd.

19. Watchdog Timer Module

19.1 Introduction

The watchdog timer is a 16-bit timer used to help software recover from runaway code or give an interrupt when the operation has run longer than expected. The watchdog timer has a free-running down-counter (watchdog counter) that generates a reset or interrupts on underflow. To prevent a reset, software must periodically restart the countdown by servicing the watchdog.

19.2 Modes of Operation

This subsection describes the operation of the watchdog timer in low-power modes and debug mode of operation.

19.2.1 Wait Mode

In wait mode with the WAIT bit set in the Watchdog Control Register (WCR), watchdog timer operation stops. In wait mode with the WAIT bit clear, the watchdog timer continues to operate normally.

19.2.2 Doze Mode

In doze mode with the DOZE bit set in WCR, watchdog timer module operation stops. In doze mode with the DOZE bit clear, the watchdog timer continues to operate normally.

19.2.3 Stop Mode

In stop mode with the STOP bit set in WCR, watchdog operation stops in stop mode. When stop mode is exited, the watchdog operation continues operation from the state it was in prior to entering stop mode. In stop mode with the STOP bit clear, the watchdog timer continues to operate normally.

19.2.4 Debug Mode

In debug mode with the DBG bit set in WCR, watchdog timer module operation stops. In debug mode with the DBG bit clear, the watchdog timer continues to operate normally. When debug mode is exited, watchdog timer operation continues from the state it was in before entering debug mode, but any updates made in debug mode remain.

©Levetop Electronics Co., Ltd.

19.3 Block Diagram

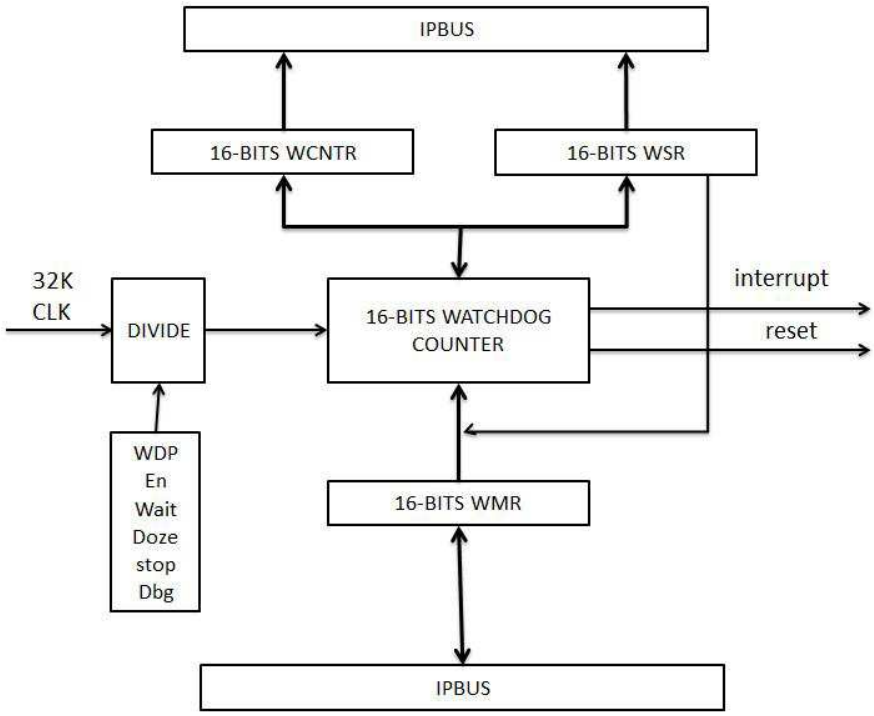


Figure 19–1: Watchdog Timer Block Diagram

19.4 Signals

The watchdog timer module has no off-chip signals.

©Levetop Electronics Co., Ltd.

19.5 Memory Map and Registers

This subsection describes the memory map and registers for the watchdog timer.

19.5.1 Memory Map (Base: 0x4013_0000, 0x4014_0000)

Regard **Table 19–1** to for an overview of the watchdog memory map.

Table 19–1: Watchdog Timer Module Memory Map

Offset address	Bits 15-8	Bits 7-0	Access(1)
0x0000	Watchdog Modulus Register (WMR)		S
0x0002	Watchdog Control Register (WCR)		S
0x0004	Watchdog Service Register (WSR)		S/U
0x0006	Watchdog Count Register (WCNTR)		S/U

Note:

1. S = CPU supervisor mode access only. S/U = CPU supervisor or user mode access. User mode accesses to supervisor only addresses have no effect and result in a cycle termination transfer error.

19.5.2 Registers

The watchdog timer programming model consists of these registers:

- The Watchdog Control Register (WCR) configures watchdog timer operation. See 16.5.2.2 Watchdog Control Register.
- The Watchdog Modulus Register (WMR) determines the timer modulus reload value. See 16.5.2.4 Watchdog Count Register.
- The Watchdog Count Register (WCNTR) provides visibility to the watchdog counter value. See 16.5.2.4 Watchdog Count Register.
- The Watchdog Service Register (WSR) requires a service sequence to prevent reset. See the read-only WC[15:0] field reflects the current value in the watchdog counter. Reading the 16-bit WCNTR with two 8-bit reads is not guaranteed to return a coherent value. Writing to WCNTR has no effect, and write cycles are terminated normally. This register is for watchdog work domain, so the read value maybe not stabilization, please read it time after time continuously.

©Levetop Electronics Co., Ltd.

19.5.2.1 Watchdog Modules Register

Offset Address: 0x0000

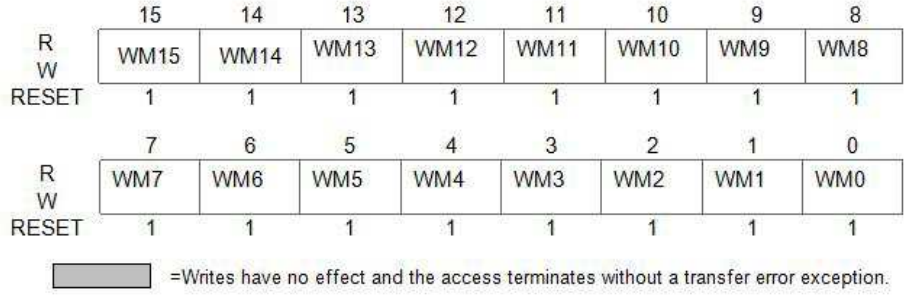


Figure 19–2: Watchdog Modulus Register (WMR)

WM[15:0] — Watchdog Modulus Field

WM[15:0] field contains the modulus that is reloaded into the watchdog counter by a service sequence. Writing to WMR immediately loads the new modulus value into the watchdog counter. The new value is also used at the next and all subsequent reloads.

Reading WMR returns the value in the modulus register. Reset initializes the WM[15:0] field to 0xFFFF.

19.5.2.2 Watchdog Control Register

The 16-bit read/write Watchdog Control Register (WCR) configures watchdog timer operation.

Offset Address: 0x0002

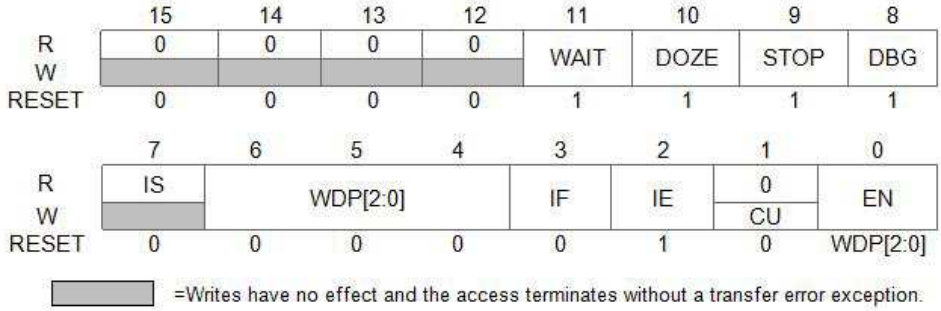


Figure 19–3: Watchdog Control Register (WCR)

WAIT — Wait Mode Bit

WAIT bit controls the function of the watchdog timer in wait mode. Reset sets WAIT.

1 = Watchdog timer stopped in wait mode

0 = Watchdog timer not affected in wait mode

DOZE — Doze Mode Bit

DOZE bit controls the function of the watchdog timer in doze mode. Reset sets DOZE.

1 = Watchdog timer stopped in doze mode

0 = Watchdog timer not affected in doze mode

©Levetop Electronics Co., Ltd.

STOP — STOP Mode Bit

STOP bit controls the function of the watchdog timer in stop mode. Reset sets STOP.

1 = Watchdog timer stopped in stop mode

0 = Watchdog timer not affected in stop mode

DBG — Debug Mode Bit

DBG bit controls the function of the watchdog timer in debug mode. During debug mode, watchdog timer registers can be written and read normally. When debug mode is exited, timer operation continues from the state it was in before entering debug mode, but any updates made in debug mode remain.

1 = Watchdog timer stopped in debug mode

0 = Watchdog timer not affected in debug mode

Note:

Changing the DBG bit from 1 to 0 during debug mode starts the watchdog timer. Changing the DBG bit from 0 to 1 during debug mode stops the watchdog timer.

EN — Watchdog Enable Bit

EN bit enables the watchdog timer.

1 = Watchdog timer enabled

0 = Watchdog timer disabled

CU — Watchdog Change Update Bit

Write One to CU bit update the WDP[2:0] and WMR to the work latch.

IE — Watchdog Interrupt Enable Bit

IE bit enables the watchdog timer interrupt mode. Once interrupt generated and the EN bit is 1, this bit will be auto clear.

1 = Watchdog timer interrupt mode enabled

0 = Watchdog timer interrupt mode disabled

IF — Watchdog Interrupt Flag Bit

Write One to this bit will clear the flag.

IS — Watchdog Clock Domain Interrupt Status Bit

This bit is read-only, if this bit is 1'b1, the status of watchdog clock domain is not cleared, so if CPU want to sleep or stop, it will be wakeup again. So before CPU want to sleep or stop, please check this bit first.

WDP[2:0] — Watchdog Timer Prescaler

The WDP[2:0] bits determine the watchdog timer prescaling when the watchdog timer is running. The different prescaling values and their corresponding time-out periods are shown in **Table 19–2.**

Table 19–2: Watchdog Timer Prescaler

WDP[2:0]	Prescaler
000	64ms
001	32ms
010	16ms
011	8ms
100	4ms
101	2ms
110	1ms
111	0.5ms

©Levetop Electronics Co., Ltd.

19.5.2.3 Watchdog Service Register

When the watchdog timer is enabled, writing 0x5555 and then 0xAAAA to the Watchdog Service Register (WSR) before the watchdog counter times out prevents a reset. If WSR is not serviced before the timeout, the watchdog timer sends a signal to the reset controller or interrupt controller module and asserts a system reset or interrupt.

Both writes must occur in the order listed before the timeout, but any number of instructions can be executed between the two writes. However, writing any value other than 0x5555 or 0xAAAA to WSR resets the servicing sequence, requiring both values to be written to keep the watchdog timer from causing a reset.

Offset Address: 0x0004

	15	14	13	12	11	10	9	8
R W	WS15	WS14	WS13	WS12	WS11	WS10	WS9	WS8
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R W	WS7	WS6	WS5	WS4	WS3	WS2	WS1	WS0
RESET	0	0	0	0	0	0	0	0

Figure 19–4: Watchdog Service Register (WSR)

19.5.2.4 Watchdog Count Register

Offset Address: 0x0006

	15	14	13	12	11	10	9	8
R	WC15	WC14	WC13	WC12	WC11	WC10	WC9	WC0
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	WC7	WC6	WC5	WC4	WC3	WC2	WC1	WC0
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 19–5: Watchdog Count Register (WCNTR)

WC[15:0] — Watchdog Count Field

The read-only WC[15:0] field reflects the current value in the watchdog counter. Reading the 16-bit WCNTR with two 8-bit reads is not guaranteed to return a coherent value. Writing to WCNTR has no effect, and write cycles are terminated normally. This register is for watchdog work domain, so the read value maybe not stabilization, please read it time after time continuously

©Levetop Electronics Co., Ltd.

20. Edge Port Module (EPORT)

20.1 Introduction

The edge port module (EPORT) has eight external interrupt pins. Each pin can be configured individually as a low level-sensitive interrupt pin, an edge-detecting interrupt pin (rising edge, falling edge, or both), or a general-purpose input/output/ (I/O) pin. See **Figure 20–1.**

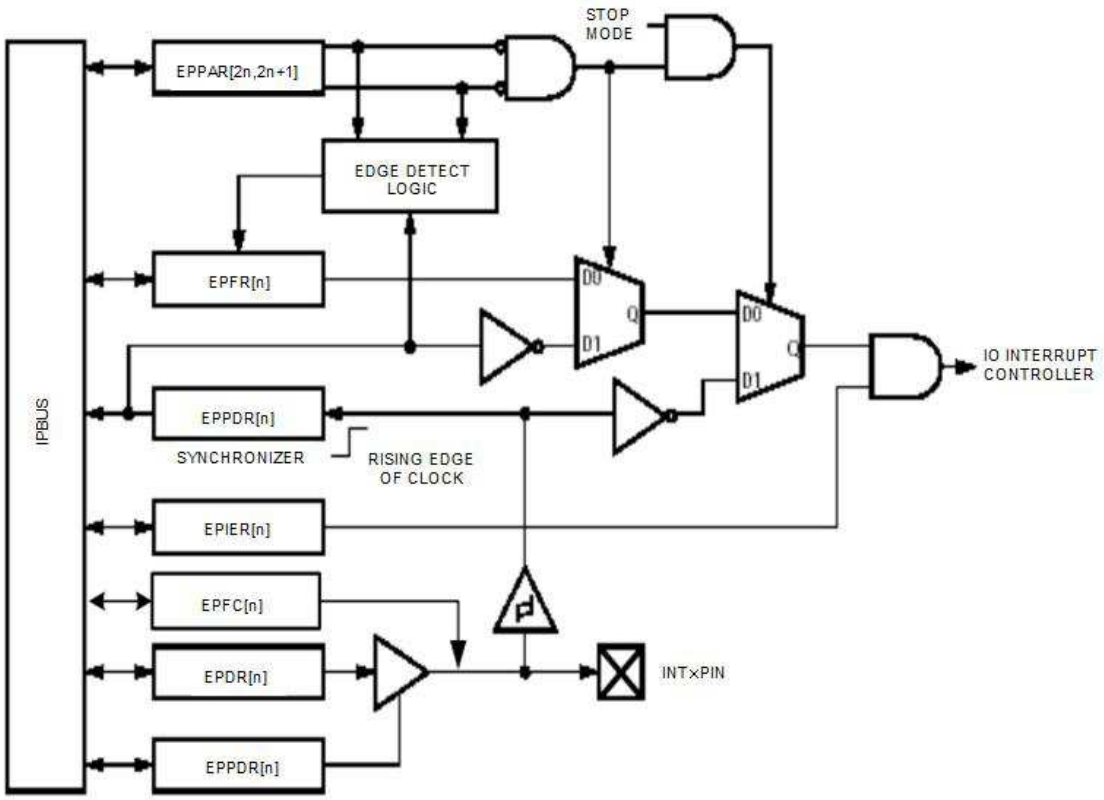


Figure 20–1: EPORT Block Diagram

©Levetop Electronics Co., Ltd.

20.2 Low-Power Mode Operation

This subsection describes the operation of the EPORT module in low-power modes.

20.2.1 Wait and Doze Modes

In wait and doze modes, the EPORT module continues to operate normally and may be configured to exit the low-power modes by generating an interrupt request on either a selected edge or a low level on an external pin.

20.2.2 Stop Mode

In stop mode, there are no clocks available to perform the edge-detect function. Only the level-detect logic is active (if configured) to allow any low level on the external interrupt pin to generate an interrupt (if enabled) to exit stop mode.

Note:
The input pin synchronizer is bypassed for the level-detect logic since no clocks are available.

20.3 Interrupt/General-Purpose I/O Pin Descriptions

All pins default to general-purpose input pins at reset. The pin value is synchronized to the rising edge of CLKOUT when read from the EPORT Pin Data Register (EPPDR). The values used in the edge/level detect logic are also synchronized to the rising edge of CLKOUT. These pins use Schmitt triggered input buffers which have built in hysteresis designed to decrease the probability of generating false edge-triggered interrupts for slow rising and falling input signals.

©Levetop Electronics Co., Ltd.

20.4 Memory Map and Registers

This subsection describes the memory map and register structure.

20.4.1 Memory Map (Base: 0x400f_0000, 0x4010_0000)

Refer to **Table 20–1** for a description of the EPORT memory map.

Table 20–1: Module Memory Map

Address Offset	Bits 15-8	Bits 7-0	Access(1)
0x0000	EPORT Data Direction Register (EPDDR)	EPORT Interrupt Enable Register (EPIER)	S
0x0002	EPORT Pin Assignment Register (EPPAR)		S
0x0004	EPORT Flag Register (EPFR)	EPORT Pin Pull-up enable Register (EPPUE)	S/U
0x0006	EPORT Data Register (EPDR)	EPORT Pin Data Register (EPPDR)	S/U
0x0008	EPORT Digital Filter Control Register(EPFC)	EPORT Bit Set Register (EPBSR)	S
0x000a	EPORT Level Polarity Register (EPLPR)	EPORT Open Drain Register (EPODE)	S
0x000c	Reversed		S
0x000e	EPORT Bit Clear Register (EPBCR)		S

Note:

1. S = CPU supervisor mode access only. S/U = CPU supervisor or user mode access. User mode accesses to supervisor only addresses have no effect and result in a cycle termination transfer error.

20.4.2 Registers

The EPORT programming model consists of these registers:

- The EPORT Pin Assignment Register (EPPAR) controls the function of each pin individually.
- The EPORT Data Direction Register (EPDDR) controls the direction of each one of the pins individually.
- The EPORT Interrupt Enable Register (EPIER) enables interrupt requests for each pin individually.
- The EPORT Data Register (EPDR) holds the data to be driven to the pins.
- The EPORT Pin Data Register (EPPDR) reflects the current state of the pins.
- The EPORT Flag Register (EPFR) individually latches EPORT edge events.
- The EPORT Pin Pull-up Enable Register (EPPUE) controls the pull-up of each one of the pins individually
- The EPORT Level Polarity Register (EPLPR) controls the level polarity of each one of the pins for level-sensitive.
- The EPORT Open Drain Enable Register (EPODE) controls the Open Drain of each one of the pins for output individually.
- The EPORT Digital Filter Control Register (EPFC) enables the filter and control the width of input pulse will be filtered.

©Levetop Electronics Co., Ltd.

20.4.2.1 Edge Port Interrupt Enable Register

Address Offset: 0x0000

	7	6	5	4	3	2	1	0
R W	EPIE7	EPIE6	EPIE5	EPIE4	EPIE3	EPIE2	EPIE1	EPIE0
RESET	1	1	1	1	1	1	1	1

Figure 20–2: EPORT Port Interrupt Enable Register (EPIER)

EPIE[7:0] — Edge Port Interrupt Enable Bits

The read/write EPIE[7:0] bits enable EPORT interrupt requests. If a bit in EPIER is set, EPORT generates an interrupt request when:

- The corresponding bit in the EPORT Flag Register (EPFR) is set or later becomes set, or
- The corresponding pin level is low and the pin is configured for level-sensitive operation

Clearing a bit in EPIER negates any interrupt request from the corresponding EPORT pin. Reset clears EPIE[7:0].

1 = Interrupt requests from corresponding EPORT pin enabled
0 = Interrupt requests from corresponding EPORT pin disabled

20.4.2.2 EPORT Data Direction Register

Address Offset: 0x0001

	7	6	5	4	3	2	1	0
R W	EPDD7	EPDD6	EPDD5	EPDD4	EPDD3	EPDD2	EPDD1	EPDD0
RESET	0	0	0	0	0	0	0	0

Figure 20–3: EPORT Data Direction Register (EPDDR)

EPDD[7:0] — Edge Port Data Direction Bits

Setting any bit in the EPDDR configures the corresponding pin as an output. Clearing any bit in EPDDR configures the corresponding pin as an input. Pin direction is independent of the level/edge detection configuration. Reset clears EPDD[7:0].

To use an EPORT pin as an external interrupt request source, its corresponding bit in EPDDR must be clear. Software can generate interrupt requests by programming the EPORT Data Register when the EPDDR selects output.

1 = Corresponding EPORT pin configured as output
0 = Corresponding EPORT pin configured as input

©Levetop Electronics Co., Ltd.

20.4.2.3 EPORT Pin Assignment Register

Address Offset: 0x0002 through 0x0003

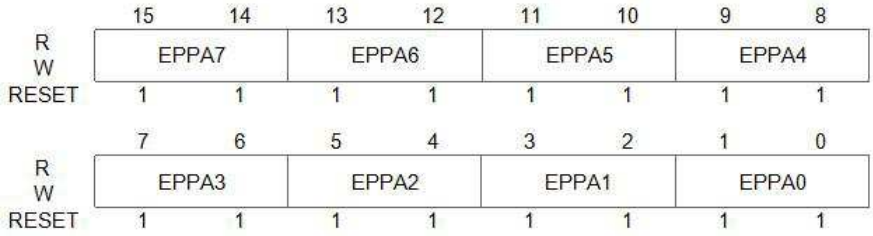


Figure 20–4: EPORT Pin Assignment Register (EPPAR)

EPPA[7:0] — EPORT Pin Assignment Select Fields

The read/write EPPAx fields configure EPORT pins for level detection and rising and/or falling edge detection as shows.

Pins configured as level-sensitive are inverted so that a logic 0 or logic 1 on the external pin represents a valid interrupt request. Level-sensitive interrupt inputs are not latched. To guarantee that a level-sensitive interrupt request is acknowledged, the interrupt source must keep the signal asserted until acknowledged by software. Level sensitivity must be selected to bring the device out of stop mode with an INTx interrupt.

Pins configured as edge-triggered are latched and need not remain asserted for interrupt generation. A pin configured for edge detection is monitored regardless of its configuration as input or output.

Table 20–2: EPPAx Field Settings

EPPAx	Pin Configuration
00	Pin INTx level-sensitive
01	Pin INTx rising edge triggered
10	Pin INTx falling edge triggered
11	Pin INTx both falling edge and rising edge triggered

Interrupt requests generated in the EPORT module can be masked by the interrupt controller module. EPPAR functionality is independent of the selected pin direction.

Reset clears the EPPAx fields.

©Levetop Electronics Co., Ltd.

20.4.2.4 EPORT Pin Pull-up Enable Register

Address Offset: 0x0004

	7	6	5	4	3	2	1	0
R	EPPUE	EPPUE	EPPUE	EPPUE	EPPUE	EPPUE	EPPUE	EPPUE
W	7	6	5	4	3	2	1	0
RESET	1	1	1	1	1	1	1	1

Figure 20–5: EPORT Pin Pull-up Enable Register (EPPUE)

EPPUE[7:0] — Edge Port Pin Pull-up enable Bits

Setting any bit in the EPPUE configures the corresponding pin to enable pull-up. Clearing any bit in EPPUE configures the corresponding pin disable pull-up. Reset sets EPPUE[7:0].

1 = Corresponding EPORT pin configured to enable pull-up

0 = Corresponding EPORT pin configured to disable pull-up

20.4.2.5 Edge Port Flag Register

Address Offset: 0x0005

	7	6	5	4	3	2	1	0
R	EPFR7	EPFR6	EPFR5	EPFR4	EPFR3	EPFR2	EPFR1	EPFR0
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 20–6: EPORT Port Flag Register (EPFR)

EPF[7:0] — Edge Port Flag Bits

When an EPORT pin is configured for edge triggering, its corresponding read/write bit in EPFR indicates that the selected edge has been detected. Reset clears EPF[7:0].

1 = Selected edge for INTx pin has been detected.

0 = Selected edge for INTx pin has not been detected.

Bits in this register are set when the selected edge is detected on the corresponding pin. A bit remains set until cleared by writing a 1 to it. Writing 0 has no effect. If a pin is configured as level-sensitive (EPPARx = 00), pin transitions do not affect this register.

©Levetop Electronics Co., Ltd.

20.4.2.6 Edge Port Pin Data Register

Address Offset: 0x0006

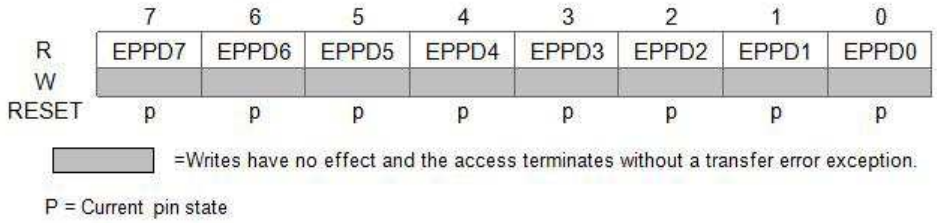


Figure 20–7: EPORT Port Pin Data Register (EPPDR)

EPPD[7:0] — Edge Port Pin Data Bits

The read-only EPPDR reflects the current state of the EPORT pins. Writing to EPPDR has no effect, and the write cycle terminates normally. Reset does not affect EPPDR.

20.4.2.7 Edge Port Data Register

Address Offset: 0x0007

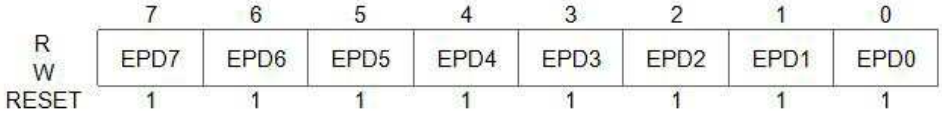


Figure 20–8: EPORT Port Data Register (EPDR)

EPD[7:0] — Edge Port Data Bits

Data written to EPDR is stored in an internal register; if any pin of the port is configured as an output, the bit stored for that pin is driven onto the pin. Reading EDPR returns the data stored in the register. Reset sets EPD[7:0].

©Levetop Electronics Co., Ltd.

20.4.2.8 EPORT Port Bit Set Register

Address Offset: 0x0008

	7	6	5	4	3	2	1	0
R	1	1	1	1	1	1	1	1
W	EPBSR 7	EPBSR 6	EPBSR 5	EPBSR 4	EPBSR 3	EPBSR 2	EPBSR 1	EPBSR 0
RESET :	0	0	0	0	0	0	0	0

Figure 20–9: EPORT Port Bit Set Register (EPBSR)

EPBSR[7:0] — EPORT Port Bit Set Register
1 = The corresponding bit of EPDR will be set;
0 = The corresponding bit of EPDR will not be effected;

20.4.2.9 EPORT Digital Filter Control Register

Address Offset: 0x0009

	7	6	5	4	3	2	1	0
R W	Filter_Width[6:0]							Filter_En
RESET	0	0	0	0	0	0	0	0

Figure 20–10: EPORT Digital Filter Control Register (EPFC)

Filter_En — EPORT Digital Filter Enable Bit
1 = EPORT digital filter enable;
0 = EPORT digital filter disable;

Filter_Width[6:0] — EPORT Digital Filter Pulse Width Select Bit
Filter_Width[6:0] determine the width of input pulse will be filtered. If the input pulse width less than (Filter_Width[6:0]+2), the pulse will be filtered.

©Levetop Electronics Co., Ltd.

20.4.2.10 EPORT Open Drain Enable Register

Address Offset: 0x000a

	7	6	5	4	3	2	1	0
R W	EPODE7	EPODE6	EPODE5	EPODE4	EPODE3	EPODE2	EPODE1	EPODE0
RESET	1	1	1	1	1	1	1	1

Figure 20–11: EPORT Open Drain Enable Register (EPODE)

EPODE[7:0] — Edge Port Open Drain enable Bits

If EPORT are configured to output, setting any bit in the EPODE configures the corresponding pin to Open Drain output. Clearing any bit in EPODE configures the corresponding pin CMOS output. Reset clears EPODE[7:0].

1 = Corresponding EPORT pin configured to Open Drain output

0 = Corresponding EPORT pin configured to CMOS output

20.4.2.11 EPORT Level Polarity Register

Address Offset: 0x000b

	7	6	5	4	3	2	1	0
R W	EPLPR7	EPLPR6	EPLPR5	EPLPR4	EPLPR3	EPLPR2	EPLPR1	EPLPR0
RESET	0	0	0	0	0	0	0	0

Figure 20–12: EPORT Level Polarity Register (EPLPR)

EPLPR[7:0] — Edge Port Level Polarity Bits

If EPORT are configured to level-sensitive, setting any bit in the EPLPR configures the corresponding pin high level-sensitive. Clearing any bit in EPLPR configures the corresponding pin low level-sensitive. Reset clears EPLPR[7:0].

1 = Corresponding EPORT pin configured to high level-sensitive

0 = Corresponding EPORT pin configured to low level-sensitive

20.4.2.12 EPORT Port Bit Clear Register

Address Offset: 0x000f

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W	EPBCR7	EPBCR6	EPBCR5	EPBCR4	EPBCR3	EPBCR2	EPBCR1	EPBCR0
RESET	0	0	0	0	0	0	0	0

Figure 20–13: EPORT Port Bit Clear Register (EPBCR)

EPBCR[7:0] — EPORT Port Bit Clear Register

1 = The corresponding bit of EPDR will be clear;

0 = The corresponding bit of EPDR will not be effected;

©Levetop Electronics Co., Ltd.

21. Sensor Controller Module (SCM)

21.1 Overview

The SCM module contains two-channel Master Serial Peripheral Interface (SPI), two separated receive buffer, two separated transmit buffer which can communicate with two SPI slaves simultaneously. But the two-channel master SPI share the same SPICLK port, which can communicate with two SPI slave synchronously

The two slave select ports can be controlled individually. Software

21.2 Features

Features include:

- SPI Master mode
- Shared MSCK ports
- Two 4 entries deep read FIFO
- Two 4 entries deep write FIFO
- Interrupt generation after 1, 2, 3, or 4 transfers
- MSB/LSB selectable • Variable baud-rate during communication
- Serial clock with programmable polarity and phase
- 16/32 bit transmit/received data width
- Byte reorader
- Controllable slave select (MSS0/1) bit

21.3 Block Diagram

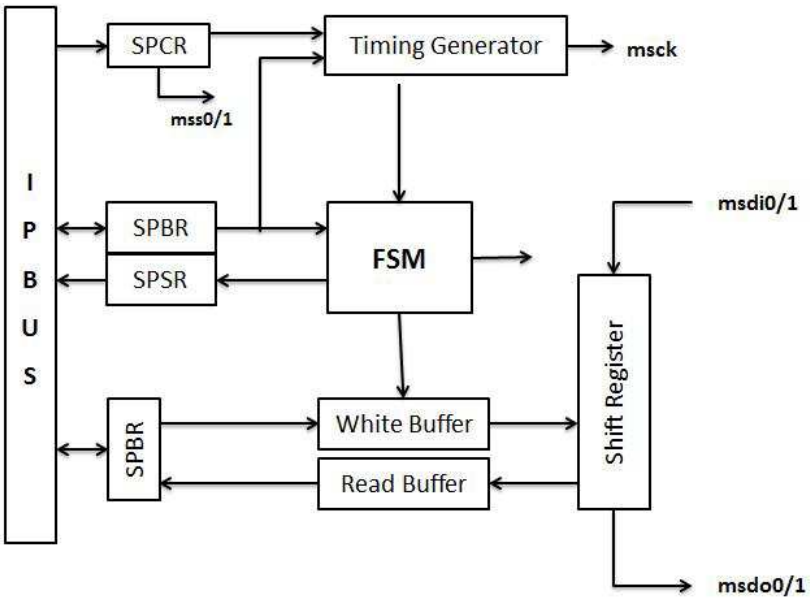


Figure 21–1: SCM Block Diagram

©Levetop Electronics Co., Ltd.

21.4 Signal Description

An overview of the signals is provided in **Table 21–1** .

Table 21–1: Signal Properties

Name	Function
MSCK	Master SPI Serial Clock
MSDI0	Master SPI Channel 0 Data In
MSDO0	Master SPI Channel 0 Data Out
MSS0	Master SPI Channel 0 select
MSDI1	Master SPI Channel 1 Data In
MSDO1	Master SPI Channel 1 Data Out
MSS1	Master SPI Channel 1 select

21.5 Memory Map and Registers (Base: 0x400c_0000)

Table 21–2 shows the SCM memory map.

Table 21–2: SCM Memory Map

Address	31:24	23:16	15:8	7:0	Access[1]
0x0000	SCM Control Register (SPCR)				S/U
0x0004	SCM Status Register (SPSR)				S/U
0x0008	SCM Channel-0 Buffer Register (SPB0R)				S/U
0x000c	SCM Channel-1 Buffer Register (SPB1R)				S/U
0x0010	SCM Baud Rate Register (SPBR)				S/U
0x0014	SCM Bit Period Select Register (SPIBPSR)				S/U
0x0018	SCMGPIOEN	SCMGPIODDR	SCMGPIODR	SCMPULLENR	S/U
0x001c	SCM DMA Control Register (SPIDMACR)				S/U

Notes:

1. S = supervisor-only access. User mode accesses to supervisor only address locations have no effect and result in a cycle termination transfer error.
2. Reading unimplemented addresses (0x001c through 0x001f) returns 0s. Writing to unimplemented addresses has no effect. Accessing unimplemented addresses does not generate an error response. SPB0R and SPB1R can only accept word access when work in 32-bit transfer width or word or half-word access when work in 16-bit transfer width, otherwise an access error will occur.

©Levetop Electronics Co., Ltd.

21.5.1 Register Description

This subsection provides a description of the SCM module registers.

21.5.1.1 SCM Control Register

Register Offset Address: 0x0000

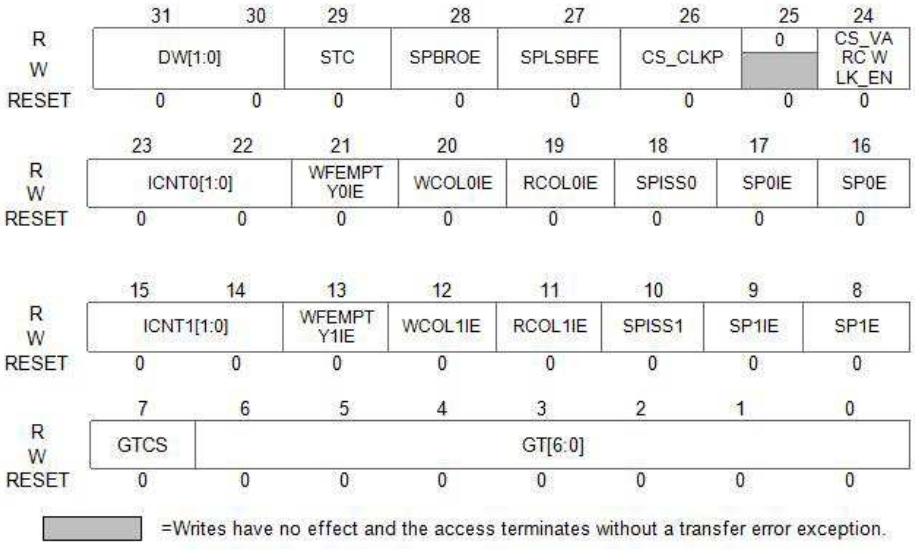


Figure 21–2: SCM Control Register (SPCR)

CS_CLKP — Clock Polarity Bit
1 = Active-low clock; MSCK idles high
0 = Active-high clock; MSCK idles low

DW — SPI transfer data width control bit
2'b11: 32-bit transfer data width
2'b10: 24-bit transfer data width
2'b01: 16-bit transfer data width
2'b00: 8-bit transfer data width

CS_VARCLK_EN — SPI variable clock mode enable bit
1 = SPI variable clock mode enabled
0 = SPI fixed clock mode enabled

ICNT0[1:0] — Interrupt Count
The Interrupt Count bits determine the SPI channel-0 transfer block size. The SPI0F bit is set after ICNT0 transfers. Thus it is possible to reduce kernel overhead due to reduced interrupt service calls. If you change ICNT0 value, you'd better clear SP0E first.
00: SPI0F is set after every completed transfer
01: SPI0F is set after every two completed transfers
10: SPI0F is set after every three completed transfers
11: SPI0F is set after every four completed transfers

©Levetop Electronics Co., Ltd.

ICNT1[1:0] — Interrupt Count

The Interrupt Count bits determine the SPI channel-1 transfer block size. The SPI0F bit is set after ICNT0 transfers. Thus it is possible to reduce kernel overhead due to reduced interrupt service calls. If you change ICNT1 value, you'd better clear SP1E first.

00: SPI1F is set after every completed transfer
01: SPI1F is set after every two completed transfers
10: SPI1F is set after every three completed transfers
11: SPI1F is set after every four completed transfers

STC — Start Transmit Control Bit

If STC is 0, SPI channel-0 write fifo not empty and spiss0 is low will start data transfer. If SPI channel-1 is enabled, SPISS1 is low and channel-1 write FIFO is not empty, then the data in SPI channel-1 write FIFO will also be transmitted. If channel-0 writes FIFO is empty, then no data will be transmitted. If STC is 1, SPI channel-1 writes FIFO not empty and spiss1 is low will start data transfer. If SPI channel-0 is enabled, SPISS1 is low and channel-0 write FIFO is not empty, then the data in SPI channel-0 write FIFO will also be transmitted. If channel-1 writes FIFO is empty, then no data will be transmitted.

1 = SPI channel-1 write FIFO not empty will start data transfer
0 = SPI channel-0 write FIFO not empty will start data transfer

SPBROE — SPI byte re-order enable control bit during transmitting data

For 32-bit transfer width with MSB first, the default transfer order is BYTE3, BYTE2, BYTE1, BYTE0, when byte re-order enabled, the transfer order is BYTE0, BYTE1, BYTE2, BYTE3.

For 32-bit transfer width with LSB first, the default transfer order is BYTE0, BYTE1, BYTE2, BYTE3, when byte re-order enabled, the transfer order is BYTE3, BYTE2, BYTE1, BYTE0.

For 16-bit transfer width with MSB first, the default transfer order is BYTE1, BYTE0, when byte re-order enabled, the transfer order is BYTE0, BYTE1.

For 16-bit transfer width with LSB first, the default transfer order is BYTE0, BYTE1, when byte re-order enabled, the transfer order is BYTE1, BYTE0.

1 = SPI byte re-order enabled
0 = SPI byte re-order disabled

SPLSBE — SPI LSB enable control bit

1 = SPI channel-0 LSB enabled
0 = SPI channel-0 MSB enabled

SPISS0 — SPI channel-0 slave select bit

If you want to transmit data through SPI channel-0, you should set MSS0 bit to logic 0 state to select slave SPI first. Even if the SPI channel-0 writing FIFO is not empty, there is no data send if MSS0 is high.

1 = SPI channel-0 slave select signal (MSS0) is in high state
0 = SPI channel-0 slave select signal (MSS0) is in low state

SPISS1 — SPI channel-1 slave select bit

If you want to transmit data through SPI channel-1, you should set MSS1 bit to logic 0 state to select slave SPI first. Even if the SPI channel-1 writes FIFO is not empty, there is no data send if MSS1 is high.

1 = SPI channel-1 slave select signal (spiss1) is in high state
0 = SPI channel-1 slave select signal (spiss1) is in low state

WFEMPTY0IE — SPI channel-0 writes FIFO empty interrupt Enable Bit

1 = SPI channel-0 write FIFO empty interrupt Enabled
0 = SPI channel-0 write FIFO empty interrupt Disabled

©Levetop Electronics Co., Ltd.

WFEMPTY1IE — SPI channel-1 writes FIFO empty interrupt Enable Bit
1 = SPI channel-1 write FIFO empty interrupt Enabled
0 = SPI channel-1 write FIFO empty interrupt Disabled

WCOL0IE — SPI channel-0 writes FIFO overrun interrupt Enable Bit
1 = SPI channel-0 write FIFO overrun interrupt Enabled
0 = SPI channel-0 write FIFO overrun interrupt Disabled

WCOL1IE — SPI channel-1 writes FIFO overrun interrupt Enable Bit
1 = SPI channel-1 write FIFO overrun interrupt Enabled
0 = SPI channel-1 write FIFO overrun interrupt Disabled

RCOL0IE — SPI channel-0 read FIFO overrun interrupt Enable Bit
1 = SPI channel-0 read FIFO overrun interrupt Enabled
0 = SPI channel-0 read FIFO overrun interrupt Disabled

RCOL1IE — SPI channel-1 read FIFO overrun interrupt Enable Bit
1 = SPI channel-1 read FIFO overrun interrupt Enabled
0 = SPI channel-1 read FIFO overrun interrupt Disabled

SP0IE — SPI channel-0 transfer done interrupt Enable Bit
When the channel-0 Interrupt Enable is set (‘1’) and the Serial transfer done Flag in the status register is set, the host is interrupted. Setting this bit while the SPI0F flag is set generates an interrupt.
1 = SPI channel-0 interrupt enabled
0 = SPI channel-0 interrupt disabled

SP1IE — SPI channel-1 transfer done interrupt Enable Bit
When the channel-1 Interrupt Enable is set (‘1’) and the Serial transfer done Flag in the status register is set, the host is interrupted. Setting this bit while the SPI1F flag is set generates an interrupt.
1 = SPI channel-1 interrupt enabled
0 = SPI channel-1 interrupt disabled

SP0E — SPI channel-0 Enable Bit
When the SPI channel-0 Enable bit is set (‘1’), the channel-0 is enabled. When it is cleared (‘0’), the channel-0 is disabled and in reset state. The channel-0 SPI only transfers data when enabled.
1 = SPI channel-0 enabled
0 = SPI channel-0 disabled and in reset state

SP1E — SPI channel-1 Enable Bit
When the SPI channel-1 Enable bit is set (‘1’), the channel-1 is enabled. When it is cleared (‘0’), the channel-1 is disabled and in reset state. The channel-1 SPI only transfers data when enabled.
1 = SPI channel-1 enabled
0 = SPI channel-1 disabled and in reset state

GTCS— Guard Timer Clock Select Bits in variable clock mode, refer to variable clock mode. Guard Timer Clock is not effective in fixed clock mode.
1 = CLK1 clock selected in variable clock mode
0 = CLK0 clock selected in variable clock mode

GT[6:0] — Guard Time Bits
The Guard Timer counter will be started after a transfer finished during the master mode. If the value of GT[6:0] is 0, the Guard Time safety will be disabled. If the value of GT[6:0] is 2, then the data will be transmitted two cycles of the selected SPI clock later after last data transfer completely.

©Levetop Electronics Co., Ltd.

21.5.1.2 SCM Status Register

Register Offset Address: 0x0004

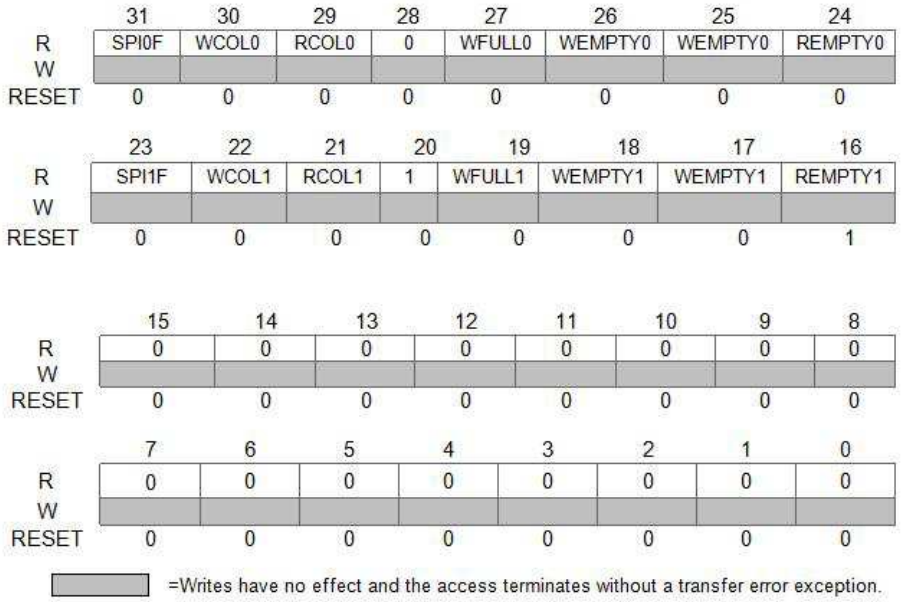


Figure 21–3: SCM Status Register (SPSR)

SPI0F — SPI channel-0 transfer done Interrupt Flag

The SPI channel-0 transfer done Interrupt Flag is set upon completion of a transfer block. If SPI0F is asserted (‘1’) and SP0IE is set, an interrupt is generated. The flag is clear by writing the status register with the SPI0F bit set (‘1’).

SPI1F — SPI channel-1 transfer done Interrupt Flag

The SPI channel-1 transfer done Interrupt Flag is set upon completion of a transfer block. If SPI1F is asserted (‘1’) and SP1IE is set, an interrupt is generated. The flag is clear by writing the status register with the SPI1F bit set (‘1’).

WCOL0 — SPI channel-0 Write FIFO Overrun

The SPI channel-0 Write FIFO Overrun flag is set when the SPI channel-0 Write Buffer register is written to, while the SPI channel-0 Write FIFO is full. The SPI channel-0 Write FIFO Overrun flag is clear by writing the status register with theWCOL0 bit set (‘1’). If the corresponding interrupt enabled, an interrupt will occur

WCOL1 — SPI channel-1 Write FIFO Overrun

The SPI channel-1 Write FIFO Overrun flag is set when the SPI channel-1 Write Buffer register is written to, while the SPI channel-1 Write FIFO is full. The SPI channel-1 Write FIFO Overrun flag is clear by writing the status register with the WCOL1 bit set (‘1’). If the corresponding interrupt enabled, an interrupt will occur

RCOL0 — SPI channel-0 Read FIFO Overrun

The SPI channel-0 Read FIFO Overrun flag is set when the SPI channel-0 Read FIFO is writing, while the SPI channel-0 Read FIFO is full. The SPI channel-0 Read FIFO Overrun flag is clear by writing the status register with the RCOL0 bit set (‘1’). If the corresponding interrupt enabled, an interrupt will occur.

RCOL1 — SPI channel-1 Read FIFO Overrun

The SPI channel-1 Read FIFO Overrun flag is set when the SPI channel-1 Read FIFO is writing, while the SPI channel-1 Read FIFO is full. The SPI channel-1 Read FIFO Overrun flag is clear by writing the status register with the RCOL1 bit set (‘1’). If the corresponding interrupt enabled, an interrupt will occur.

©Levetop Electronics Co., Ltd.

WFFULL0 — SPI channel-0 Write FIFO Full
SPI channel-0 Write FIFO Full and Write FIFO empty bits show the status of the channel-0 write FIFO.

WFFULL1 — SPI channel-1 Write FIFO Full
SPI channel-1 Write FIFO Full and Write FIFO empty bits show the status of the channel-1 write FIFO.

WFEMPTY0 — SPI channel-0 Write FIFO Empty
SPI channel-0 Write FIFO Full and Write FIFO empty bits show the status of the channel-0 write FIFO. If the corresponding interrupt enabled, an interrupt will occur.

WFEMPTY1 — SPI channel-1 Write FIFO Empty
SPI channel-1 Write FIFO Full and Write FIFO empty bits show the status of the channel-1 write FIFO. If the corresponding interrupt enabled, an interrupt will occur

RFFULL0 — SPI channel-0 Read FIFO Full
SPI channel-0 Read FIFO Full and Read FIFO empty bits show the status of the channel-0 read FIFO.

RFFULL1 — SPI channel-1 Read FIFO Full
SPI channel-1 Read FIFO Full and Read FIFO empty bits show the status of the channel-1 read FIFO.

RFEMPTY0 — SPI channel-0 Read FIFO Empty
SPI channel-0 Read FIFO Full and Read FIFO empty bits show the status of the channel-0 read FIFO.

RFEMPTY1 — SPI channel-1 Read FIFO Empty
SPI channel-1 Read FIFO Full and Read FIFO empty bits show the status of the channel-1 read FIFO.

©Levetop Electronics Co., Ltd.

21.5.1.3 SCM channel-0 Buffer Register

Register Address: 0x0008

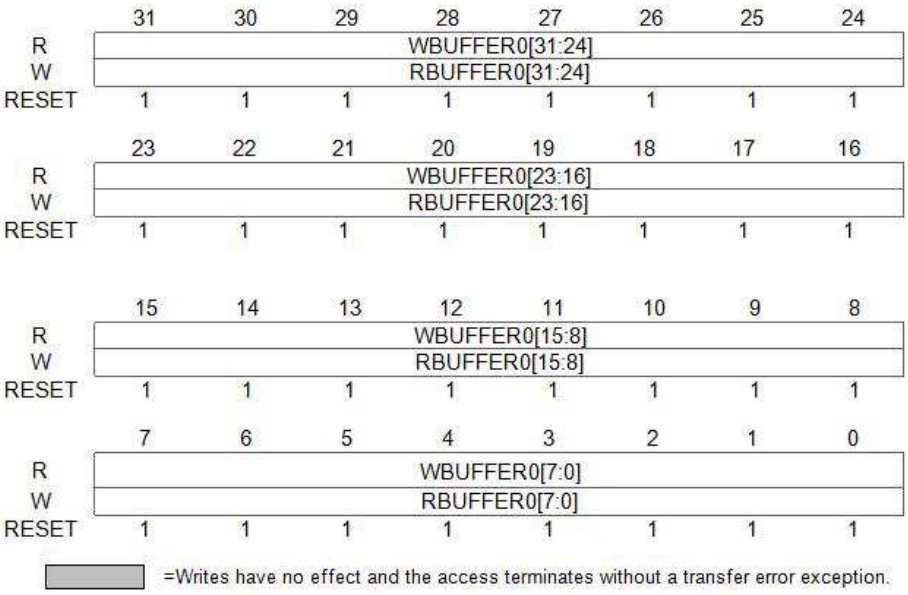


Figure 21–4: SCM Channel-0 Buffer Register (SPB0R)

WBUFFER0[31:0]

The WBUFFER0 is a 4-depth FIFO belongs to SPI channel-0. Writing to the WBUFFER0 adds the data to the channel-0 Write FIFO when not full. Writing to the WBUFFER0 while the channel-0 FIFO is full sets the Write FIFO Collision [WCOL0] bit and won't overwrite the data in Write FIFO. When the SPI channel-0 Enable [SP0E] bit is cleared ('0'), the channel-0 Read Buffer will be reset. When the [SP0E] bit is set ('1'),SPISS0 is low and the write buffer is not empty, the channel-0 initiates SPI transfers if STC is 0. When the transfer is initiated, the data byte is shift out from the channel-0 Write FIFO. When SPI transfer data width is 16-bits, the WBUFFER0[15:0] is the entry of the valid transfer data.

If data transfer is 32 bits (DW is 3), this register can only accept word (32-bit) access, and otherwise an access error will occur.

If data transfer is 24 bits (DW is 2), this register can only accept word (32-bit) access, and otherwise an access error will occur.

If data transfer is 16 bits (DW is 1), this register can accept word (32-bit) or half-word(16-bit) access, and byte access will cause an access error.

If data transfer is 8 bits (DW is 0), this register can accept word (32-bit) or half-word(16-bit) access or byte access.

RBUFFER0[31:0]

The RBUFFER0 is a 4-depth FIFO. When SPI channel-0 Enable [SP0E] bit is cleared ('0'), the channel-0 Read Buffer will be reset. When an SPI channel-0 transfer is finished, the received data byte is added to the channel-0 Read FIFO when not full. Writes to the RBUFFER0 while the channel-0 Read FIFO is Full sets the Read FIFO Collision [RCOL0] bit. When SPI transfer data width is 16-bits, the RBUFFER0[15:0] is the entry of the valid transfer data.

If data transfer is 32 bits (DW is 3), this register can only accept word (32-bit) access, and otherwise an access error will occur.

©Levetop Electronics Co., Ltd.

If data transfer is 24 bits (DW is 2), this register can only accept word (32-bit) access, and otherwise an access error will occur.

If data transfer is 16 bits (DW is 1), this register can accept word (32-bit) or half-word (16-bit) access, byte access will cause an access error.

If data transfer is 8 bits (DW is 0), this register can accept word(32-bit), half-word(16-bit) access or byte access.

©Levetop Electronics Co., Ltd.

21.5.1.4 SCM Channel-1 Buffer Register

Register Offset Address: 0x000c

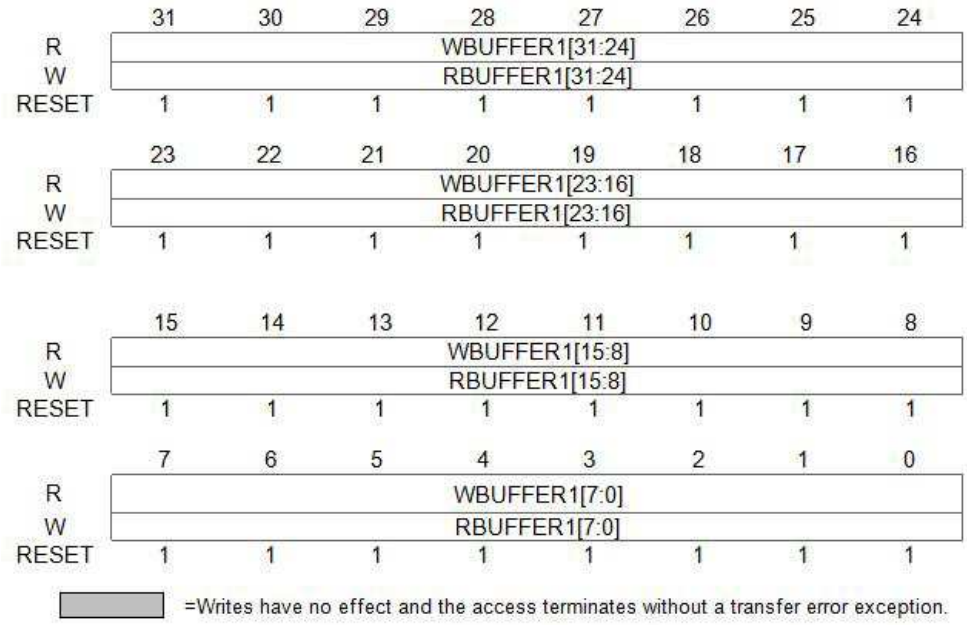


Figure 21–5: SCM Channel-1 Buffer Register (SPB1R)

WBUFFER1[31:0]

The WBUFFER1 is a 4-depth FIFO belongs to SPI channel-1. Writing to the WBUFFER1 adds the data to the channel-1 Write FIFO when not full. Writing to the WBUFFER1 while the channel-1 FIFO is full sets the Write FIFO Collision [WCOL1] bit and won't overwrite the oldest data in Write FIFO. When the SPI channel-1 Enable [SP1E] bit is cleared ('0'), the channel-1 Read Buffer will be reset. When the [SP1E] bit is set ('1'), SPISS1 is low and the write buffer is not empty, the channel-1 initiates SPI transfers if STC is 1. When the transfer is initiated, the data byte is removed from the channel-1 Write FIFO. When SPI transfer data width is 16-bits, the WBUFFER1[15:0] is the entry the valid transfer data.

If data transfer is 32 bits (DW is 3), this register can only accept word(32-bit) access, and otherwise an access error will occur.

If data transfer is 24 bits (DW is 2), this register can only accept word(32-bit) access, and otherwise an access error will occur.

If data transfer is 16 bits (DW is 1), this register can accept word(32-bit) or half-word(16-bit) access, byte access will cause an access error.

If data transfer is 8 bits DW is 0), this register can accept word(32-bit) or half-word(16-bit) access or byte access.

RBUFFER1[31:0]

The RBUFFER1 is a maximum 4-depth FIFO. When SPI channel-1 Enable [SP1E] bit is cleared ('0'), the channel-1 Read Buffer will be reset. When an SPI channel-1 transfer is finished, the received data byte is added to the channel-1 Read FIFO when not full. Writes to the RBUFFER1 while the channel-1 Read FIFO is Full sets the Read FIFO Collision [RCOL1] bit. When SPI transfer data width is 16-bits, the RBUFFER1[15:0] is the entry the valid transfer data.

If data transfer is 32 bits (DW is 3), this register can only accept word (32-bit) access, and otherwise an access error will occur.

©Levetop Electronics Co., Ltd.

If data transfer is 24 bits (DW is 2), this register can only accept word (32-bit) access, and otherwise an access error will occur.

If data transfer is 16 bits (DW is 1), this register can accept word (32-bit) or half-word (16-bit) access, byte access will cause an access error.

If data transfer is 8 bits (DW is 0), this register can accept word (32-bit) or half-word (16-bit) access or byte access.

21.5.1.5 SCM Baud Rate Register

Register address: 0x0010

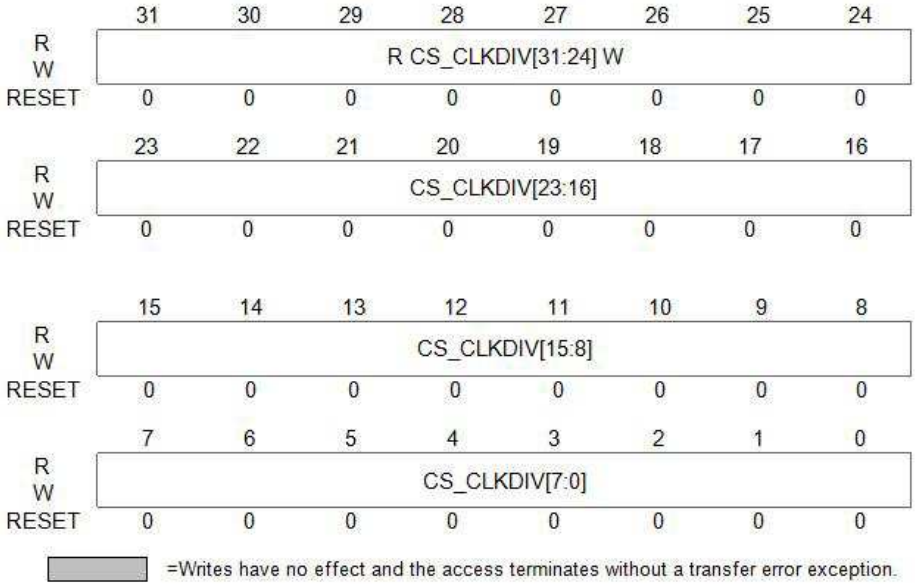


Figure 21–6: SCM Baud Rate Register (SPIBR)

CS_CLKDIV[31:0]

When SCM work in fixed clock mode (bit CS_VARCLK_EN is 0), SPICLK baud rate is defined by CS_CLKDIV[15:0]. When SCM work in variable clock mode (bit CS_VARCLK_EN is 1), SPICLK CLK1 rate is defined by CS_CLKDIV[31:16], SPICLK CLK0 rate is defined by CS_CLKDIV[15:0].

$$F_{spiclk_fixed} = F_{ipg_clk}/2(CS_CLKDIV[15:0] + 1)$$

$$F_{spi_clk0} = F_{ipg_clk}/2(CS_CLKDIV[15:0] + 1)$$

$$F_{spi_clk1} = F_{ipg_clk}/2(CS_CLKDIV[31:16] + 1)$$

©Levetop Electronics Co., Ltd.

21.5.1.6 SCM Bit Period Select Register

Register address: 0x0014

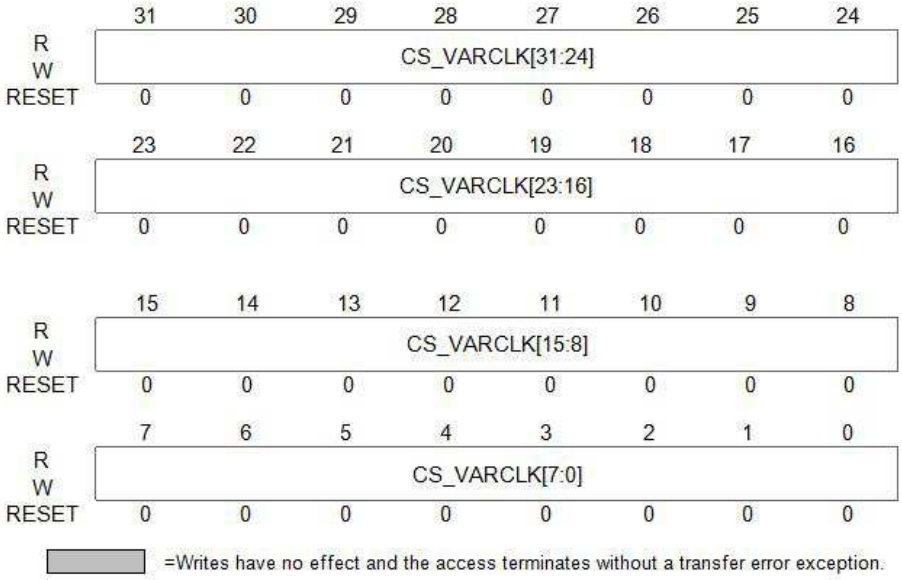


Figure 21–7: SCM Bit Period Select Register (SPIBPSR)

CS_VARCLK[31:0]

CS_VARCLK[15:0] defines each bit rate in 16-bit transfer, variable clock mode and CS_VARCLK[31:0] defines each bit rate in 32-bit transfer, variable clock mode. If CS_VARCLK[*] is 1, the corresponding bit rate is CLK1 rate, else the corresponding bit rate is CLK0 rate. CS_VARCLK is valid only in variable clock mode.

©Levetop Electronics Co., Ltd.

21.5.1.7 SCM GPIO Control and Status Register

Register address: 0x0018

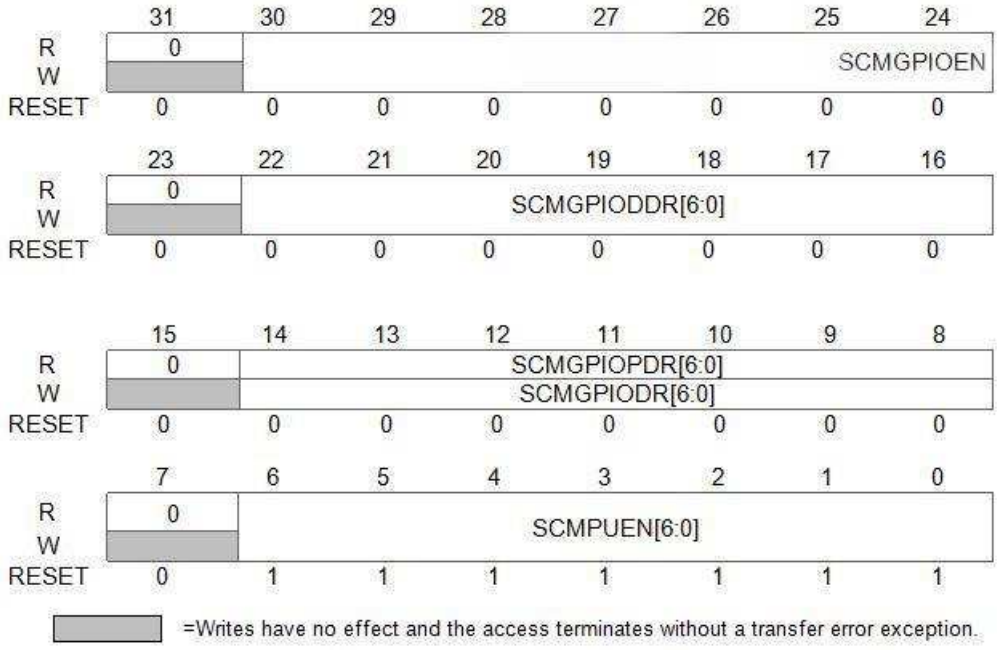


Figure 21–8: SCM GPIO Control and Status Register (SPIGPIOCSR)

SCMGPIOEN: SCM Port GPIO function control bit
1 = SCM Port GPIO function enabled
0 = SCM Port main function enabled

SCMGPIODDR[6:0]: SCM Port GPIO Direction function control bit
1 = SCM Port GPIO output enabled
0 = SCM Port GPIO input enabled

Table 21–3: SCMGPIODDR Controlled Port List

SCMGPIODDR						
6	5	4	3	2	1	0
MSS1	MSDO1	MSDI1	MSS0	MSDO0	MSDI0	MSCK

SCMGPIOPDR[6:0]
SCM GPIO port status bit, reading this address returns the status of external port.

SCMGPIODR[6:0]
SCM GPIO port data bit, writing this address will drive the write data to the external pin when output is enabled.

SCMPUEN[6:0]: SCM PORT Pull-up Control bit
1 = pull-up is enabled
0 = pull-up is disabled

©Levetop Electronics Co., Ltd.

21.5.1.8 SCM DMA Control Register

Register address: 0x001c

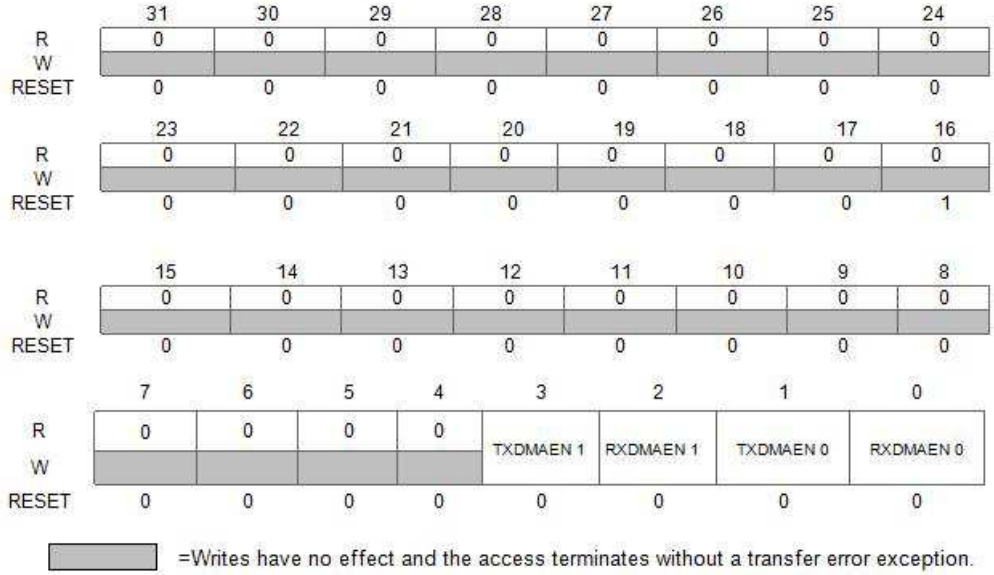


Figure 21–9: SCM DMA Control Register (SPIDMACR)

TXDMAEN0 — SPI channel0 Transmitter DMA request Control Bit.
1 = SPI channel-0 transmitter DMA request enabled.
0 = SPI channel-0 transmitter DMA request disabled.

TXDMAEN0 — SPI channel0 Receiver DMA Control Bit.
1 = SPI channel-0 receiver DMA request enabled.
0 = SPI channel-0 receiver DMA request disabled.

TXDMAEN1 — SPI channel1 Transmitter DMA request Control Bit.
1 = SPI channel-1 transmitter DMA request enabled.
0 = SPI channel-1 transmitter DMA request disabled.

TXDMAEN1 — SPI channel1 Receiver DMA Control Bit.
1 = SPI channel-1 receiver DMA request enabled.
0 = SPI channel-1 receiver DMA request disabled.

©Levetop Electronics Co., Ltd.

21.6 Operation

21.6.1 Variable Clock Mode

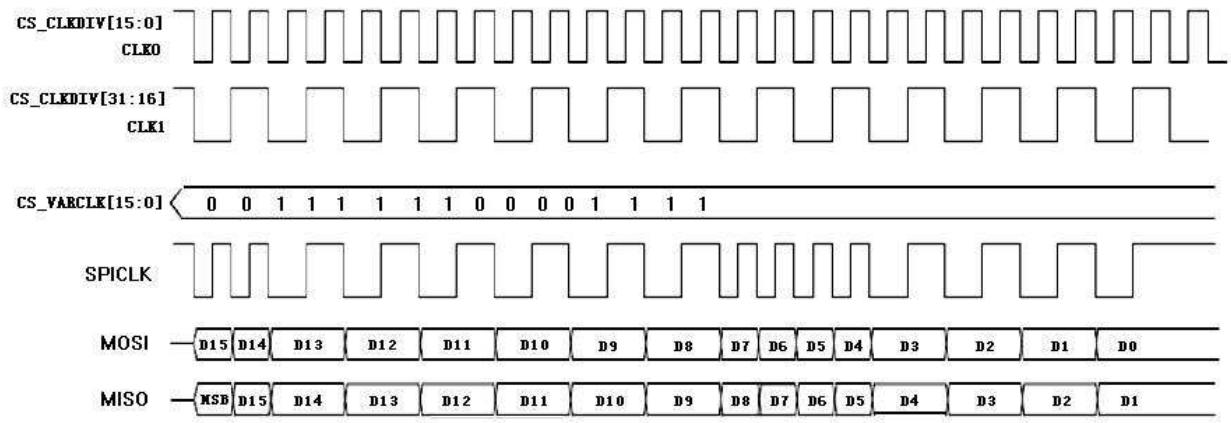


Figure 21–10: Variable Clock Mode

21.6.2 Clock Polarity

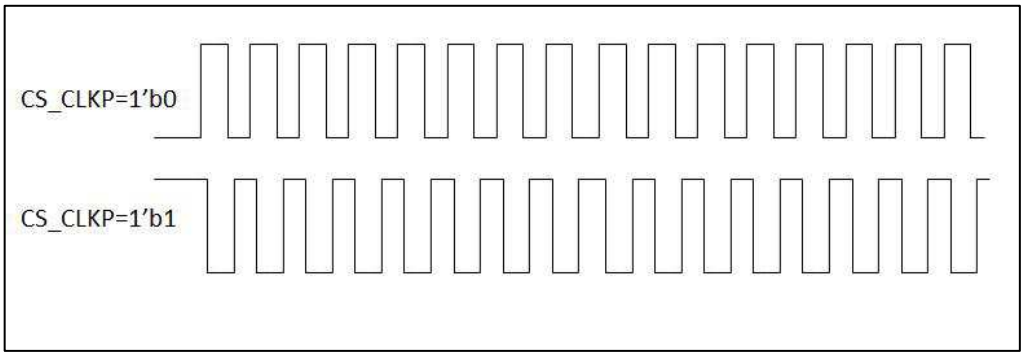


Figure 21–11: Clock Polarity

21.6.3 Serial Communication

The serial data is shift out from MSDO at the falling edge, and the MSDI from slave is sampled into the shift register at the rising edge. When MSS transition to low from high, and before the valid data is shift out, the MSDO is stay at low to avoid slave go to unknown state.

©Levetop Electronics Co., Ltd.

22. Serial Communications Interface Module (SCI)

22.1 Introduction

The serial communications interface (SCI) allows asynchronous serial communications with peripheral devices and other microcontroller units (MCU).

22.2 Features

Features of each SCI module include:

- Standard mark/space non-return-to-zero (NRZ) format
- The baud rate divisor is a 22-bit number consisting of 16-bit integer and 6-bit fractional part
- Programmable 7-bit,8-bit or 9-bit data format
- Separately enabled transmitter and receiver
- Separate receiver and transmitter central processor unit (CPU) interrupt requests
- Programmable transmitter output polarity
- Two receiver wakeup methods:
  - Idle line wakeup
  - Address mark wakeup
- Interrupt-driven operation with eight flags:
  - Transmitter empty
  - Transmission complete
  - Receiver full
  - Idle receiver input
  - Receiver overrun
  - Noise error
  - Framing error
  - Parity error
- Receiver framing error detection
- Hardware parity checking
- 1/16 bit-time noise detection
- General-purpose, I/O capability
- Serial IR interface low-speed, IrDA-compatible(up to 115.2Kbit/s)

22.3 Modes of Operation

- Serial RS-232 NRZ format
- IrDA

©Levetop Electronics Co., Ltd.

22.4 Block Diagram

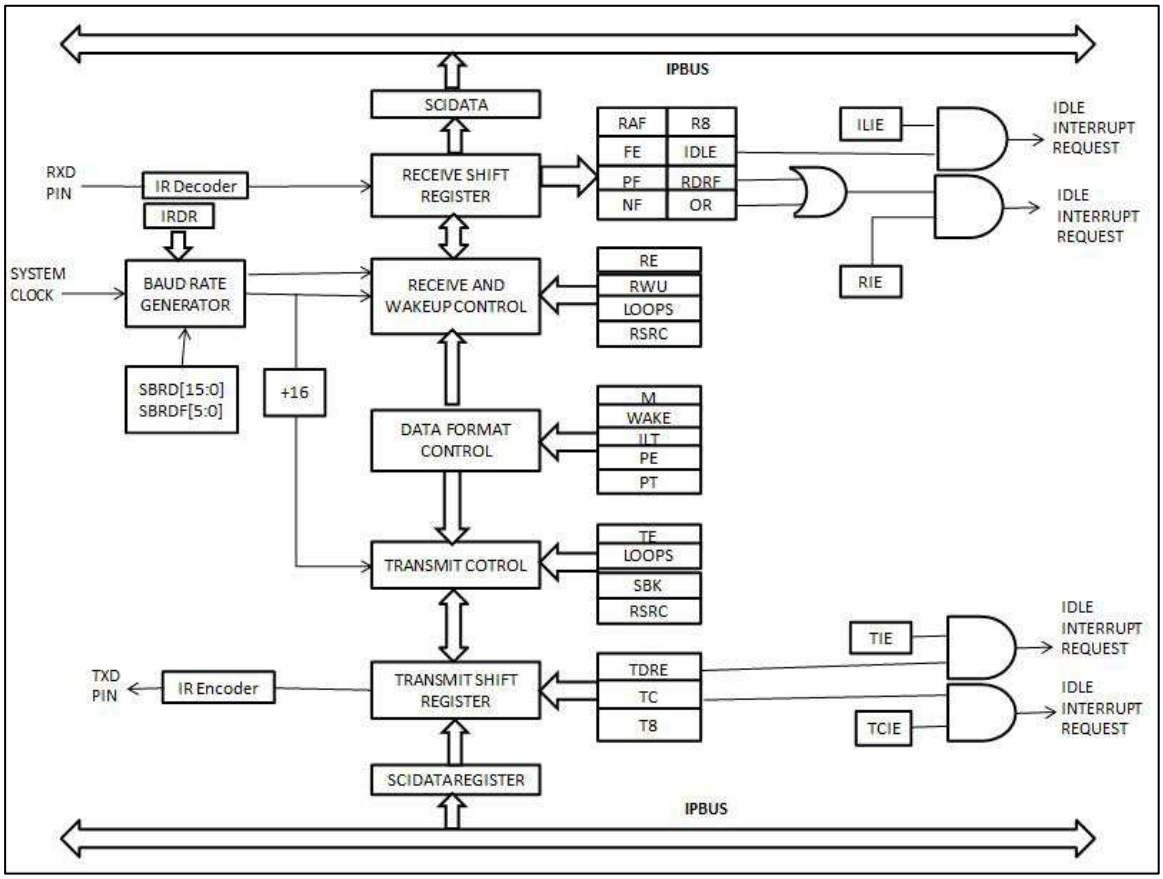


Figure 22–1: SCI Block Diagram

©Levetop Electronics Co., Ltd.

22.5 Modes of Operation

SCI operation is identical in run, special, and emulation modes. The SCI has two low-power modes, doze and stop.

Note:
Run mode is the normal mode of operation and the WAIT instruction does not affect SCI operation.

22.5.1 Doze Mode

When the SCIDOZ bit in the SCI Pullup and Reduced Drive (SCIPURD) Register is set, the DOZE instruction stops the SCI clock and puts the SCI in a low-power state. The DOZE instruction does not affect SCI register states. Any transmission or reception in progress stops at doze mode entry and resumes when an internal or external interrupt request brings the CPU out of doze mode. Exiting doze mode by reset aborts any transmission or reception in progress and resets the SCI. See 19.7.7 SCI Data Direction Register

When the SCIDOZ bit is clear, execution of the DOZE instruction has no effect on the SCI. Normal module operation continues, allowing any SCI interrupt to bring the CPU out of doze mode.

22.5.2 Stop Mode

The STOP instruction stops the SCI clock and puts the SCI in a low-power state. The STOP instruction does not affect SCI register states. Any transmission or reception in progress halts at stop mode entry and resumes when an external interrupt request brings the CPU out of stop mode. Exiting stop mode by reset aborts any transmission or reception in progress and resets the SCI.

22.6 Signal Description

Table 22–1 gives an overview of the signals which are described here.

Table 22–1: Signal Properties

Name	Function	Port	Reset State	Default Pullup State
RXD	Receive data pin	SCIPORT0	0	Disabled
TXD	Transmit data pin	SCIPORT1	0	Disabled

22.6.1 RXD

RXD is the SCI receiver pin. RXD is available for general-purpose I/O when it is not configured for receiver operation.

22.6.2 TXD

TXD is the SCI transmitter pin. TXD is available for general-purpose I/O when it is not configured for transmitter operation.

©Levetop Electronics Co., Ltd.

22.7 Memory Map and Registers (Base: 0x4009_0000)

Table 22–2 shows the SCI memory map

Note:
Reading unimplemented addresses (0x0000_000f) returns 0s. Writing to unimplemented addresses has no effect. Accessing unimplemented addresses does not generate an error response.

Table 22–2: Serial Communications Interface Module Memory Map[1]

SCI	Bits 7-0	Access[2]
0x0000	SCI Control Register 2 (SCICR2)	S/U
0x0001	SCI Control Register 1 (SCICR1)	S/U
0x0002	SCI Integer Baud-Rate Divisor Register Low (SCIBRDIL)	S/U
0x0003	SCI Integer Baud-Rate Divisor Register High (SCIBRDIH)	S/U
0x0004	SCI Data Register Low (SCIDRL)	S/U
0x0005	SCI Data Register High (SCIDRH)	S/U
0x0006	SCI Status Register 2 (SCISR2)	S/U
0x0007	SCI Status Register 1 (SCISR1)	S/U
0x0008	SCI Fractional Baud-Rate Divisor Register (SCIBRDF)	S/U
0x0009	SCI Data Direction Register (SCIDDR)	S/U
0x000a	SCI Port Data Register (SCIPORT)	S/U
0x000b	SCI Pullup and Reduced Drive Register (SCIPURD)	S/U
0x000c	Reserved[3]	S/U
0x000d	SCI InfraRed Divisor Register (SCIIRDR)	S/U
0x000e	SCI InfraRed Control Register (SCIIRCR)	S/U
0x000f	SCI Test Register (SCITR)	S/U

Notes:

1. Each module is assigned 64 Kbytes of address space, all of which may not be decoded. Accesses outside of the specified module memory map generate a bus error exception.
2. S/U = CPU supervisor or user mode access. User mode accesses to supervisor only addresses have no effect and result in a cycle termination transfer error.
3. Within the specified module memory map, accessing reserved addresses does not generate a bus error exception. Reads of reserved addresses return 0s and writes have no effect.

©Levetop Electronics Co., Ltd.

22.7.1 SCI Control Register 2

Address:
SCI: 0x0000_0000;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET	0	0	0	0	0	0	0	0

Figure 22–2: SCI Control Register 2 (SCICR2)

Read: Anytime

Write: Anytime

TIE — Transmitter Interrupt Enable Bit
This read/write bit allows the TDRE flag to generate interrupt requests. Reset clears TIE.
1 = TDRE interrupt requests enabled
0 = TDRE interrupt requests disabled

TCIE — Transmission Complete Interrupt Enable Bit
This read/write bit allows the TC flag to generate interrupt requests. Reset clears TCIE.
1 = TC interrupt requests enabled
0 = TC interrupt requests disabled

RIE — Receiver Interrupt Enable Bit
This read/write bit allows the RDRF and OR flags to generate interrupt requests. Reset clears RIE.
1 = RDRF and OR interrupt requests enabled
0 = RDRF and OR interrupt requests disabled

ILIE — Idle Line Interrupt Enable Bit
This read/write bit allows the IDLE flag to generate interrupt requests. Reset clears ILIE.
1 = IDLE interrupt requests enabled
0 = IDLE interrupt requests disabled

TE — Transmitter Enable Bit
This read/write bit enables the transmitter and configures the TXD pin as the transmitter output. Toggling TE queues an idle frame. Reset clears TE.
1 = Transmitter enabled
0 = Transmitter disabled

RE — Receiver Enable Bit
This read/write bit enables the receiver. Reset clears RE.
1 = Receiver enabled
0 = Receiver disabled

Note:
When LOOPS = 0 and TE = RE = 1, the RXD pin is an input and the TXD pin is an output regardless of the state of the DDRSC1 (TXD) and DDRSC0 (RXD) bits.

RWU — Receiver Wakeup Bit
This read/write bit puts the receiver in a standby state that inhibits receiver interrupt requests. The WAKE bit determines whether an idle input or an address mark wakes up the receiver and clears RWU. Reset clears RWU.
1 = Receiver asleep when RE = 1
0 = Receiver awake when RE = 1

©Levetop Electronics Co., Ltd.

SBK — Send Break Bit

Setting this read/write bit causes the SCI to send break frames of 10 (M = 0) or 11 (M =1) logic 0s. To send one break frame, set SBK and then clear it before the break frame is finished transmitting. As long as SBK is set, the transmitter continues to send break frames.

1 = Transmitter sends break frames.

0 = Transmitter does not send break frames.

©Levetop Electronics Co., Ltd.

22.7.2 SCI Control Register 1

Address:
SCI: 0x0000_0001;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	LOOPS	WOMS	RSRC	M	WAKE	ILT	PE	PT
RESET	0	0	0	0	0	0	0	0

Figure 22–3: SCI Control Register 1 (SCICR1)

Read: Anytime

Write: Anytime

LOOPS — Loop Select Bit

This read/write control bit switches the SCI between normal mode and loop mode. Reset clears LOOPS.

1 = Loop mode SCI operation
0 = Normal mode SCI operation

The SCI operates normally (LOOPS = 0, RSRC = X) when the output of its transmitter is connected to the TXD pin, and the input of its receiver is connected to the RXD pin.
In loop mode (LOOPS =1, RSRC = 0), the input to the SCI receiver is internally disconnected from the RXD pin logic and instead connected to the output of the SCI transmitter. The behavior of TXD is governed by the DDRSC1 bit in SCIDDR. If DDRSC1 = 1, the TXD pin is driven with the output of the SCI transmitter. If DDRSC1 = 0, the TXD pin idles high. See 19.15 Loop Operation for additional information. For either loop mode or single-wire mode to function, both the SCI receiver and transmitter must be enabled by setting the RE and TE bits in SCIxCR2.

Note:
The RXD pin becomes general-purpose I/O when LOOPS = 1, regardless of the state of the RSRC bit. DDRSC0 in SCIDDR is the data direction bit for the RXD pin.

Table 22–3 Table 22–3shows how the LOOPS, RSRC, and DDRSC0 bits affect SCI operation and the configuration of the RXD and TXD pins.

Table 22–3: SCI Normal, Loop, and Single-Wire Mode Pin Configurations

LOOPS	RSRC	SCI MODE	Receiver Input	RXD Pin Function	DDRSC0	Transmitter Output	TXD Pin Function
0	X	Normal	Tied to RXD input buffer	Receive pin	X	Tied to TXD output driver	Transmit pin
1	0	Loop	Tied to transmitter output	General-purpose I/O	0	Tied to receiver input only	None (idles high)
					1	Tied to receiver input and TXD output driver	Transmit pin
	1	Single-wire	Tied to TXD		0	No connection	Receive pin
					1	Tied to TXD output driver	Transmit pin

©Levetop Electronics Co., Ltd.

WOMS — Wired-OR Mode Select Bit

This read/write bit configures the TXD and RXD pins for open-drain operation. This allows all of the TXD pins to be tied together in a multiple-transmitter system. WOMS also affects the TXD and RXD pins when they are general-purpose outputs. External pullup resistors are necessary on open-drain outputs. Reset clears WOMS.

1 = TXD and RXD pins open-drain when outputs
0 = TXD and RXD pins CMOS drive when outputs

Note:

This bit has no effect in this part. Thus wired-or mode is not supported.

RSRC — Receiver Source Bit

This read/write bit selects the internal feedback path to the receiver input. Reset clears RSRC. when LOOPS = 1.

1 = Receiver input tied to TXD pin when LOOPS = 1
0 = Receiver input tied to transmitter output when LOOPS = 1

M — Data Format Mode Bit

This read/write bit selects 11-bit or 10-bit frames. Reset clears M.

1 = Frames have 1 start bit, 9 data bits, and 1 stop bit.
0 = Frames have 1 start bit, 8 data bits, and 1 stop bit.

WAKE — Wakeup Bit

This read/write bit selects the condition that wakes up the SCI receiver when it has been placed in a standby state by setting the RWU bit in SCICR2. When WAKE is set, a logic 1 (address mark) in the most significant bit position of a received data character wakes the receiver. An idle condition on the RXD pin does so when WAKE = 0. Reset clears WAKE.

1 = Address mark receiver wakeup
0 = Idle line receiver wakeup

ILT — Idle Line Type Bit

This read/write bit determines when the receiver starts counting logic 1s as idle character bits. The counting begins either after the start bit or after the stop bit. If the count begins after the start bit, then a string of logic 1s preceding the stop bit may cause false recognition of an idle character. Beginning the count after the stop bit avoids false idle character recognition, but requires properly synchronized transmissions. Reset clears ILT.

1 = Idle frame bit count begins after stop bit.
0 = Idle frame bit count begins after start bit.

PE — Parity Enable Bit

This read/write bit enables the parity function. When enabled, the parity function inserts a parity bit in the most significant bit position of an SCI data word. Reset clears PE.

1 = Parity function enabled
0 = Parity function disabled

PT — Parity Type Bit

This read/write bit selects even parity or odd parity. With even parity, an even number of 1s clears the parity bit and an odd number of 1s sets the parity bit. With odd parity, an odd number of 1s clears the parity bit and an even number of 1s sets the parity bit. Reset clears PT.

1 = Odd parity when PE = 1
0 = Even parity when PE = 1

©Levetop Electronics Co., Ltd.

22.7.3 SCI Baud Rate Divisor Registers

Address:
SCI: 0x0000_0003;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	SBRDI15	SBRDI14	SBRDI13	SBRDI12	SBRDI11	SBRDI10	SBRDI9	SBRDI8
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–4: SCI Integer Baud Rate Divisor Register High (SCIBRDIH)

Address:
SCI: 0x0000_0002;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	SBRDI17	SBRDI16	SBRDI15	SBRDI14	SBRDI13	SBRDI12	SBRDI11	SBRDI10
RESET	0	0	0	0	0	0	0	0

Figure 22–5: SCI Integer Baud Rate Divisor Register Low (SCIBRDIL)

Address:
SCI: 0x0000_0008;

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	SBRDI5	SBRDI4	SBRDI3	SBRDI2	SBRDI1	SBRDI0
Write								
RESET	0	0	0	0	0	0	0	0

Figure 22–6: SCI Fractional Baud Rate Divisor Register (SCIBRDF)

Read: Anytime

Write: Anytime

SBRDI[15:0]— SCI Integer Baud Rate Divisor Bits

SBRDF[5:0]— SCI Fractional Baud Rate Divisor Bits

$$BRD = BRDI + BRDF = \frac{fsys}{16*SCI\ baudrate}.$$

SBRDI=integer(BRD)=BRDI;

©Levetop Electronics Co., Ltd.

SBRDF=integer(BRDF*64+0.5);

These read/write bits control the SCI baud rate:

$$\text{SCI baudrate} = \frac{fsys}{16*SBRD}.$$

where:

1≤SBRDI≤65536
1≤SBRDF≤64

$$SBRD = SBRDI + \frac{SBRDF}{64}.$$

Note:

The baud rate generator is disabled until the TE bit or the RE bit in SCICR2 is set for the first time after reset. The baud rate generator is disabled when SBRDI[15:0] = 0 and SBRDF[5:0]=0.

Writing to SCIBRDH and SCIBRDF has no effect without also writing to SCIBRDL. Writing to SCIBRDH and SCIBRDF puts the data in a temporary location until data is written to SCIBRDL.

©Levetop Electronics Co., Ltd.

22.7.4 SCI Data Registers

Address:
SCI: 0x0000_0005;

	Bit7	6	5	4	3	2	1	Bit0
Read	R8	T8	0	0	0	0	0	0
Write								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–7: SCI Data Register High (SCIDRH)

Address:
SCI: 0x0000_0004;

	Bit7	6	5	4	3	2	1	Bit0
Read	R7	R6	R5	R4	R3	R2	R1	R0
Write	T7	T6	T5	T4	T3	T2	T1	T0
RESET	0	0	0	0	0	0	0	0

Figure 22–8: SCI Data Register Low (SCIDRL)

Read: Anytime

Write: Anytime; writing to R8 has no effect

R8 — Receive Bit 8
The R8 bit is the ninth received data bit when using the 9-bit data format (M = 1).Reset clears R8.
T8 — Transmit Bit 8
The T8 bit is the ninth transmitted data bit when using the 9-bit data format (M = 1). Reset clears T8.

R[7:0] — Receive Bits [7:0]
The R[7:0] bits are receive bits [7:0] when using the 9-bit/8-bit/7-bit data format. Reset clears R[7:0].

T[7:0] — Transmit Bits [7:0]
The T[7:0] bits are transmit bits [7:0] when using the 9-bit/8-bit/7-bit data format. Reset clears T[7:0].

Note:
If the value of T8 is the same as in the previous transmission, T8 does not have to be rewritten. The same value is transmitted until T8 is rewritten.
When using the 8-bit data format, only SCIDRL needs to be accessed. When using 8-bit write instructions to transmit 9-bit data, write first to SCIDRH, then to SCIDRL.

©Levetop Electronics Co., Ltd.

22.7.5 SCI Status Register 2

Address:
SCI: 0x0000_0006;

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	0	0	0	0	0	0
Write								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–9: SCI Status Register 2 (SCISR2)

Read: Anytime

Write: Has no meaning or effect

RAF — Receiver Active Flag
The RAF flag is set when the receiver detects a logic 0 during the RT1 time period of the start bit search. When the receiver detects an idle character, it clears RAF. Reset clears RAF.
1 = Reception in progress
0 = No reception in progress

©Levetop Electronics Co., Ltd.

22.7.6 SCI Status Register 1

Address:
SCI: 0x0000_0007;

	Bit7	6	5	4	3	2	1	Bit0
Read	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
Write								
RESET	1	1	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–10: SCI Status Register 1 (SCISR1)

Read: Anytime

Write: Has no meaning or effect

TDRE — Transmit Data Register Empty Flag

The TDRE flag is set when the transmit shift register receives a word from the SCI Data Register. It signals that the SCIDRH and SCIDRL are empty and can receive new data to transmit. If the TIE bit in the SCICR2 is also set, TDRE generates an interrupt request. Clear TDRE by reading SCISR1 and then writing to SCIDRL. Reset sets TDRE.

1 = Transmit data register empty
0 = Transmit data register not empty

RDRF — Receive Data Register Full Flag

The RDRF flag is set when the data in the receive shift register is transferred to SCIDRH and SCIDRL. It signals that the received data is available to the MCU. If the RIE bit is set in SCICR2, RDRF generates an interrupt request. Clear RDRF by reading the SCISR1 and then reading SCIDRL. Reset clears RDRF.

1 = Received data available in SCIDRH and SCIDRL
0 = Received data not available in SCIDRH and SCIDRL

IDLE — Idle Line Flag

The IDLE flag is set when 10 (if M = 0) or 11 (if M = 1) consecutive logic 1s appear on the receiver input. If the ILIE bit in SCICR2 is set, IDLE generates an interrupt request. Once IDLE is cleared, a valid frame must again set the RDRF flag before an idle condition can set the IDLE flag. Clear IDLE by reading SCISR1 and then reading SCIDRL. Reset clears IDLE.

1 = Receiver idle
0 = Receiver active or idle since reset or idle since IDLE flag last cleared

Note:

When RWU of SCICR2 =1, an idle line condition does not set the IDLE flag.

OR — Overrun Flag

The OR flag is set if data is not read from SCIDRL before the receive shift register receives the stop bit of the next frame. This is a receiver overrun condition. If the RIE bit in SCICR2 is set, OR generates an interrupt request. The data in the shift register is lost, but the data already in the SCIDRH and SCIDRL is not affected. Clear OR by reading SCISR1 and then reading SCIDRL. Reset clears OR.

1 = Overrun
0 = No overrun

NF — Noise Flag

The NF flag is set when the SCI detects noise on the receiver input. NF is set during the same cycle as the RDRF flag but does not get set in the case of an overrun. Clear NF by reading SCISR1 and then reading SCIDRL. Reset clears NF.

1 = Noise
0 = No noise

©Levetop Electronics Co., Ltd.

FE — Framing Error Flag

The FE flag is set when a logic 0 is accepted as the stop bit. FE is set during the same cycle as the RDRF flag but does not get set in the case of an overrun. FE inhibits further data reception until it is cleared. Clear FE by reading SCISR1 and then reading SCIDRL. Reset clears FE.

1 = Framing error

0 = No framing error

PF — Parity Error Flag

The PF flag is set when PE = 1 and the parity of the received data does not match its parity bit. Clear PF by reading SCISR1 and then reading SCIDRL. Reset clears PF.

1 = Parity error

0 = No parity error

22.7.7 SCI Data Direction Register

Address:
SCI: 0x0000_0009;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	RSVD7	RSVD6	RSVD5	RSVD4	RSVD3	RSVD2	DDRSC1	DDRSC0
RESET	0	0	0	0	0	0	0	0

Figure 22–11: SCI Data Direction Register (SCIDDR)

Read: Anytime

Write: Anytime

RSVD[7:2] — Reserved

Writing to these read/write bits updates their values but has no effect on functionality.

DDRSC[1:0] — SCIPORT Data Direction Bits

These bits control the data direction of the SCIPORT pins. Reset clears DDRSC[1:0].

1 = Corresponding pin configured as output

0 = Corresponding pin configured as input

Note:

When LOOPS = 0 and TE = RE = 1, the RXD pin is an input and the TXD pin is an output regardless of the state of the DDRSC1 (TXD) and DDRSC0 (RXD) bits.

©Levetop Electronics Co., Ltd.

22.7.8 SCI Port Data Register

Address:
SCI: 0x0000_000a;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	RSVD7	RSVD6	RSVD5	RSVD4	RSVD3	RSVD2	PORTSC1	PORTSC0
RESET	0	0	0	0	0	0	0	0
Pin function:							TXD	TXD

Figure 22–12: SCI Port Data Register (SCIPORT)

Read: Anytime; when DDRSCx = 0, its pin is configured as an input, and reading PORTSCx returns the pin level; when DDRSCx = 1, its pin is configured as an output, and reading PORTSCx returns the pin driver output level.

Write: Anytime; data stored in internal latch drives pin only if DDRSC bit = 1

RSVD[7:2] — Reserved
Writing to these read/write bits updates their values but has no effect on functionality.

PORTSC[1:0] — SCIPORT Data Bits
These are the read/write data bits of the SCI port.

Note:
Writes to SCIPORT do not change the pin state when the pin is configured for SCI input.

To ensure correct reading of the SCI pin values from SCIPORT, always wait at least one cycle after writing to SCIDDR before reading SCIPORT.

©Levetop Electronics Co., Ltd.

22.7.9 SCI Pullup and Reduced Drive Register

Address:
SCI: 0x0000_000b;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	SCISDOZ	0	RSVD5	RSVD4	0	0	RSVD1	PUPSCI
RESET	1	1	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–13: SCI Pullup and Reduced Drive Register (SCIPURD)

Write: Anytime

SCISDOZ — SCI Stop in Doze Mode Bit
The SCISDOZ bit disables the SCI in doze mode.
1 = SCI disabled in doze mode
0 = SCI enabled in doze mode

RSVD[5,4,1] — Reserved
Writing to these read/write bits updates their values but has no effect on functionality.

PUPSCI — Pullup Enable Bit
This read/write bit enables the pullups on pins TXD and RXD. If a pin is programmed as an output, the pullup is disabled.
1 = TXD and RXD pullups enabled
0 = TXD and RXD pullups disabled

Note:
The PUPSCI bit has no effect in this part. Pullups are always enabled.

©Levetop Electronics Co., Ltd.

22.7.10 SCI InfraRed Divisor Register

Address:
SCI: 0x0000_000d;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	IRDR7	IRDR6	IRDR5	IRDR4	IRDR3	IRDR2	IRDR1	IRDR0
RESET	1	1	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 22–14: SCI InfraRed Divisor Register (SCIIRDR)

Read: Anytime

Write: Anytime

IRDR[7:0]— SCI InfraRed Divisor Bits

$$Irclk = \frac{fsys}{IRDR}$$

Note:
When the IrDA transceiver emit a light pulse which is shorter than the SCI baudrate sampling clock, the user must insure the frequency of IR sampling clock is high enough to measure the pulse. The user should set IRSC to 1, and set IRDR to get high frequency IR sampling clock which is divided by system clock.

Let's take 2 examples, with the Minimum Pulse Duration equals to the MPD of the IrDA specification (in SIR).

Example 1: Calculation of Baudrate Sampling Clock Period (BS_clock Period < 1.41us)

The user wants to receive IrDA data at 115.2 Kbit/s. The SCIBRDI and SCIBRDF registers are set in order to create the baudrate sampling clock (BS_clock) with a frequency of 16*baud rate = 16 * 115.2 = 1.843MHz. But at the same time, in order to correctly detect the pulse, the user must be sure that N* BS_clock period is lower than 1.41 us. (N is decided by RNUM)

Let's check:
BS_clock period = 1/1843000 = 542 ns
So 2*BS_clock period = 1.09 us < 1.41 us. It is fine. N can be 1 or 2.RNUM can be set 2'b00 or 2'b01.

Example 2: Calculation of Baudrate Sampling Clock Period (BS_clock Period > 1.41 us)

This time the user wants to receive at 19.2 Kbit/s. So, the BS_clock is set to 16*19200 = 307.2 kHz

Let's check if N* BS_clock period < 1.41 us:

BS_clock period =1/307200 = 3.25 us,

So N*BS_clock period>>1.41us. It doesn't work. In this case, the BS_clock can't be used to measure the pulse duration and the user must select the IR sampling clock by setting IRSC =1.

©Levetop Electronics Co., Ltd.

When system clock frequency is 32MHz:

SYS_clock period = 1/32000000=0.03125us = IR_clock period / IRDR;

So when N*IR_clock period < 1.41us (N<=4), only when IRDR < 1.41/(0.03125*N)

When N=1,IRDR set to 8'h2d; IR_clock period=1.406us<1.41us;(N can only be 1)
When N=2,IRDR set to 8'h16; 2*IR_clock period=1.375us<1.41us;(N can only be 1 or 2)
When N=3,IRDR set to 8'h0f; 3*IR_clock period=1.406us<1.41us;(N can only be 1 or 2 or 3)
When N=4,IRDR set to 8'h0b; 4*IR_clock period=1.375us<1.41us;(N can only be 1 or 2 or 3 or 4)

Usually we set RNUM=2'b01 (N=2) to sampling pulse.

22.7.11 SCI InfraRed Control Register

Address:
SCI: 0x0000_000e;

	Bit7	6	5	4	3	2	1	Bit0
Read Write	TNUM		RNUM		TINV	RINV	IRSC	IREN
RESET	1	0	0	1	0	0	0	0

Figure 22–15: SCI Infrared Control Register (SCIIRCR)

Read: Anytime

Write: Anytime

IREN — InfraRed Interface Enable Bit
This read/write control bit enable/disable the IR interface. Reset clears IREN.
1 = IR interface enable
0 = IR interface disable

IRSC — InfraRed Interface Sampling Clock Select Bit
This read/write control bit select the sampling clock of the IR interface. Reset clears IRSC.
1 = IR interface use the divided clock of system clk(according to SCIIRDR)
0 = IR interface use the baudrate sampling clock (16xbaudrate)

Note:
The value to apply to IRSC bit is based on 2 parameters: the baud rate and the Minimum Pulse Duration (MPD) of the transceiver. According to IrDA Standard Specification, for SIR (Serial IR) baud rates from 2.4 Kbit/s to 115.2 Kbit/s this nominal pulse duration is equal to 3/16 of a bit duration (at the selected baud rate). But, for all the baud rates a Minimum Pulse Duration is also specified. According to IrDA Standard, a Zero is represented by a light pulse, so the IrDA transceiver can't emit a light pulse shorter than the MPD. For SIR, the MPD is constant and equal to 1.41 µs. This means at any time, the user must insure the frequency of IR sampling clock is high enough to measure the pulse

In normal operation, IRSC=0, the pulse must last at least 1 baudrate sampling clock cycle(IR sample clock cycle num decided by RNUM).If this condition is not fulfilled, IRSC must be set to 1.

©Levetop Electronics Co., Ltd.

RINV — Inverted Infrared Reception Bit

This read/write control bit determines the logic level for the detection. Reset clears RINV.

1 = Active high detection. The infrared logic block expects an active high or positive IR 3/16 pulse for 0's and active low are expected for 1's.

0 = Active low detection. The infrared logic block expects an active low or negative IR 3/16 pulse for 0's and active high are expected for 1's.

TINV — Inverted Infrared Transmission Bit

This read/write control bit sets the active level for the transmission. Reset clears TINV.

1 = Active low transmission. The infrared logic block transmits an active low or negative IR 3/16 pulse for all 0's and active high are transmitted for 1's.

0 = Active high transmission. The infrared logic block transmits an active high or positive IR 3/16 pulse for all 0's and active low are transmitted for 1's.

RNUM[1:0] — Reception Number Bit

These read/write control bits determines the reception sample numbers for vote logic. Reset value is 2'b01.

Table 22–4: RNUM Description

RNUM[1:0]	Sample NUM
00	1 times
01	2 times
10	3 times
11	4 times

Note:

The IR sampling clock is determined by IRSC.

When set RNUM[1:0]=11,users should set IRSC=1,because according to IrDA spec, 1.41us<MPD

TNUM[1:0] — Transmission Number Bit

This read/write control bit determines the transmission clock cycle numbers for generate pulse. Reset value is 2'b10.

Table 22–5: TNUM Description

TNUM[1:0]	Trans Time
00	1/16 Baudrate
01	2/16 Baudrate
10	3/16 Baudrate
11	Reserved

22.7.12 SCI Test Register

Address:
SCI: 0x0000_000f;

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	0	0	0	0	0	0
Write								
RESET	0	0	0	0	0	0	0	0

Figure 22–16: SCI Test Register (SCITR)

©Levetop Electronics Co., Ltd.

22.8 Functional Description

The SCI allows full-duplex, asynchronous, non-return-to-zero (NRZ) serial communication between the MCU and remote devices, including other MCUs. The SCI transmitter and receiver operate independently, although they use the same baud rate generator. The CPU monitors the status of the SCI, writes the data to be transmitted, and processes received data.

The SCI module also includes a serial IR (SIR) encoder/decoder block that can be connected to an infrared transceiver to implement an IrDA SIR physical layer.

22.9 Data Format

The SCI uses the standard NRZ mark/space data format shown in **Figure 22–17**

Each frame has a start bit, seven/eight/nine data bits, and one or two stop bits. Clearing the M bit in SCCR1 configures the SCI for 10-bit frames. Setting the M bit configures the SCI for 11-bit frames.

When the SCI is configured for 9-bit data, the ninth data bit is the T8 bit in SCI Data Register high (SCIDRH). It remains unchanged after transmission and can be used repeatedly without rewriting it. A frame with nine data bits has a total of 11 bits.

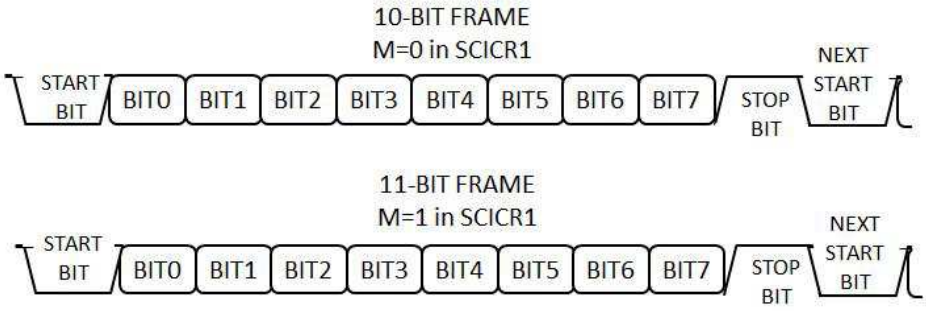


Figure 22–17: SCI Data Formats

©Levetop Electronics Co., Ltd.

22.10 Serial IR (SIR)

The SCI includes an IrDA serial-IR (SIR) encoder/decoder block. The IrDA SIR block provides functionality that converts between an asynchronous data stream and a half-duplex serial SIR interface. No analog processing is performed on-chip. The role of the SIR block is to provide a digital encoded output and decoded input to the SCI. When enabled (IREN=1), the SIR block uses the Tx and Rx pins for the SIR protocol. These signals should be connected to an infrared transceiver to implement an IrDA SIR physical layer link. The SIR block can receive and transmit, but it is only half-duplex so it cannot do both at the same time. Transmission must be stopped before data can be received. (The IrDA SIR physical layer specifies a minimum 10-ms delay between transmission and reception.)The Infrared Interface is compatible with IrDA Serial Infrared Physical Layer Specification. In this specification, a “zero” is represented by a positive pulse, and a “one” is represented by no pulse (line remains low).

In the SCI:

In TX: For each “zero” to be transmitted, a narrow positive pulse which is 1/16~3/16 of a bit time is generated (due to TNUM).For each “one” to be transmitted no pulse is generated (output is low). External circuitry has to be provided to drive an Infrared LED.

In RX: When receiving, a narrow negative pulse is expected for each “zero” received while no pulse is expected for each “one” received (input is high).

The values of TINV and RINV depend of the IrDA transceiver connected on the TXD and RXD pins of the SCI. If this transceiver is not inverting on both paths Tx and Rx, a Zero is represented by a positive pulse and a One is represented by no pulse (line remains low).In this case, the bit TINV must be set to 0 and the bit RINV must be set to 1 (because Rx IR block expects an inverted signal). On the contrary user must set TINV=1 and RINV=0 if both paths of the transceiver are inverting, that is, a Zero is represented as a negative pulse and a One is represented by no pulse (line remains high). The transceiver can also be inverting on only one path (Tx or Rx), in this case TINV and RINV must be together equal to 1 or to 0, depending on which path is inverted.

In **Figure 22–18**, TINV=0, RINV=0.

While if RINV=1,When receiving, a narrow positive pulse is expected for each “zero” transmitted while no pulse is expected for each “one” transmitted (input is low).

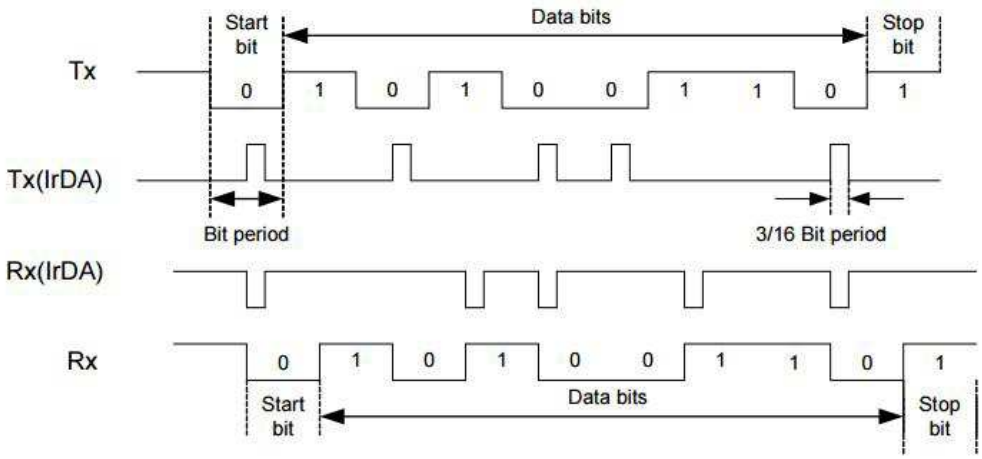


Figure 22–18: IrDA Data Modulation

©Levetop Electronics Co., Ltd.

22.11 Baud Rate Generation

The baud rate divisor is a 22-bit number consisting of 16-bit integer and 6-bit fractional part. The number formed by these two values is used by the baud-rate generator to derive the baud rate for both the receiver and the transmitter. The value written to SCIBRDIH, SCIBRDIL and SCIBRDF determines the system clock divisor. The baud rate clock is synchronized with the bus clock and drives the receiver. The baud rate clock divided by 16 drives the transmitter. The receiver acquisition rate is 16 samples per bit time.

Baud rate generation is subject to two sources of error:

1. Integer division and fractional division of the module clock may not give the exact target frequency
2. Synchronization with the bus clock can cause phase shift.

Table 22–6: Example Baud Rates (System Clock = 31 MHz)

SBRDI[15:0]	SBRDF[5:0]	Receiver Clock(Hz）	Transmitter Clock(Hz)	Target Baud Rate	Percent Error
0x44cd	0x29	1,760.00	110	110	0.00002%
0x193a	0x15	4,800.00	300.0002	300	0.00008%
0x0c9d	0x0b	9,599.98	599.999	600	0.00016%
0x064e	0x25	19,200.06	1,200.004	1200	0.00032%
0x0327	0x13	38,399.75	2,399.985	2400	0.00065%
0x0193	0x29	76,800.99	4,800.062	4800	0.00129%
0x00c9	0x35	153,596.04	9,599.752	9,600	0.00258%
0x0086	0x23	230,402.97	14,400.19	14,400	0.00129%
0x0064	0x3a	307,215.86	19,200.99	19,200	0.00516%
0x0032	0x1d	614,431.71	38,401.98	38,400	0.00516%
0x0022	0x26	896,115.63	56,007.23	56,000	0.01290%
0x0021	0x29	921,504.88	57,594.05	57,600	0.01032%
0x0010	0x34	1,843,866.17	115,241.6	115,200	0.03614%
0x000f	0x09	2,047,471.62	127,967	128,000	0.02580%
0x0008	0x1a	3,687,732.34	230,483.3	230,400	0.03614%
0x0004	0x000d	7,375,464.68	460,966.5	460,800	0.03614%
0x0001	0x3c	16,000,000.00	1,000,000	1,000,000	0.00000%
0x0000	0x3e	32,000,000.00	2,000,000	2,000,000	0.00000%

©Levetop Electronics Co., Ltd.

22.12 Transmitter

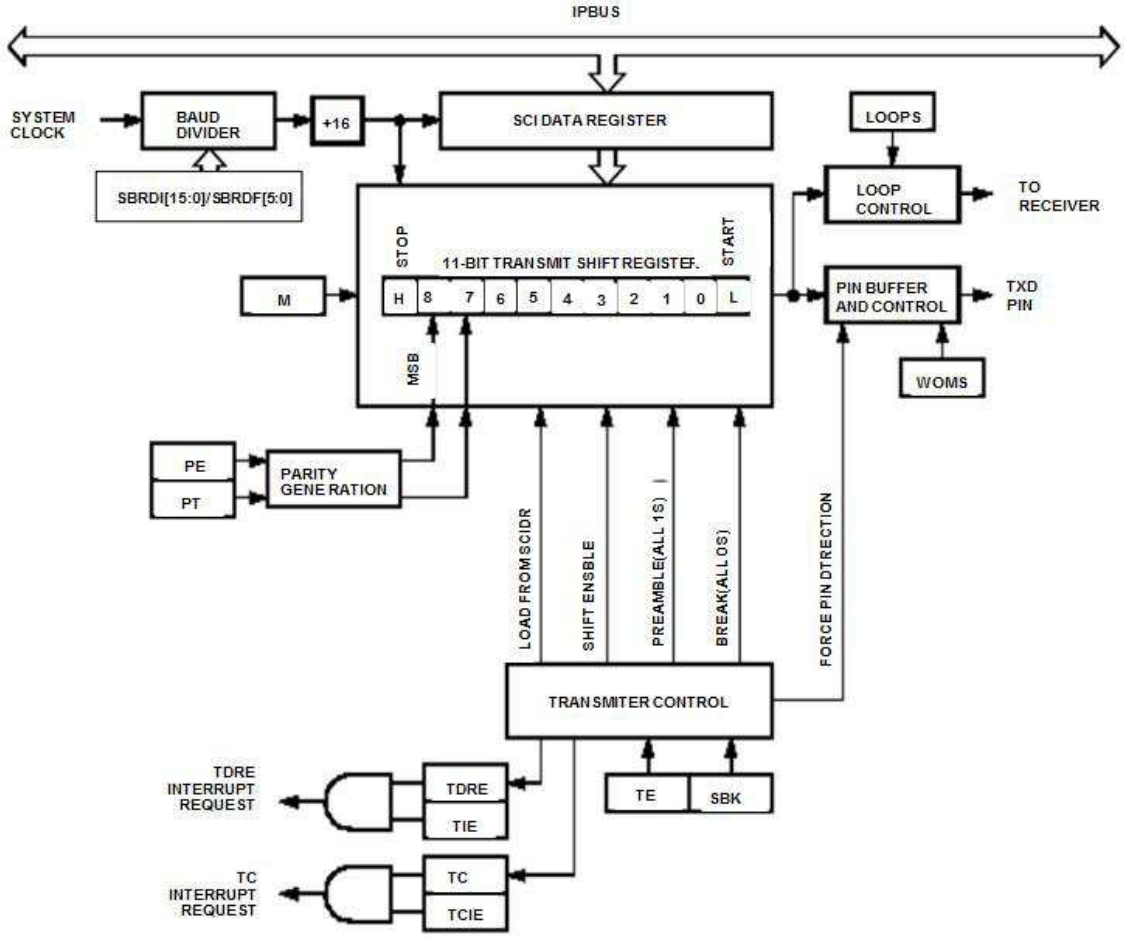


Figure 22–19: Transmitter Block Diagram

©Levetop Electronics Co., Ltd.

22.12.1 Frame Length

The transmitter can generate either 10-bit or 11-bit frames. In SCICR1, the M bit selects frame length, and the PE bit enables the parity function. One data bit may be an address mark or an extra stop bit. All frames begin with a start bit and end with one or two stop bits. When transmitting 9-bit data, bit T8 in SCI Data Register high (SCIDRH) is the ninth bit (bit 8).

Table 22–7: Example 10-Bit and 11-Bit Frames

M Bit	Frame Length	Start Bit	Data Bits	Parity Bit	Address Mark[1]	Stop Bit(s)
0	10 bits	1	8	No	No	1
		1	7	No	No	2
		1	7	No	Yes	1
		1	7	Yes	No	1
1	11 bits	1	9	No	No	1
		1	8	No	No	2
		1	8	No	Yes	1
		1	8	Yes	No	1
		1	7	No	Yes	2
		1	7	Yes	No	2

Note:

1. When implementing a multidrop network using the SCI, the address mark bit is used to designate subsequent data frames as a network address and not device data.

©Levetop Electronics Co., Ltd.

22.12.2 Transmitting A Frame

To begin an SCI transmission:

1. Configure the SCI:
   - a Write a baud rate value to SCIBDH and SCIBDL.
   - b Write to SCICR1 to:
     - i. Enable or disable loop mode and select the receiver feedback path
     - ii. Select open-drain or wired-OR SCI outputs
     - iii. Select 10-bit or 11-bit frames
     - iv. Select the receiver wakeup condition: address mark or idle line
     - v. Select idle line type
     - vi. Enable or disable the parity function and select odd or even parity
   - c Write to SCICR2 to:
     - i. Enable or disable TDRE, TC, RDRF, and IDLE interrupt requests
     - ii. Enable the transmitter and queue a break frame
     - iii. Enable or disable the receiver
     - iv. Put the receiver in standby if required
2. Transmit a byte:
   - a Clear the TDRE flag by reading SCISR1 and, if sending 9-bit data, write the ninth data bit to SCDRH.
   - b Write the byte to be transmitted (or low-order 8 bits if sending 9-bit data) to SCIDRL.
3. Repeat step 2 for each subsequent transmission.

Writing the TE bit from 0 to 1 loads the transmit shift register with a preamble of 10 (if M = 0) or 11 (if M = 1) logic 1s. When the preamble shifts out, the SCI transfers the data from SCIDRH and SCIDRL to the transmit shift register. The transmit shift register prefaces the data with a 0 start bit and appends the data with a 1 stop bit and begins shifting out the frame.

The SCI sets the TDRE flag every time it transfers data from SCIDRH and SCIDRL to the transmit shift register. TDRE indicates that SCIDRH and SCIDRL can accept new data. If the TIE bit is set, TDRE generates an interrupt request.

Note:
SCIDRH and SCIDRL transfer data to the transmit shift register and sets TDRE 9/16ths of a bit time after the previous frame's stop bit starts to shift out.

Hardware supports odd or even parity. When parity is enabled, the most significant data bit is the parity bit.

When the transmit shift register is not transmitting a frame, the TXD pin goes to the idle condition, logic 1. Clearing the TE bit while the transmitter is idle will return control of the TXD pin to the SCI data direction (SCIDDR) and SCI port (SCIPORT) registers.

If the TE bit is cleared while a transmission is in progress (while TC = 0), the frame in the transmit shift register continues to shift out. Then the TXD pin reverts to being a general-purpose I/O pin even if there is data pending in the SCI Data Register. To avoid accidentally cutting off a message, always wait until TDRE is set after the last frame before clearing TE.

©Levetop Electronics Co., Ltd.

To separate messages with preambles with minimum idle line time, use this sequence between messages:

1. Write the last byte of the first message to SCIDRH and SCIDRL.
2. Wait until the TDRE flag is set, indicating the transfer of the last frame to the transmit shift register.
3. Queue a preamble by clearing and then setting the TE bit.
4. Write the first byte of the second message to SCIDRH and SCIDRL.

When the SCI relinquishes the TXD pin, the SCIPORT and SCIDDR registers control the TXD pin.

To force TXD high when turning off the transmitter, set bit 1 of the SCI Port Register (SCIPORT) and bit 1 of the SCI Data Direction Register (SCIDDR). The TXD pin goes high as soon as the SCI relinquishes control of it. See **22.7.7 SCI Data Direction Register**.

22.12.3 Break Frames

Setting the SBK bit in SCICR2 loads the transmit shift register with a break frame. A break frame contains all logic 0s and has no start, stop, or parity bit. Break frame length depends on the M bit in the SCICR1 register. As long as SBK is set, the SCI continuously loads break frames into the transmit shift register. After SBK is clear, the transmit shift register finishes transmitting the last break frame and then transmits at least one logic 1. The automatic logic 1 at the end of a break frame guarantees the recognition of the next start bit.

The SCI recognizes a break frame when a start bit is followed by eight or nine 0 data bits and a 0 where the stop bit should be. Receiving a break frame has these effects on SCI registers:

- Sets the FE flag • Sets the RDRF flag
- Clears the SCIDRH and SCIDRL
- May set the OR flag, NF flag, PE flag, or the RAF flag

22.12.4 Idle Frames

An idle frame contains all logic 1s and has no start, stop, or parity bit. Idle frame length depends on the M bit in the SCICR1 register. The preamble is a synchronizing idle frame that begins the first transmission after writing the TE bit from 0 to 1.

If the TE bit is cleared during a transmission, the TXD pin becomes idle after completion of the transmission in progress. Clearing and then setting the TE bit during a transmission queues an idle frame to be sent after the frame currently being transmitted.

Note:

When queuing an idle frame, return the TE bit to logic 1 before the stop bit of the current frame shifts out to the TXD pin. Setting TE after the stop bit appears on TXD causes data previously written to SCIDRH and SCIDRL to be lost. Toggle TE to queue an idle frame, while the TDRE flag is set, immediately before writing new data to SCIDRH and SCIDRL.

©Levetop Electronics Co., Ltd.

22.13 Receiver

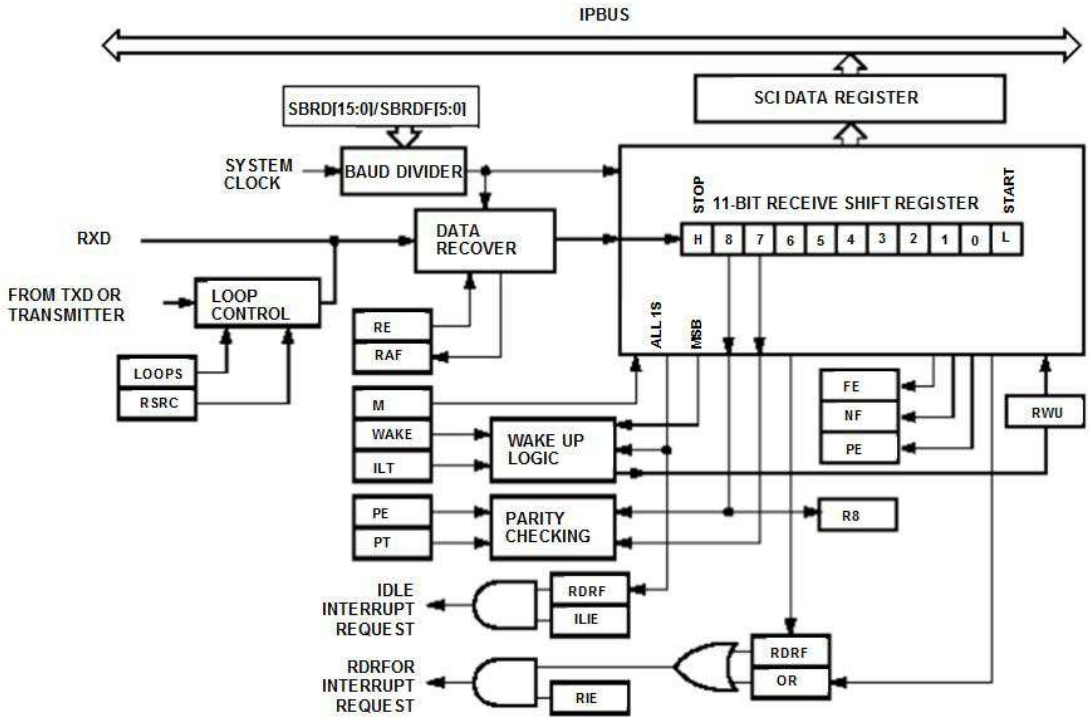


Figure 22–20: SCI Receiver Block Diagram

22.13.1 Frame Length

The receiver can handle either 7/8-bit or 9-bit data. The state of the M bit in SCICR1 selects frame length. When receiving 9-bit data, bit R8 in SCIDRH is the ninth bit (bit 8).

22.13.2 Receiving A Frame

When the SCI receives a frame, the receive shift register shifts the frame in from the RXD pin.

After an entire frame shifts into the receive shift register, the data portion of the frame transfers to SCIDRH and SCIDRL. The RDRF flag is set, indicating that the RX data buffer can be read. If the RIE bit is also set, RDRF generates an interrupt request.

22.13.3 Data Sampling

The receiver samples the RXD pin at the RT clock rate. The RT clock is an internal signal with a frequency 16 times the baud rate. To adjust for baud rate mismatch, the RT clock resynchronizes:

- After every start bit
- After the receiver detects a data bit change from logic 1 to logic 0 (after the majority of data bit samples at RT8, RT9, and RT10 returns a valid logic 1 and the majority of the next RT8, RT9, and RT10 samples returns a valid logic 0)

To locate the start bit, data recovery logic does an asynchronous search for a 0 preceded by three 1s. When the falling edge of a possible start bit occurs, the RT clock begins to count to 16.

©Levetop Electronics Co., Ltd.

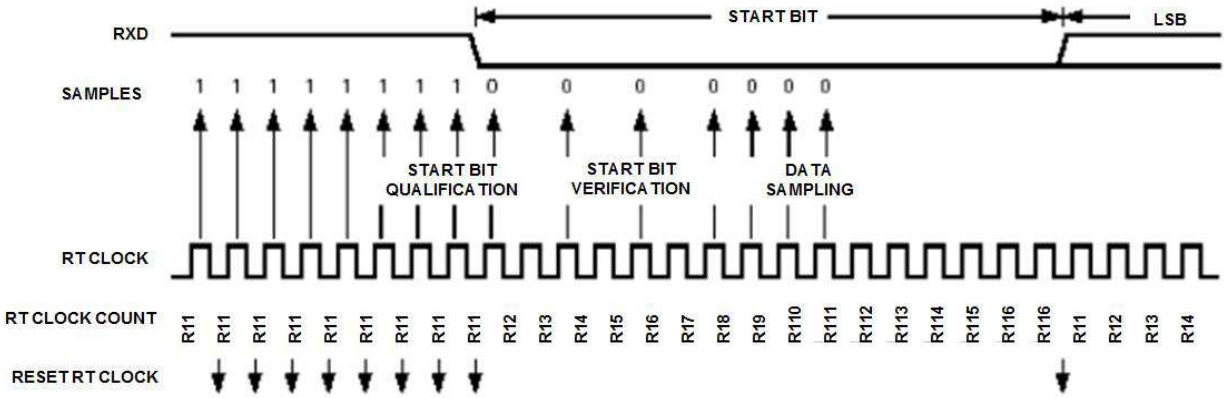


Figure 22–21: Receiver Data Sampling

To verify the start bit and to detect noise, data recovery logic takes samples at RT3, RT5, and RT7.

Table 22–8: Start Bit Verification

RT3, RT5, and RT7 Samples	Start Bit Verification	Noise Flag
000	Yes	0
001	Yes	1
010	Yes	1
011	No	0
100	Yes	1
101	No	0
110	No	0
111	No	0

If start bit verification is not successful, the RT clock is reset and a new search for a start bit begins.

To determine the value of a data bit and to detect noise, recovery logic takes samples at RT8, RT9, and RT10.

Table 22–9: Data Bit Recovery

RT8, RT9, and RT10 Samples	Data Bit Determination	Noise Flag
000	0	0
001	0	1
010	0	1
011	1	1
100	0	1
101	1	1
110	1	1
111	1	0

©Levetop Electronics Co., Ltd.

Note:

The RT8, RT9, and RT10 data samples do not affect start bit verification. If any or all of the RT8, RT9, and RT10 samples are logic 1s following a successful start bit verification, the NF flag is set and the receiver interprets the bit as a start bit (logic 0).

The RT8, RT9, and RT10 samples also verify stop bits.

Table 22–10: Stop Bit Recovery

RT8, RT9, and RT10 Samples	Framing Error Flag	Noise Flag
000	1	0
001	1	1
010	1	1
011	0	1
100	1	1
101	0	1
110	0	1
111	0	0

In **Figure 22–22** the verification samples RT3 and RT5 determine that the first low detected was noise and not the beginning of a start bit. The RT clock is reset and the start bit search begins again. The NF flag is not set because the noise occurred before the start bit was verified.

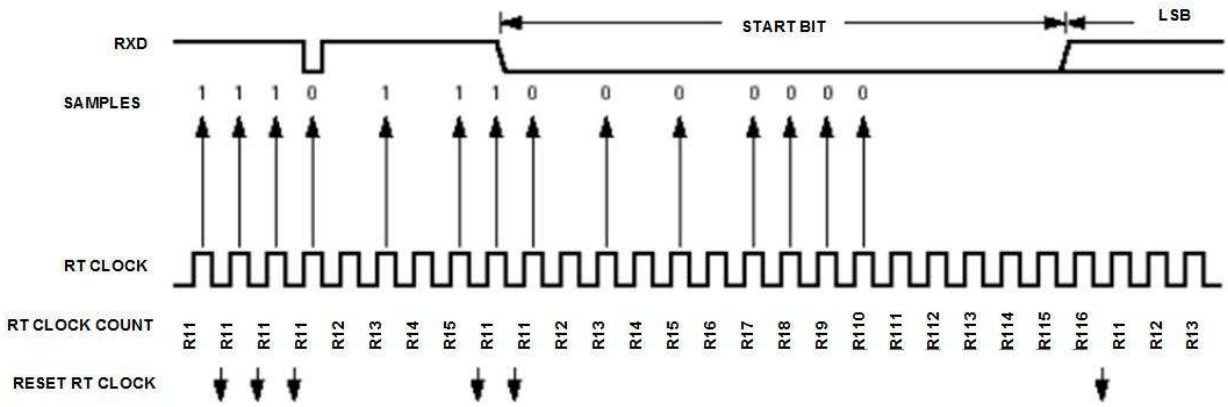


Figure 22–22: Start Bit Search Example 1

©Levetop Electronics Co., Ltd.

In **Figure 22–23** noise is perceived as the beginning of a start bit although the RT3 sample is high. The RT3 sample sets the noise flag. Although the perceived bit time is misaligned, the RT8, RT9, and RT10 data samples are within the bit time, and data recovery is successful.

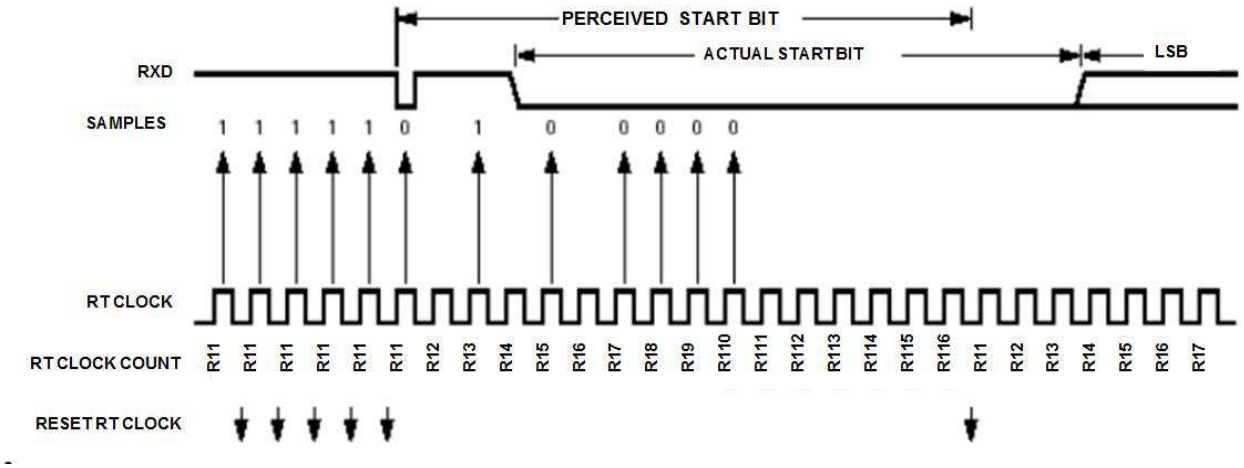


Figure 22–23: Start Bit Search Example 2

In **Figure 22–24** a large burst of noise is perceived as the beginning of a start bit, although the RT5 sample is high. The RT5 sample sets the noise flag. Although this is a worst-case misalignment of perceived bit time, the data samples RT8, RT9, and RT10 are within the bit time and data recovery is successful.

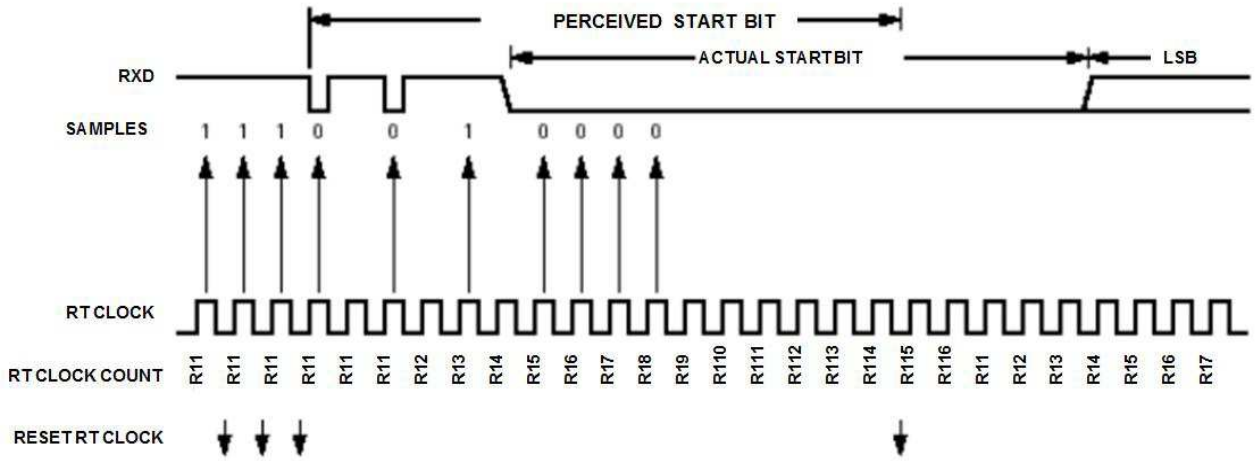


Figure 22–24: Start Bit Search Example 3

©Levetop Electronics Co., Ltd.

Figure 22–25 shows the effect of noise early in the start bit time. Although this noise does not affect proper synchronization with the start bit time, it does set the noise flag.

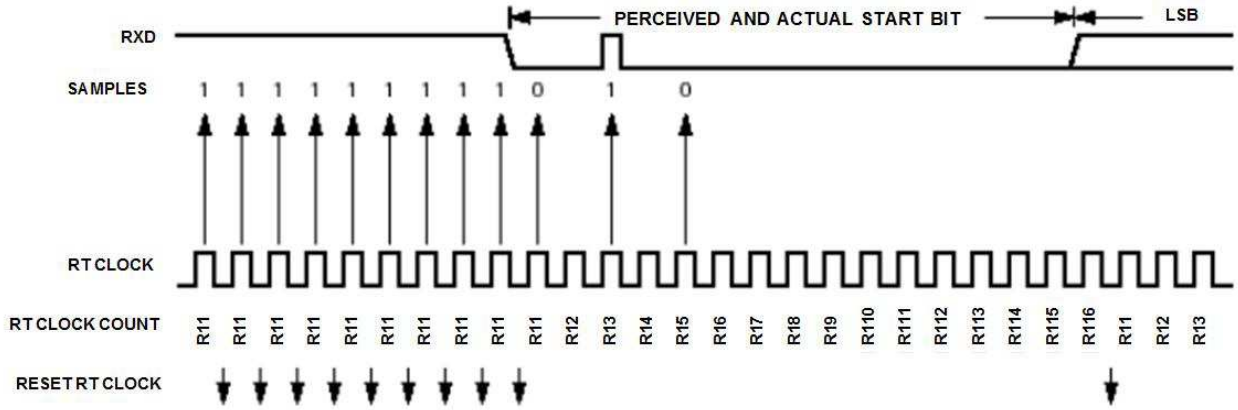


Figure 22–25: Start Bit Search Example 4

Figure 22–26 shows a burst of noise near the beginning of the start bit that resets the RT clock. The sample after the reset is low but is not preceded by three high samples that would qualify as a falling edge. Depending on the timing of the start bit search and on the data, the frame may be missed entirely or it may set the framing error flag.

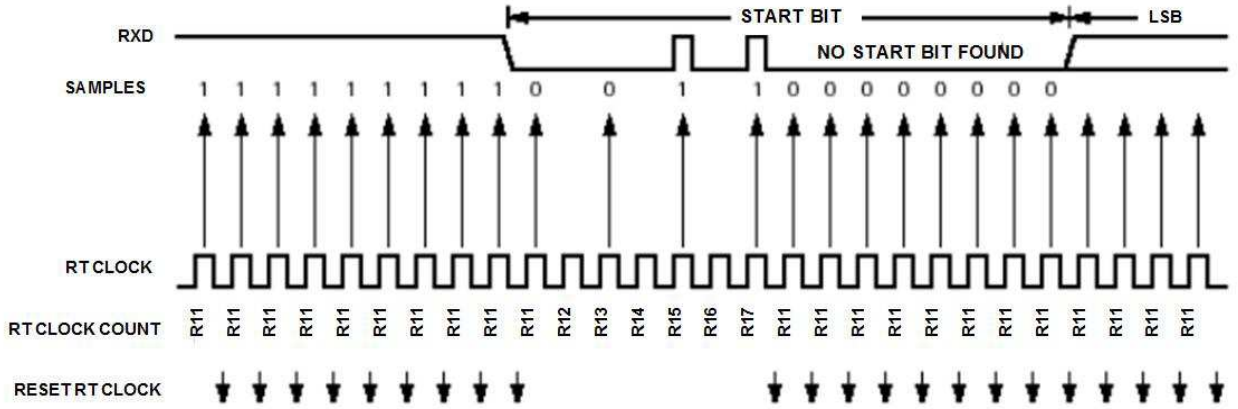


Figure 22–26: Start Bit Search Example 5

©Levetop Electronics Co., Ltd.

In **Figure 22–27** a noise burst makes the majority of data samples RT8, RT9, and RT10 high. This sets the noise flag but does not reset the RT clock. In start bits only, the RT8, RT9, and RT10 data samples are ignored.

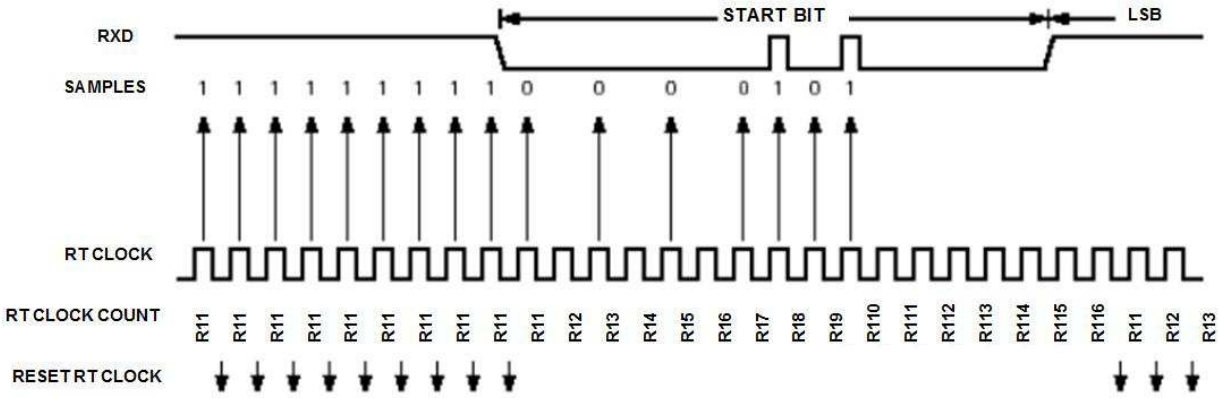


Figure 22–27: Start Bit Search Example 6

22.13.4 Framing Errors

If the data recovery logic does not detect a 1 where the stop bit should be in an incoming frame, it sets the FE flag in SCISR1. A break frame also sets the FE flag because a break frame has no stop bit. The FE flag is set at the same time that the RDRF flag is set.

©Levetop Electronics Co., Ltd.

22.13.5 Baud Rate Tolerance

A transmitting device may be operating at a baud rate below or above the receiver baud rate. Accumulated bit time misalignment can cause one of the RT8, RT9, and RT10 stop bit data samples to fall outside the stop bit. A noise error occurs if the samples are not all the same value. If more than one of the samples is outside the stop bit, a framing error occurs. In most applications, the baud rate tolerance is much more than the degree of misalignment that is likely to occur.

As the receiver samples an incoming frame, it resynchronizes the RT clock on any valid falling edge within the frame. Resynchronization within frames corrects misalignments between transmitter bit times and receiver bit times.

22.13.5.1 Slow Data Tolerance

Figure 22–28 shows how much a slow received frame can be misaligned without causing a noise error or a framing error. The slow stop bit begins at RT8 instead of RT1 but arrives in time for the stop bit data samples at RT8, RT9, and RT10.

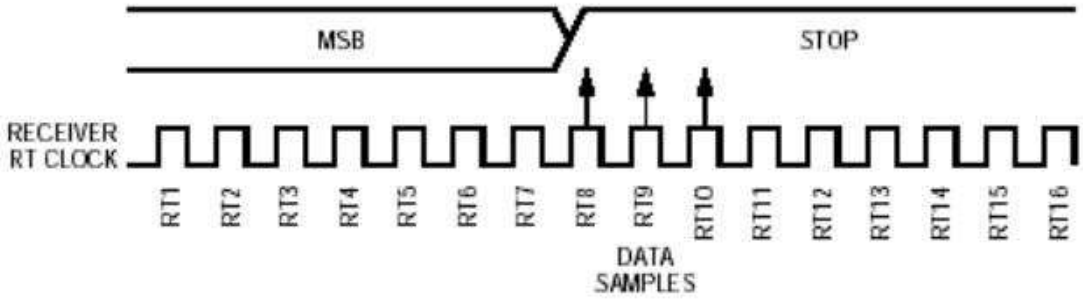


Figure 22–28: Slow Data

For 8-bit data, sampling of the stop bit takes the receiver:

9 bit times×16 RT cycles+10 RT cycles=154 RT cycles

With the misaligned data shown in **Figure 22–28** , the receiver counts 154 RT cycles at the point when the count of the transmitting device is:

9 bit times×16 RT cycles+3 RT cycles=147 RT cycles

The maximum percent difference between the receiver count and the transmitter count for slow 8-bit data with no errors is:

$$\left|\frac{154-147}{154}\right| \times 100 = 4.45\%$$

For 9-bit data, sampling of the stop bit takes the receiver:

10 bit times×16 RT cycles+10 RT cycles=170 RT cycles

With the misaligned data shown in **Figure 22–28**, the receiver counts 170 RT cycles at the point when the count of the transmitting device is:

10 bit times×16 RT cycles+3 RT cycles=163 RT cycles

The maximum percent difference between the receiver count and the transmitter count for slow 9-bit data with no errors is:

$$\left|\frac{170-163}{170}\right| \times 100 = 4.12\%$$

©Levetop Electronics Co., Ltd.

22.13.5.2 Fast Data Tolerance

Figure 22–29 shows how much a fast received frame can be misaligned without causing a noise error or a framing error. The fast stop bit ends at RT10 instead of RT16 but is still sampled at RT8, RT9, and RT10.

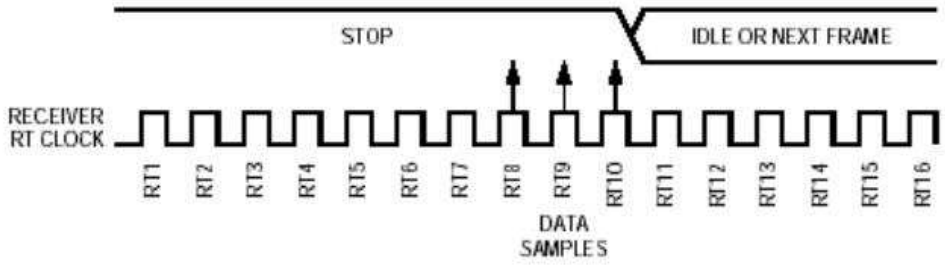


Figure 22–29: Fast Data

For 8-bit data, sampling of the stop bit takes the receiver:

9 bit times×16 RT cycles+10 RT cycles=154 RT cycles

With the misaligned data shown in **Figure 22–29**, the receiver counts 154 RT cycles at the point when the count of the transmitting device is:

10 bit times×16 RT cycles=160 RT cycles

The maximum percent difference between the receiver count and the transmitter count for fast 8-bit data with no errors is:

$$\left|\frac{154-160}{154}\right| \times 100 = 3.90\%$$

For 9-bit data, sampling of the stop bit takes the receiver:

10 bit times×16 RT cycles+10 RT cycles=170 RT cycles

With the misaligned data shown in **Figure 22–29**, the receiver counts 170 RT cycles at the point when the count of the transmitting device is:

11 bit times×16 RT cycles=176 RT cycles

The maximum percent difference between the receiver count and the transmitter count for fast 9-bit data with no errors is:

$$\left|\frac{170-176}{170}\right| \times 100 = 3.53\%$$

©Levetop Electronics Co., Ltd.

22.13.6 Receiver Wakeup

So that the SCI can ignore transmissions intended only for other devices in multiple-receiver systems, the receiver can be put into a standby state. Setting the RWU bit in SCICR2 puts the receiver into a standby state during which receiver interrupts are disabled.

The transmitting device can address messages to selected receivers by including addressing information in the initial frame or frames of each message.

The WAKE bit in SCICR1 determines how the SCI is brought out of the standby state to process an incoming message. The WAKE bit enables either idle line wakeup or address mark wakeup.

22.13.6.1 Idle Input Line Wakeup (WAKE = 0)

When WAKE = 0, an idle condition on the RXD pin clears the RWU bit and wakes up the receiver. The initial frame or frames of every message contain addressing information. All receivers evaluate the addressing information, and receivers for which the message is addressed process the frames that follow. Any receiver for which a message is not addressed can set its RWU bit and return to the standby state. The RWU bit remains set and the receiver remains on standby until another idle frame appears on the RXD pin.

Idle line wakeup requires that messages be separated by at least one idle frame and that no message contains idle frames.

The idle frame that wakes up the receiver does not set the IDLE flag or the RDRF flag.

The ILT bit in SCICR1 determines whether the receiver begins counting logic 1s as idle frame bits after the start bit or after the stop bit.

22.13.6.2 Address Mark Wakeup (WAKE = 1)

When WAKE = 1, an address mark clears the RWU bit and wakes up the receiver. An address mark is a 1 in the most significant data bit position. The receiver interprets the data as address data. When using address mark wakeup, the MSB of all non-address data must be 0. User code must compare the address data to the receiver's address and, if the addresses match, the receiver processes the frames that follow. If the addresses do not match, user code must put the receiver back to sleep by setting the RWU bit. The RWU bit remains set and the receiver remains on standby until another address frame appears on the RXD pin.

The address mark clears the RWU bit before the stop bit is received and sets the RDRF flag.

Address mark wakeup allows messages to contain idle frames but requires that the most significant byte (MSB) be reserved for address data.

Note:
With the WAKE bit clear, setting the RWU bit after the RXD pin has been idle can cause the receiver to wake up immediately.

©Levetop Electronics Co., Ltd.

22.14 Single-Wire Operation

Normally, the SCI uses the TXD pin for transmitting and the RXD pin for receiving (LOOPS = 0, RSRC = X). In single-wire mode, the RXD pin is disconnected from the SCI and is available as a general-purpose I/O pin. The SCI uses the TXD pin for both receiving and transmitting.

In single-wire mode (LOOPS = 1, RXRC = 1), setting the data direction bit for the TXD pin configures TXD as the output for transmitted data. Clearing the data direction bit configures TXD as the input for received data.

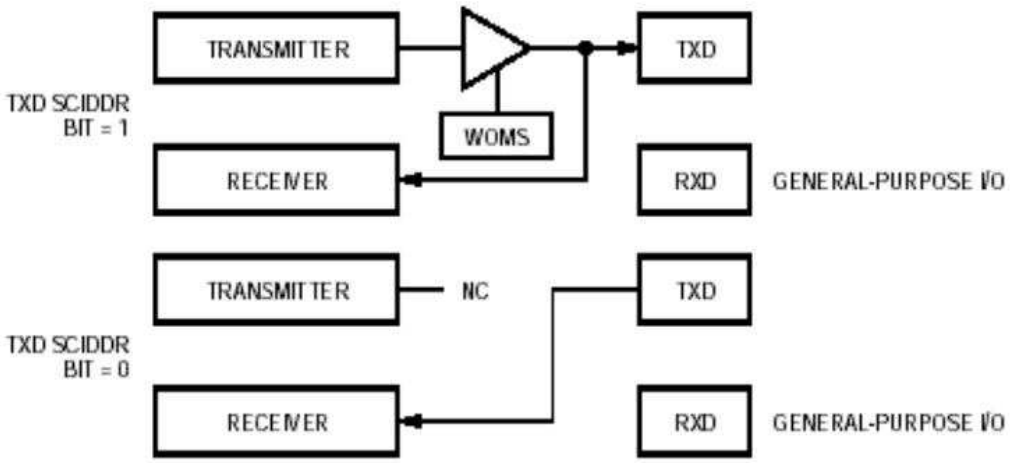


Figure 22–30: Single-Wire Operation (LOOPS = 1, RSRC = 1)

Enable single-wire operation by setting the LOOPS bit and the RSRC bit in SCICR1. Setting the LOOPS bit disables the path from the RXD pin to the receiver. Setting the RSRC bit connects the receiver input to the output of the TXD pin driver. Both the transmitter and receiver must be enabled (TE = 1 and RE = 1).

The WOMS bit in the SCICR1 register configures the TXD pin for full CMOS drive or for open-drain drive. WOMS controls the TXD pin in both normal operation and in single-wire operation. When WOMS is set, the DDR bit for the TXD pin does not have to be cleared for transmitter to receive data.

©Levetop Electronics Co., Ltd.

22.15 Loop Operation

In loop mode (LOOPS = 1, RSRC = 0), the transmitter output goes to the receiver input. The RXD pin is disconnected from the SCI and is available as a general-purpose I/O pin.

Setting the DDR bit for the TXD pin connects the transmitter output to the TXD pin. Clearing the data direction bit disconnects the transmitter output from the TXD pin.

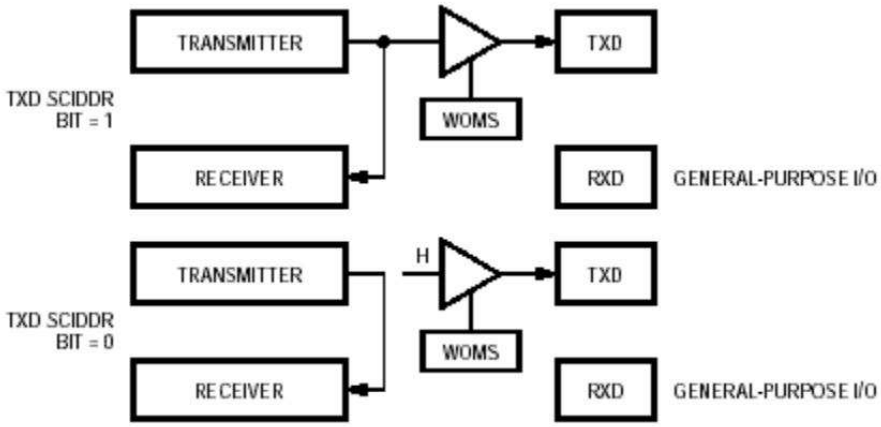


Figure 22–31: Loop Operation (LOOPS = 1, RSRC = 0)

Enable loop operation by setting the LOOPS bit and clearing the RSRC bit in SCICR1. Setting the LOOPS bit disables the path from the RXD pin to the receiver. Clearing the RSRC bit connects the transmitter output to the receiver input. Both the transmitter and receiver must be enabled (TE = 1 and RE = 1).

The WOMS bit in SCICR1 configures the TXD pin for full CMOS drive or for open-drain drive. WOMS controls the TXD pin during both normal operation and loop operation.

22.16 I/O Ports

The SCIPORT register is associated with two pins:

- The TXD pin is connected to SCIPORT1.
- The RXD pin is connected to SCIPORT0.

The SCI Data Direction Register (SCIDDR) configures the pins as inputs or outputs (see **22.7.7 SCI Data Direction Register**).

22.17 Reset

Reset initializes the SCI registers to a known startup state as described in **22.7 Memory Map and Registers.**

©Levetop Electronics Co., Ltd.

22.18 Interrupts

Lists the five interrupt requests associated with each SCI module.

Table 22–11: SCI Interrupt Request Sources

Source	Flag	Enable Bit
Transmitter	TDRE	TIE
	TC	TCIE
Receiver	RDRF	RIE
	OR	RIE
	IDLE	ILIE

22.18.1 Transmit Data Register Empty

The TDRE flag is set when the transmit shift register receives a byte from the SCI Data Register. It signals that SCIDRH and SCIDRL are empty and can receive new data to transmit. If the TIE bit in SCICR2 is also set, TDRE generates an interrupt request. Clear TDRE by reading SCISR1 and then writing to SCIDRL. Reset sets TDRE.

22.18.2 Transmission Complete

The TC flag is set when TDRE = 1 and no data, preamble, or break frame is being transmitted. It signals that no transmission is in progress. If the TCIE bit is set in SCICR2, TC generates an interrupt request. When TC is set, the TXD pin is idle (logic 1). TC is cleared automatically when a data, preamble, or break frame is queued. Clear TC by reading SCISR1 with TC set and then writing to the SCIDRL register. TC cannot be cleared while a transmission is in progress.

22.18.3 Receive Data Register Full

The RDRF flag is set when the data in the receive shift register transfers to SCIDRH and SCIDRL. It signals that the received data is available to be read. If the RIE bit is set in SCICR2, RDRF generates an interrupt request. Clear RDRF by reading SCISR1 and then reading SCIDRL.

22.18.4 Idle Receiver Input

The IDLE flag is set when 10 (if M = 0) or 11 (if M = 1) consecutive logic 1s appear on the receiver input. This signals an idle condition on the receiver input. If the ILIE bit in SCICR2 is set, IDLE generates an interrupt request. Once IDLE is cleared, a valid frame must again set the RDRF flag before an idle condition can set the IDLE flag. Clear IDLE by reading SCISR1 with IDLE set and then reading SCIDRL.

22.18.5 Overrun

The OR flag is set if data is not read from SCIDRL before the receive shift register receives the stop bit of the next frame. This signals a receiver overrun condition. If the RIE bit in SCICR2 is set, OR generates an interrupt request. The data in the shift register is lost, but the data already in SCIDRH and SCIDRL is not affected. Clear OR by reading SCISR1 and then reading SCIDRL.

©Levetop Electronics Co., Ltd.

23. USB2.0 Full-Speed Device Controller

23.1 Introduction

This section describes the USB2.0 Full Speed Device-Only controller. The device implementation in this module provides solutions for implementing a USB 2.0 full-speed/low-speed compliant peripheral.

23.2 Features

Features of the USB module include:

- USB2.0 Device-Only function controller
- USB 2.0 compliant
  - 12 Mbps full-speed (FS) data rate
  - USB data control logic:
    Packet identification and decoding/generation CRC generation and checking
    NRZI (non-return-to-zero inverted) encoding/decoding Bit-stuffing
    Sync detection
    End-of-packet detection
- Eight USB endpoints
- USB RAM
  - 2048 bytes of buffer RAM shared between system and USB module
  - RAM may be allocated as buffers for USB controller or extra system RAM resource
- USB reset options
  - USB Module reset generated by MCU
  - Bus reset generated by the host, which triggers a CPU interrupt
- Suspend and resume operations with remote wakeup support
- Transceiver features
  - Converts USB differential voltages to digital logic signal levels
  - On-chip USB pullup resistor
- On-chip 3.3-V regulator

©Levetop Electronics Co., Ltd.

23.3 Block Diagram

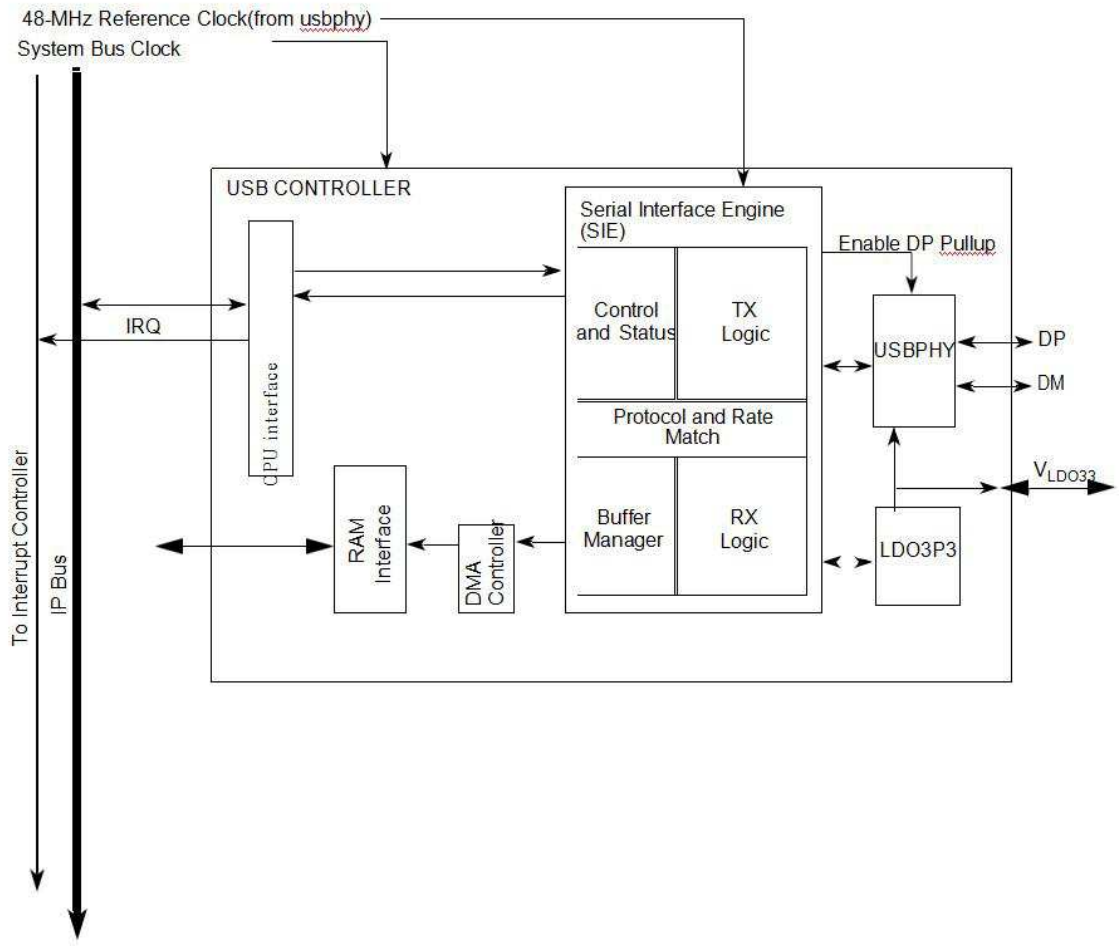


Figure 23–1: USB Full-speed Device(USB module) Block Diagram

©Levetop Electronics Co., Ltd.

23.4 Modes of Operation

This subsection describes the three low-power modes.

23.4.1 Wait Mode

In wait mode, the USB module can be continues to operate normally and can be configured to exit the low-power mode by generating an interrupt request.

23.4.2 Doze Mode

In Doze mode, the USB module can be continues to operate normally and can be configured to exit the low-power mode by generating an interrupt request.

23.4.3 Stop Mode

The USB Module is optionally available in stop mode. A reduced current consumption mode may be required for USB suspend mode per USB Specification Rev. 2.0, and stop mode is useful for achieving lower current consumption for the MCU and hence the overall USB device. Before entering stop via firmware, the user must ensure that the device settings are configured for stop such that the USB suspend current consumption targets are achieved

The USB module is notified about entering suspend mode when the SLEEP flag is set; this occurs after the USB bus is idle for 3ms. The USB device suspend mode current consumption level requirements are defined by the USB Specification Rev.2.0 (500uA for low-power and 2.5 mA for high-power with remote-wakeup enabled).

If USBRESMEN is set, and a K-state (resume signaling) is detected on the USB bus, the RESUME bit will become set. This will trigger an asynchronous interrupt that will wake the MCU from stop mode and enable clocks to the USB module.

©Levetop Electronics Co., Ltd.

23.5 Memory Map and Registers

This subsection describes the memory map and register structure for USB module.

23.5.1 Memory Map

Refer to for a description fo the memory map.

Table 23–1: USB Module Memory Map

Offset Address	Bits 31-0	Access(1)
0x0000	Reserved	S/U
0x0004	Reserved	S/U
0x0008	Reserved	S/U
0x000C	Reserved	S/U
0x001C	USBPHY Control Register 1 (USBPHY_CTRL1)	S/U
0x0080	Interrupt Status Register (INT_STAT)	S/U
0x0084	Interrupt Enable Register (INT_ENB)	S/U
0x0088	Error Interrupt Status Register (ERR_STAT)	S/U
0x008C	Error Interrupt Enable Register (ERR_ENB)	S/U
0x0090	Status Register (STAT)	S/U
0x0094	Control Register (CTL)	S/U
0x0098	Address Register (ADDR)	S/U
0x009C	EBT Page Register 1 (EBT_PAGE_01)	S/U
0x00A0	Frame Number Register (FRMNUML)	S/U
0x00A4	Frame Number Register (FRMNUMH)	S/U
0x00A8	Reserved	S/U
0x00AC	Reserved	S/U
0x00B0	EBT Page Register 2 (EBT_PAGE_02)	S/U
0x00B4	EBT Page Register 3 (EBT_PAGE_03)	S/U
0x00C0	Endpoint Control Registers (ENDPT0)	S/U
0x00C4	Endpoint Control Registers (ENDPT1)	S/U
0x00C8	Endpoint Control Registers (ENDPT2)	S/U
0x00CC	Endpoint Control Registers (ENDPT3)	S/U

©Levetop Electronics Co., Ltd.

Table 23–1: USB Module Memory Map (Continued)

Offset Address	Bits 31-0	Access(1)
0x00D0	Endpoint Control Registers (ENDPT4)	S/U
0x00D4	Endpoint Control Registers (ENDPT5)	S/U
0x00D8	Endpoint Control Registers (ENDPT6)	S/U
0x00DC	Endpoint Control Registers (ENDPT7)	S/U
0x0100	USBPHY Control Register 2 (USBPHY_CTRL2)	S/U
0x0104	Reserved	S/U
0x010C	USB Resume Wakeup Enable Register (USB_RESMEN)	S/U
0x0118	USBPHY Control Register 3 (USBPHY_CTRL3)	S/U

NOTES:

S = CPU supervisor mode access only. S/U = CPU supervisor or user mode access. User mode accesses to supervisor only addresses have no effect and result in a cycle termination transfer error.

©Levetop Electronics Co., Ltd.

23.5.2 Registers

23.5.2.1 USBPHY Control Register 1 (USBPHY_CTRL1)

The USBPHY Control Register controls the operation of Data Line termination resistors.

Register Offset Address: 0x001C

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	DP_HIGH	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–2: Interrupt Status Register (INT_STAT)

DP_HIGH— D+ Data Line pullup resistor enable
0 = D+ pullup resistor is not enabled
1 = D+ pullup resistor is enabled.

©Levetop Electronics Co., Ltd.

23.5.2.2 Interrupt Status Register (INT_STAT)

The Interrupt Status Register contains bits for each of the interrupt sources within the USB Module. Each of these bits are qualified with their respective interrupt enable bits. All bits of this register are logically OR'd together along with the Interrupt Status Register (INT_STAT) to form a single interrupt source for the processor's interrupt controller. After an interrupt bit has been set it may only be cleared by writing a one to the respective interrupt bit. This register contains the value of 0x00 after a reset.

Register Offset Address: 0x0080

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	STALL	0	RESUME	SLEEP	TOKDNE	SOF_TOK	ERROR	USB_RST
W								
RESET:	0	0	0	1	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–3: Interrupt Status Register (INT_STAT)

STALL— Stall Interrupt, and clear by writing 1 to this bit.
In Device mode this bit is asserted when a STALL handshake is sent by the USB Module. In Host mode this bit is set when the USB Module detects a STALL acknowledge during the handshake phase of a USB transaction. This interrupt can be use to determine whether the last USB transaction was completed successfully or if it stalled.

RESUME— Resume Interrupt and clear by writing 1 to this bit.
his bit is set depending on the DP/DM signals, and can be used to signal remote wake-up signaling on the USB bus. When not in suspend mode this interrupt should be disabled.

SLEEP— Sleep Interrupt, and clear by writing 1 to this bit.
This bit is set when the USB Module detects a constant idle on the USB bus for 3 milliseconds. The sleep timer is reset by activity on the USB bus.

TOKDNE— Token Done Interrupt, and clear by writing 1 to this bit.
This bit is set when the current token being processed has completed. The processor should immediately read the STAT register to determine the EndPoint and EB entry used for this token. Clearing this bit (by writing a one) causes the STAT register to be cleared or the STAT holding register to be loaded into the STAT register.

SOF_TOK— SOF Token Interrupt, and clear by writing 1 to this bit.
This bit is set when the USB Module receives a Start Of Frame (SOF) token. In Host mode this bit is set when the SOF threshold is reached, so that software can prepare for the next SOF.

©Levetop Electronics Co., Ltd.

ERROR— Error Interrupt, and clear by writing 1 to this bit.
This bit is set when any of the error conditions within the ERR_STAT register occur. The processor must then read the ERR_STAT register to determine the source of the error.

USB_RST— USB RST Interrupt, and clear by writing 1 to this bit.
This bit is set when the USB Module has decoded a valid USB reset. This informs the Microprocessor that it should write 0x00 into the address register and enable endpoint 0. USB_RST is set after a USB reset has been detected for 2.5 microseconds. It is not asserted again until the USB reset condition has been removed and then reasserted.

23.5.2.3 Interrupt Enable Register (INT_ENB)

The Interrupt Enable Register contains enable bits for each of the interrupt sources within the USB Module. Setting any of these bits enables the respective interrupt source in the INT_STAT register. This register contains the value of 0x00 after a reset.

Register Offset Address: 0x0084

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	STALL_EN		RESUME_EN	SLEEP_EN	TOKDNE_EN	SOF_TOK_EN	ERROR_EN	USB_RST_EN
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–4: Interrupt Enable Register (INT_ENB)

STALL_EN— Stall Interrupt Enable
0 = Stall Interrupt is disabled.
1 = Stall Interrupt is enabled.

RESUME_EN— Resume Interrupt Enable.
0 = Resume Interrupt is disabled.
1 = Resume Interrupt is enabled.

SLEEP_EN— Sleep Interrupt Enable.
0 = Sleep Interrupt is disabled.
1 = Sleep Interrupt is enabled.

TOKDNE_EN— Token Done Interrupt Enable.
0 = Token Done Interrupt is disabled.
1 = Token Done Interrupt is enabled.

SOF_TOK_EN— SOF Toke Interrupt Enable.
0 = SOF Token Interrupt is disabled.

©Levetop Electronics Co., Ltd.

1 = SOF Token Interrupt is enabled.

ERROR_EN— Error Interrupt Enable.
0 = Error Interrupt is disabled.
1 = Error Interrupt is enabled.

USB_RST_EN— USB Reset Interrupt Enable.
0 = USB Reset Interrupt is disabled.
1 = USB Reset Interrupt is enabled.

23.5.2.4 Error Interrupt Status Register (ERR_STAT)

The Error Interrupt Status Register contains enable bits for each of the error sources within the USB Module. Each of these bits are qualified with their respective error enable bits. All bits of this Register are logically OR'd together and the result placed in the ERROR bit of the ERR_STAT register. After an interrupt bit has been set it may only be cleared by writing a one to the respective interrupt bit. Each bit is set as soon as the error conditions is detected. Therefore, the interrupt does not typically correspond with the end of a token being processed. This register contains the value of 0x00 after a reset.

Register Offset Address: 0x0088

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	BTS_ERR	0	DMA_ERR	BTO_ERR	DFN8	CRC16	CRC5_EOF	PID_ERR
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–5: Interrupt Status Register (ERR_STAT)

BTS_ERR— This bit is set when a bit stuff error is detected. If set, the corresponding packet is rejected due to the error.

DMA_ERR— This bit is set if the USB Module has requested a DMA access to read a new EBT but has not been given the bus before it needs to receive or transmit data. If processing a TX transfer this would cause a transmit data underflow condition. If processing a RX transfer this would cause a receive data overflow condition. This interrupt is useful when developing device arbitration hardware for the microprocessor and the USB Module to minimize bus request and bus grant latency. This bit is also set if a data packet to or from the host is larger than the buffer size allocated in the EBT. In this case the data packet is truncated as it is put into buffer memory.

©Levetop Electronics Co., Ltd.

BTO_ERR— This bit is set when a bus turnaround timeout error occurs. The USB Module contains a bus turnaround timer that keeps track of the amount of time elapsed between the token and data phases of a SETUP or OUT TOKEN or the data and handshake phases of a IN TOKEN. If more than 16 bit times are counted from the previous EOP before a transition from IDLE, a bus turnaround timeout error occurs.

DFN8— This bit is set if the data field received was not 8 bits in length. USB Specification 1.0 requires that data fields be an integral number of bytes. If the data field was not an integral number of bytes, this bit is set.

CRC16— This bit is set when a data packet is rejected due to a CRC16 error.

CRC5_EOF— This error interrupt has two functions. When the USB Module is operating in device mode (HOST_MODE_EN=0), this interrupt detects CRC5 errors in the token packets generated by the host. If set the token packet was rejected due to a CRC5 error. When the USB Module is operating in host mode (HOST_MODE_EN=1), this interrupt detects End Of Frame (EOF) error conditions. This occurs when the USB Module is transmitting or receiving data and the SOF counter reaches zero. This interrupt is useful when developing USB packet scheduling software to ensure that no USB transactions cross the start of the next frame.

PID_ERR— This bit is set when the PID check field fails.

23.5.2.5 Error Interrupt Enable Register (ERR_ENB)

The Error Interrupt Enable Register contains enable bits for each of the error interrupt sources within the USB Module. Setting any of these bits enables the respective interrupt source in the ERR_STAT register. Each bit is set as soon as the error conditions is detected. Therefore, the interrupt does not typically correspond with the end of a token being processed. This register contains the value of 0x00 after a reset.

Register Offset Address: 0x008C

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	BTS_ERR_EN	0	DMA_ERR_EN	BTO_ERR_EN	DFN8_EN	CRC16_EN	CRC5_EOF_EN	PID_ERR_EN
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–6: Error Interrupt Enable Register (ERR_ENB)

©Levetop Electronics Co., Ltd.

BTS_ERR_EN— BTS_ERR Interrupt Enable

0 = BTS_ERR Interrupt is disabled.

1 = BTS_ERR Interrupt is enabled.

DMA_ERR_EN— DMA_ERR Interrupt Enable.
0 = DMA_ERR Interrupt is disabled.
1 = DMA_ERR Interrupt is enabled.

BTO_ERR_EN— BTO_ERR Interrupt Enable.
0 = BTO_ERR Interrupt is disabled.
1 = BTO_ERR Interrupt is enabled.

DFN8_EN— DFN8 Interrupt Enable.
0 = DFN8 Interrupt is disabled.
1 = DFN8 Interrupt is enabled.

CRC16_EN— CRC16 Interrupt Enable.
0 = CRC16 Interrupt is disabled.
1 = CRC16 Interrupt is enabled.

CRC5_EOF_EN— CRC5_EOF Interrupt Enable.
0 = CRC5_EOF Interrupt is disabled.
1 = CRC5_EOF Interrupt is enabled.

PID_ERR_EN— PID_ERR Interrupt Enable.
0 = PID_ERR Interrupt is disabled.
1 = PID_ERR Interrupt is enabled.

©Levetop Electronics Co., Ltd.

23.5.2.6 Status Register (STAT)

The Status Register reports the transaction status within the USB Module. When the processor's interrupt controller has received a TOK_DNE interrupt the Status Register should be read to determine the status of the previous endpoint communication. The data in the status register is valid when the TOK_DNE interrupt bit is asserted. The STAT register is actually a read window into a status FIFO maintained by the USB Module. When the USB Module uses a EB entry, it updates the Status Register. If another USB transaction is performed before the TOK_DNE interrupt is serviced, the USB Module stores the status of the next transaction in the STAT FIFO. Thus the STAT register is actually a four byte FIFO that allows the processor core to process one transaction while the SIE is processing the next transaction. Clearing the TOK_DNE bit in the INT_STAT register causes the SIE to update the STAT register with the contents of the next STAT value. If the data in the STAT holding register is valid, the SIE immediately reasserts to TOK_DNE interrupt.

Register Offset Address: 0x0090

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	ENDP[3:0]				TX	ODD	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–7: Status Register (STAT)

ENDP[3:0]— **Endpoint Number.**

These four bits encode the endpoint address that received or transmitted the previous token. This allows the microcontroller to determine which EBT entry was updated by the last USB transaction.

0000 Endpoint 0
0001 Endpoint 1
0010 Endpoint 2
0011 Endpoint 3
0100 Endpoint 4
0101 Endpoint 5
0110 Endpoint 6
0111 Endpoint 7

TX— Transmit Indicator.

0 = The most recent transaction was a Receive operation.
1 = The most recent transaction was a Transmit operation.

ODD— Odd/Even Transaction.

this bit is set if the last Endpoint Buffer Table updated was in the odd bank of the EBT.

©Levetop Electronics Co., Ltd.

23.5.2.7 Control Register (CTL)

The Control Register provides various control and configuration information for the USB Module.

Register Offset Address: 0x0094

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	J_STATE	SE0	TXSUS-PEND/TO-KENBUSY	RESET	HOST_MODE_EN	RESUME	ODD_RST	USB_EN/SOF_EN
W								
RESET:	0	0	0	0	0	0	0	0

Figure 23–8: Control Register (CTL)

JSTATE— Live USB differential receiver JSTATE signal
The polarity of this signal is affected by the current state of LS_EN

SE0— Live USB Single Ended Zero signal
The polarity of this signal is affected by the current state of LS_EN

TXSUSPEND/TOKENBUSY
When the USB Module is in Host mode TOKEN_BUSY is set when the USB Module is busy executing a USB token and no more token commands should be written to the Token Register. Software should check this bit before writing any tokens to the Token Register to ensure that token commands are not lost. In Device mode TXSUSPEND is set when the SIE has disabled packet transmission and reception. Clearing this bit allows the SIE to continue token processing. This bit is set by the SIE when a Setup Token is received allowing software to dequeue any pending packet transactions in the EBT before resuming token processing.

RESET— This bit is invalid for current device-role-only function.
Setting this bit enables the USB Module to generate USB reset signaling. This allows the USB Module to reset USB peripherals. Software must set RESET to 1 for the required amount of time and then clear it to 0 to end reset signaling. For more information on RESET signaling see Section 7.1.4.3 of the USB specification version 1.0.

RESUME

When set to 1 this bit enables the USB Module to execute resume signaling. This allows the USB Module to perform remote wake-up. Software must set RESUME to 1 for the required amount of time and then clear it to 0. If the HOST_MODE_EN bit is set, the USB module appends a Low Speed End of Packet to the Resume signaling when the RESUME bit is cleared. For more information on RESUME signaling see Section 7.1.4.5 of the USB specification version 1.0.

©Levetop Electronics Co., Ltd.

ODD_RST

Setting this bit to 1 resets all the EBT ODD ping/pong bits to 0, which then specifies the EVEN EBT bank.

USB_EN/SOF_EN— USB Enable.

Setting this bit causes the SIE to reset all of its ODD bits to the EBTs. Therefore, setting this bit resets much of the logic in the SIE. When host mode is enabled, clearing this bit causes the SIE to stop sending SOF tokens.

0 = The USB Module is disabled.
1 = The USB Module is enabled.

23.5.2.8 Address Register (ADDR)

The Address Register holds the unique USB address that the USB Module decodes when in Peripheral mode. This enables the USB Module to uniquely address an USB peripheral. In Peripheral mode, the USB_EN bit within the control register must be set. The Address Register is reset to 0x00 after the reset input becomes active or the USB Module decodes a USB reset signal. This action initializes the Address Register to decode address 0x00 as required by the USB specification.

Register Offset Address: 0x0098

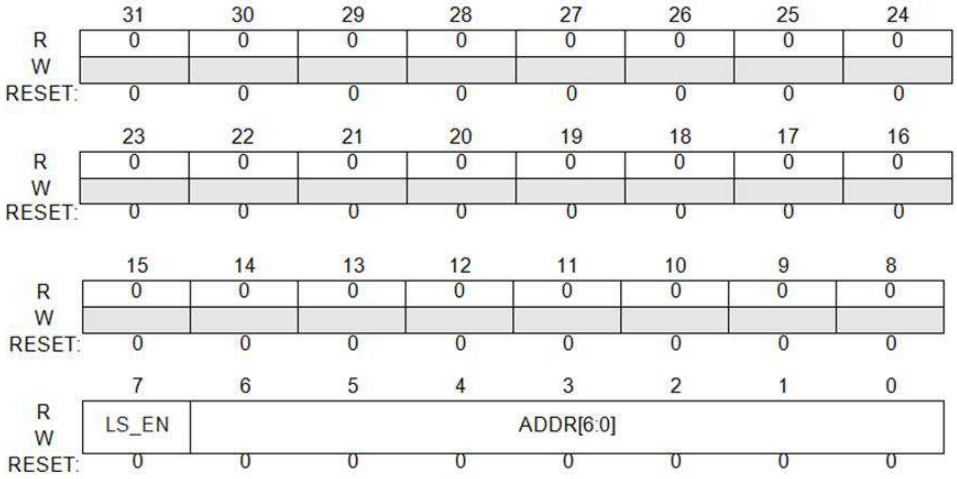


Figure 23–9: Address Register (ADDR)

LS_EN— Low Speed Enable bit.
This bit informs the USB Module that the next token command written to the token register must be performed at low speed. This enables the USB Module to perform the necessary preamble required for low-speed data transmissions.

ADDR[6:0]— USB address.
This 7-bit value defines the USB address that the USB Module decodes in peripheral mode, or transmit when in host mode.

©Levetop Electronics Co., Ltd.

23.5.2.9 EBT Page Register 1 (EBT_PAGE_01)

The Endpoint Buffer Table Page Register 1 contains an 7-bit value used to compute the address where the current Endpoint Buffer Table (EBT) resides in system memory.

Register Offset Address: 0x009C

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	EBT_BA[15:9]							0
W								
RESET:	0	0	0	0	0	0	0	0

Figure 23–10: EBT Page Register 1 (EBT_PAGE_01)

EBT_BA
This 7 bit field provides address bits 15 through 9 of the EBT base address, which defines where the Buffer Descriptor Table resides in system memory.

©Levetop Electronics Co., Ltd.

23.5.2.10 Frame Number Register (FRMNUML)

The Frame Number Registers contains the 11-bit frame number. The Frame Number Register requires two 8-bit registers to implement. The low order byte is contained in FRMNUML, and the high order byte is contained in FRMNUMH. These registers are updated with the current frame number whenever a SOF TOKEN is received.

Register Offset Address: 0x00A0

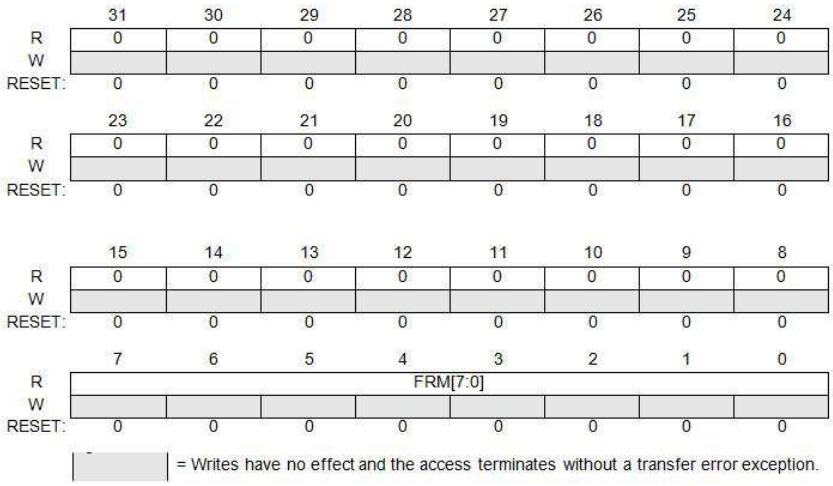


Figure 23–11: Frame Number Register (FRMNUML)

FRM[7:0]— **Frame Number**.
These bits represent the low order bits of the 11 bit Frame Number.

©Levetop Electronics Co., Ltd.

23.5.2.11 Frame Number Register (FRMNUMH)

The Frame Number Registers contains the 11-bit frame number. The Frame Number Register requires two 8-bit registers to implement. The low order byte is contained in FRMNUML, and the high order byte is contained in FRMNUMH. These registers are updated with the current frame number whenever a SOF TOKEN is received.

Register Offset Address: 0x00A4

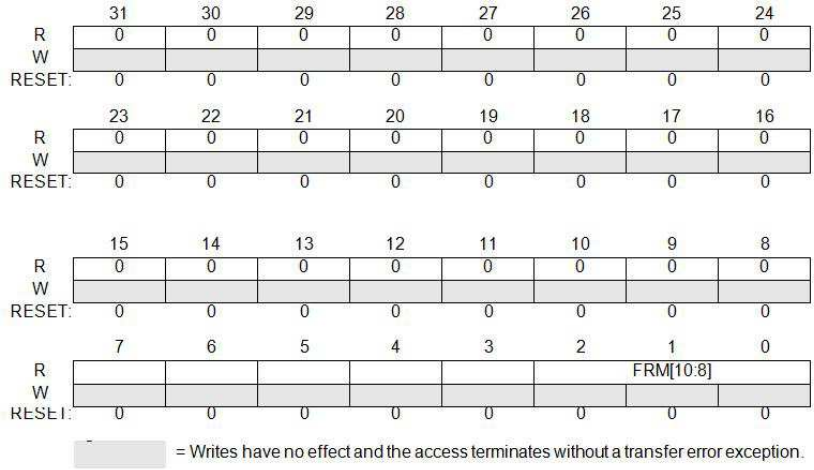


Figure 23–12: Frame Number Register (FRMNUMH)

FRM[10:8]— **Frame Number**.
These bits represent the high order bits of the 11-bit Frame Number.

23.5.2.12 EBT Page Register 2 (EBT_PAGE_02)

The Endpoint Buffer Table Page Register 2 contains an 8-bit value used to compute the address where the current Endpoint Buffer Table (EBT) resides in system memory.

Register Offset Address: 0x00B0

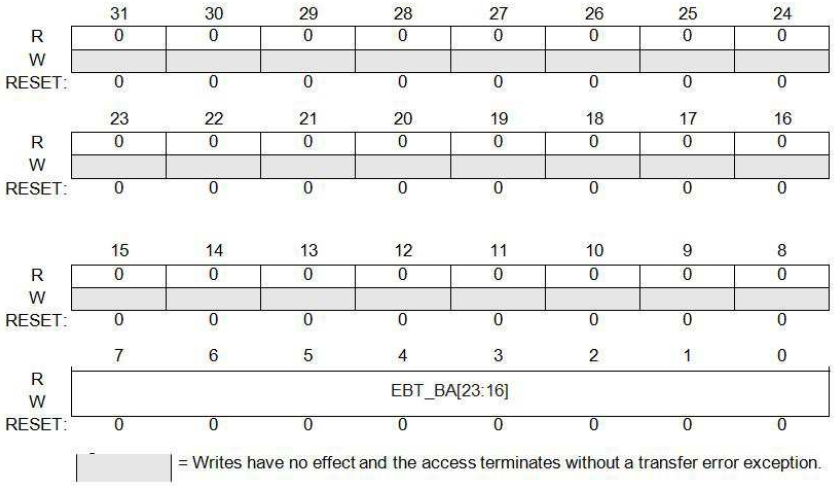


Figure 23–13: EBT Page Register 2 (EBT_PAGE_02)

EBT_BA
This 8 bit field provides address bits 23 through 16 of the EBT base address, which defines where the Buffer Descriptor Table resides in system memory.

©Levetop Electronics Co., Ltd.

23.5.2.13 EBT Page Register 3 (EBT_PAGE_03)

The Endpoint Buffer Table Page Register 3 contains an 8-bit value used to compute the address where the current Endpoint Buffer Table (EBT) resides in system memory.

Register Offset Address: 0x00B4

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	EBT_BA[31:24]							
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–14：EBT Page Register 3 (EBT_PAGE_03)

EBT_BA

This 8 bit field provides address bits 31 through 24 of the EBT base address, which defines where the Buffer Descriptor Table resides in system memory.

23.5.2.14 Endpoint Control Registers (ENDPTn, n=0-7)

The Endpoint Control Registers contain the endpoint control bits for each of the 8 endpoints available within the USB Module for a decoded address. The format for these registers is shown in the following figure. Endpoint 0 (ENDPT0) is associated with control pipe 0, which is required for all USB functions. Therefore, after a USB_RST interrupt occurs the processor core should set the ENDPT0 register to contain 0x0D.

In Host mode ENDPT0 is used to determine the handshake, retry and low speed characteristics of the host transfer. For Host mode control, bulk and interrupt transfers the EP_HSHK bit should be set to 1. For Isochronous transfers it should be set to 0. Common values to use for ENDPT0 in host mode are 0x4D for Control, Bulk, and Interrupt transfers, and 0x4C for Isochronous transfers.

Registers Offset address:

ENDPT0: = 0x00C0
ENDPT1: = 0x00C4
ENDPT2: = 0x00C8
ENDPT3: = 0x00CC
ENDPT4: = 0x00D0
ENDPT5: = 0x00D4
ENDPT6: = 0x00D8
ENDPT7: = 0x00DC

©Levetop Electronics Co., Ltd.

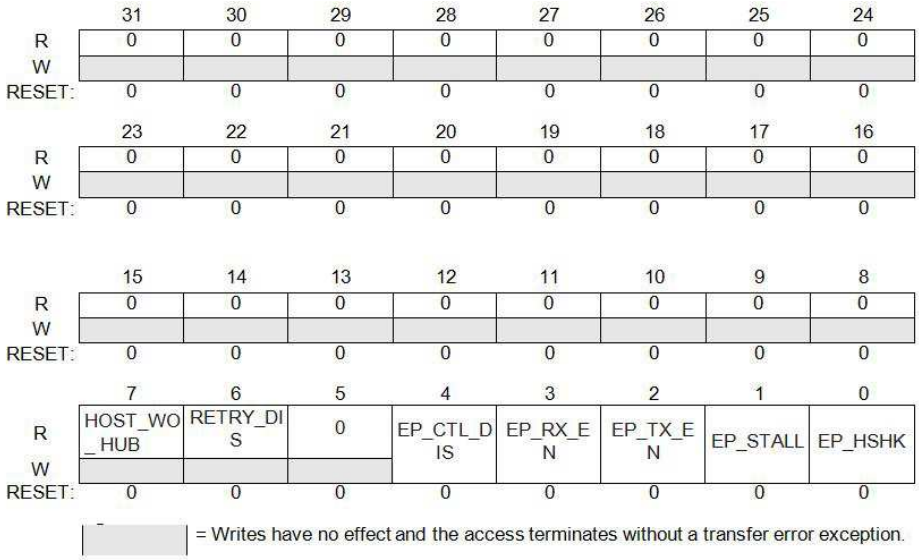


Figure 23–15：Endpoint Control Registers(ENDPTn, n=0-7)

EP_CTL_DIS— This bit, when set, disables control (SETUP) transfers. When cleared, control transfers are enabled. This applies if and only if the EP_RX_EN and EP_TX_EN bits are also set.

EP_RX_EN— This bit, when set, enables the endpoint for RX transfers.

EP_TX_EN— This bit, when set, enables the endpoint for TX transfers.

EP_STALL— When set this bit indicates that the endpoint is stalled. This bit has priority over all other control bits in the EndPoint Enable Register, but it is only valid if EP_TX_EN=1 or EP_RX_EN=1. Any access to this endpoint causes the USB Module to return a STALL handshake. After an endpoint is stalled it requires intervention from the Host Controller.

EP_HSHK— When set this bet enables an endpoint to perform handshaking during a transaction to this endpoint. This bit is generally set unless the endpoint is Isochronous.

Table 23–2：Endpoint Enable/Direction Control

Bit Name			Endpoint Enable/Direction Control
EP_CTL_D IS	EP_RX_ EN	EP_TX_ EN	
X	0	0	Disable endpoint
X	0	1	Enable endpoint for IN(TX) transfers only
X	1	0	Enable endpoint for OUT(RX) transfers only
0	1	1	Enable endpoint for IN, OUT and SETUP transfers.
1	1	1	RESERVE

©Levetop Electronics Co., Ltd.

Register Offset Address: 0x0100

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	SUSP	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–16：USBPHY Control Register 2(USBPHY_CTRL2)

SUSP— Places the USB transceiver into the suspend state.
0 = USB transceiver is not in suspend state.
1 = USB transceiver is in suspend state.

23.5.2.15 USB Resume Enable Register (USB_RESMEN)

Register Offset Address: 0x010C

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	0	0	USBRESMEN	USBPHY-CLKEN	0	0	USBIRQ	SUSPEND_RESUME
W	USB_SFT_RESET							
RESET:	0	0	0	0	0	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–17：USB Resume Enable Register (USB_RESMEN)

©Levetop Electronics Co., Ltd.

USB_SFT_RESET— USB Soft Reset control bit.
The USB Module will be reset by writing a one to this bit.

USBRESMEN— USB resume wakeup enable control bit.
If USBRESMEN is set, and a K-state (resume signaling) is detected on the USB bus, the RESUME bit will become set. This will trigger an asynchronous interrupt that will wake the MCU from stop mode and enable clocks to the USB module.

USBPHYCLKEN— USB PHY clock enable control bit.

USBIRQ— All enabled interrupt in INT_STAT register.
This bit will be set if any one of INT_STAT is set and the corresponding INT_ENB is set.

RESUME— USB DP/DM resume status bit.
If USBRESMEN is set, and a K-state (resume signaling) is detected on the USB bus, this bit will be set when CPU is waked-up from stop mode and clear by clear USBRESMEN bit.

23.5.2.16 USB PHY Control Register3 (USBPHY_CTRL3)

Register Offset Address: 0x0118

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	PHY_1.5V PD	PHY_OEN	PHY_RES ETb	USB_BYT E_SWAP	PHY_PD	PHY_SUS PEND	PHY_SUS PEND_SE L	48_60M_S EL
W								
RESET:	0	0	0	0	1	0	0	0

= Writes have no effect and the access terminates without a transfer error exception.

Figure 23–18：USB PHY Control Register 3 (USBPHY_CTRL3)

48_60M_SEL—USB PHY 48/60Mhz clock selection.
0 = 48Mhz is used as USB SIE decode clock.
1 = 60Mhz is used as USB SIE decode clock.

©Levetop Electronics Co., Ltd.

23.6 Function Description

The USB-FS 2.0 full-speed/low-speed module communicates with the processor core through status registers, control registers, and data structures in memory.

23.6.1 Data Structure

The function of the device operation is to transfer a request in the memory image to and from the Universal Serial Bus. To efficiently manage USB endpoint communications the USB module implements a Endpoint Buffer Table (EBT) in system memory. See .

23.6.2 Endpoint Buffer Table

To efficiently manage USB endpoint communications the USB module implements a Buffer Descriptor Table (EBT) in system memory. The EBT resides on a 512 byte boundary in system memory and is pointed to by the EBT Page Registers. Every endpoint direction requires two eight-byte Endpoint Buffer Table entries. Therefore, a system with 8 fully bidirectional endpoints would require 256 bytes of system memory to implement the EBT. The two Endpoint Buffer Table (EBT) entries allows for an EVEN EB entry and ODD EB entry for each endpoint direction. This allows the microprocessor to process one EB entry while the USB module is processing the other EB entry. Double buffering EB entrys in this way allows the USB SIE to easily transfer data at the maximum throughput provided by USB.

The software API intelligently manages buffers for the USB SIE by updating the EBT when needed. This allows the USB SIE to efficiently manage data transmission and reception, while the microprocessor performs communication overhead processing and other function dependent applications. Because the buffers are shared between the microprocessor and the USB module a simple semaphore mechanism is used to distinguish who is allowed to update the EBT and buffers in system memory. A semaphore bit, the OWN bit, is cleared to 0 when the EB entry is owned by the microprocessor. The microprocessor is allowed read and write access to the EB entry and the buffer in system memory when the OWN bit is 0. When the OWN bit is set to 1, the EB entry and the buffer in system memory are owned by the USB module. The USB module now has full read and write access and the microprocessor should not modify the EB entry or its corresponding data buffer. The EB entry also contains indirect address pointers to where the actual buffer resides in system memory. This indirect address mechanism is shown in the following diagram.

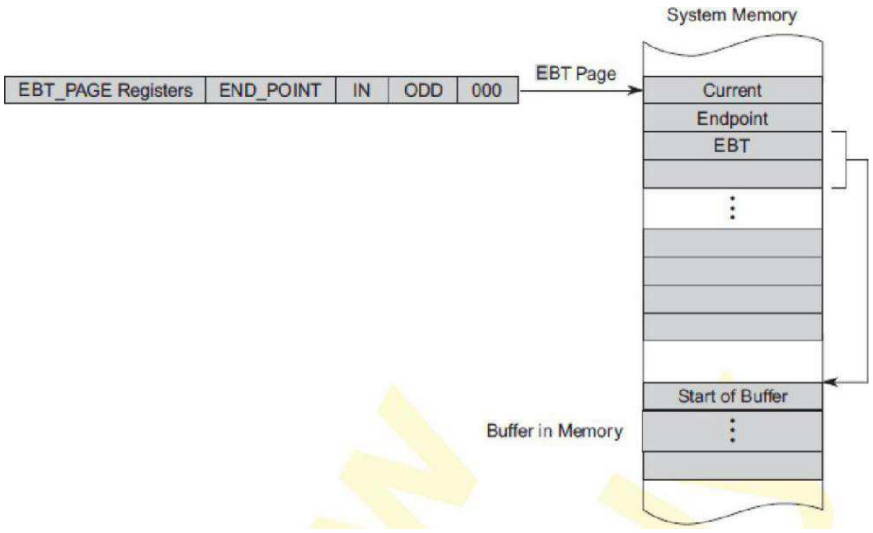


Figure 23–19：Endpoint Buffer Table

©Levetop Electronics Co., Ltd.

23.6.3 Rx vs. Tx as a USB Target Device

The centric nomenclature is used to describe the direction of the data transfer between the USB SIE core and the USB Host:

Rx (or receive)

describes transfers that move data from the USB to memory.

Tx (or transmit)

describes transfers that move data from memory to the USB.

The following table shows how the data direction corresponds to the USB token type in host and target device applications.

Table 23–3：Data Direction for USB Target Device

	RX	TX
Device	OUT or Setup	IN

23.6.4 Addressing Endpoint Buffer Table Entries

An understanding of the addressing mechanism of the Endpoint Buffer Table is useful when accessing endpoint data via the USB module or microprocessor. Some points of interest are:

- The Endpoint Buffer Table occupies up to 256 bytes of system memory.
- 8 bidirectional endpoints can be supported with a full EBT of 256 bytes.
- 16 bytes are needed for each USB endpoint direction.
- Applications with less than 16 endpoints require less RAM to implement the EBT.
- The EBT Page Registers point to the starting location of the EBT.
- The EBT must be located on a 512-byte boundary in system memory.
- All enabled TX and RX endpoint EB entries are indexed into the EBT to allow easy access via the USB module or CPU.

When a USB token on an enabled endpoint is received, the USB module uses its integrated DMA controller to interrogate the EBT. The USB SIE reads the corresponding endpoint EB entry to determine if it owns the EB entry and corresponding buffer in system memory.

To compute the entry point in to the EBT, the EBT_PAGE registers is concatenated with the current endpoint and the TX and ODD fields to form a 32-bit address. This address mechanism is shown in the following diagrams:

Table 23–4：EBT Address Calculation Fields

Field	Description
EBT_PAGE	EBT_PAGE registers in the Control Register Block
END_POINT	END POINT field from the USB TOKEN
TX	1 for an TX transmit transfers and 0 for an RX receive transfers
ODD	This bit is maintained within the USB SIE. It corresponds to the buffer currently in use. The buffers are used in a ping-pong fashion.

©Levetop Electronics Co., Ltd.

23.6.5 Endpoint Buffer Table Formats

The Endpoint Buffer Table (EBT) provide endpoint control information for the USB module and microprocessor. The Endpoint Buffer Tables have different meaning based on whether it is the USB module or microprocessor reading the EB entry in memory.

The USB SIE Controller uses the data stored in the EB entrys to determine:

- Who owns the buffer in system memory
- Data0 or Data1 PID
- Release Own upon packet completion
- No address increment (FIFO Mode)
- Data toggle synchronization enable
- How much data is to be transmitted or received
- Where the buffer resides in system memory

While the microprocessor uses the data stored in the EB entrys to determine:

- Who owns the buffer in system memory
- Data0 or Data1 PID
- The received TOKEN PID
- How much data was transmitted or received
- Where the buffer resides in system memory

The format for the EB entry is shown in the following figure.

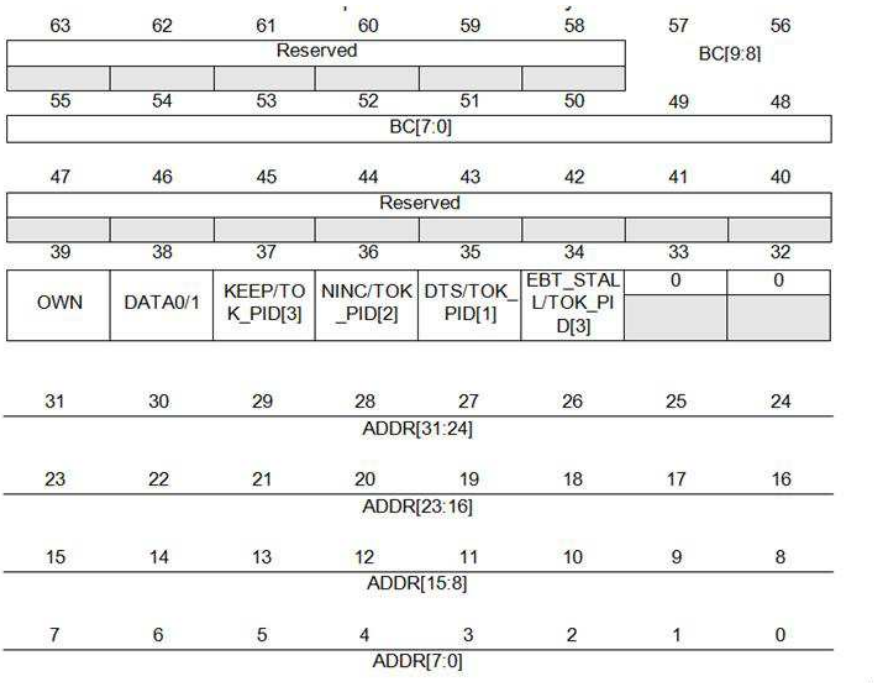


Figure 23–20：Endpoint Buffer Table Byte Format

©Levetop Electronics Co., Ltd.

Table 23–5: Endpoint Buffer Table Byte Fields

Field	Description
OWN	**OWN** — This OWN bit determines who currently owns the buffer. Except when KEEP=1,the USB SIE writes a 0 to this bit when it has completed a token. The USB module ignores all other fields in the EB entry when OWN=0. Once the EB entry has been assigned to the USB module (OWN=1), the MCU should not change it in any way. This byte of the EB entry should always be the last byte the MCU (firmware) updates when it initializes a EB entry. Although the hardware will not block the MCU from accessing the EB entry while owned by the USB SIE, doing so may cause undefined behavior and is generally not recommended. 0: The MCU has exclusive access to the entire EB entry 1: The USB module has exclusive access to the EB entry
DATA0/1	**Data Toggle** — This bit defines if a DATA0 field (DATA0/1=0) or a DATA1 (DATA0/1=1) field was transmitted or received. It is unchanged by the USB module. 0: Data 0 packet 1: Data 1 packet
KEEP/ TOK_PID[3]	**KEEP / EB entry Token PID [3]** —Typically this bit is set (that is, 1) with ISO endpoints feeding a FIFO. The microprocessor is not informed that a token has been processed, the data is simply transferred to or from the FIFO. If KEEP is set, normally the NINC bit is also set to prevent address increment. 0: Bit 3 of the current token PID is written back in to the EB entry by the USB SIE. Allows the USB SIE to release the EB entry when a token has been processed. 1: This bit is unchanged by the USB SIE. If the OWN bit also is set, the EB entry remains owned by the USB SIE forever.
NINC/ TOK_PID[2]	**No Increment / EB entry Token PID [2]** —The No Increment (NINC) bit disables the DMA engine address increment. This forces the DMA engine to read or write from the same address. This is useful for endpoints when data needs to be read from or written to a single location such as a FIFO. Typically this bit is set with the KEEP bit for ISO endpoints that are interfacing to a FIFO. 0: the USB SIE writes bit 2 of the current token PID to the EB entry. 1: This bit is unchanged by the USB SIE.

©Levetop Electronics Co., Ltd.

23.6.6 USB Transaction

When the USB SIE transmits or receives data, it computes the EBT address using the address generation shown in "Addressing Endpoint Buffer Table Entries" table.

If OWN =1, the following process occurs:

1. The USB SIE reads the EBT.
2. The SIE transfers the data via the DMA to or from the buffer pointed to by the ADDR field of the EB entry.
3. When the TOKEN is complete, the USB SIE updates the EBT and, if KEEP=0, changes the OWN bit to 0.
4. The STAT register is updated and the TOK_DNE interrupt is set.
5. When the microprocessor processes the TOK_DNE interrupt, it reads from the status register all the information needed to process the endpoint.
6. At this point, the microprocessor allocates a new EB entry so additional USB data can be transmitted or received for that endpoint, and then processes the last EB entry.

The following figure shows a timeline of how a typical USB token is processed after the EBT is read and OWN=1.

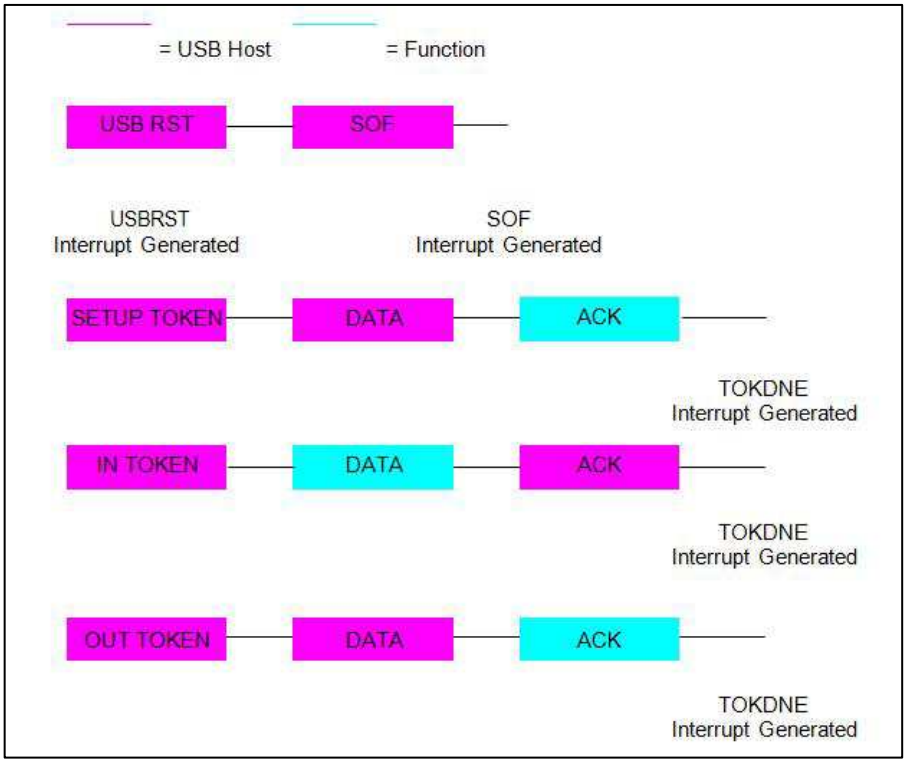


Figure 23–21: USB Token Transaction

The USB has two sources for the DMA overrun error:

Memory Latency

The memory latency on the DMA interface may be too high and cause the receive FIFO to overflow. This is predominantly a hardware performance issue, usually caused by transient memory access issues.

©Levetop Electronics Co., Ltd.

Oversized Packets

The packet received may be larger than the negotiated MaxPacket size. Typically, this is caused by a software bug. For DMA overrun errors due to oversized data packets, the USB specification is ambiguous. It assumes correct software drivers on both sides. NAKing the packet can result in retransmission of the already oversized packet data. Therefore, in response to oversized packets, the USB core continues ACKing the packet for non-isochronous transfers.

Table 23–6: USB Responses to DMA Overrun Errors

Errors due to Memory Latency	Errors due to Oversized Packets
Non-Acknowledgment (NAK) or Bus Timeout (BTO) — See bit 4 in "Error Interrupt Status Register (ERR_STAT)" as appropriate for the class of transaction.	Continues acknowledging (ACKing) the packet for non synchronous transfers.
—	The data written to memory is clipped to the MaxPacket size so as not to corrupt system memory.
The DMA_ERR bit is set in the ERR_STAT register. Depending on the values of the INT_ENB and ERR_ENB register, the core may assert an interrupt to notify the processor of the DMA error.	Asserts the DMA_ERR bit of the ERR_STAT register (which could trigger an interrupt) and a TOK_DNE interrupt fires. (Note: The TOK_PID field of the EBT is not 1111 because the DMA_ERR is not due to latency).
the EBT is not written back nor is the TOK_DNE interrupt triggered because it is assumed that a second attempt is queued and will succeed in the future.	The packet length field written back to the EBT is the MaxPacket value that represents the length of the clipped data actually written to memory.
From here, the software can decide an appropriate course of action for future transactions such as stalling the endpoint, canceling the transfer, disabling the endpoint, etc.	

©Levetop Electronics Co., Ltd.

24. I2C

24.1 Introduction

I2C is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange, minimizing the interconnection between devices. This bus is suitable for applications requiring occasional communications over a short distance between many devices. The flexible I2C allows additional devices to be connected to the bus for expansion and system development.

The two bus lines provide data transfer rates up to 100Kbits/s IN Standard Mode, data rates up to 400Kbits/s in Fast Mode and data rates up to 3.4Mbits/s in High-Speed Mode.

The I2C system is a true multiple-master bus including arbitration and collision detection that prevents data corruption if multiple devices attempt to control the bus simultaneously. This feature supports complex applications with multiprocessor control and can be used for rapid testing and alignment of end products through external connections to an assembly-line computer.

24.2 Features

- Supports 7 bit addressing.
- Supports Standard Mode, Fast Mode and High-Speed Mode
- Software option to select between High-Speed mode and Standard/Fast mode
- Compatibility with standard and fast-mode of I2C bus version 2.1 standard.
- Multiple-master operation.
- Software-programmable for one of 64 different serial clock frequencies.
- Software-selectable acknowledge bit.
- Interrupt-driven, byte-by-byte data transfer.
- Arbitration-lost interrupt with automatic mode switching from master to slave.
- Transfer completion and read configure interrupt.
- Start and stop signal generation/detection.
- Repeated START signal generation.
- Acknowledge bit generation/detection.
- Bus-busy detection.
- Option slave address receiving enable when system clock stop mode
- SCL or SDA line GPIO function supported

©Levetop Electronics Co., Ltd.

24.3 System and Block Diagram

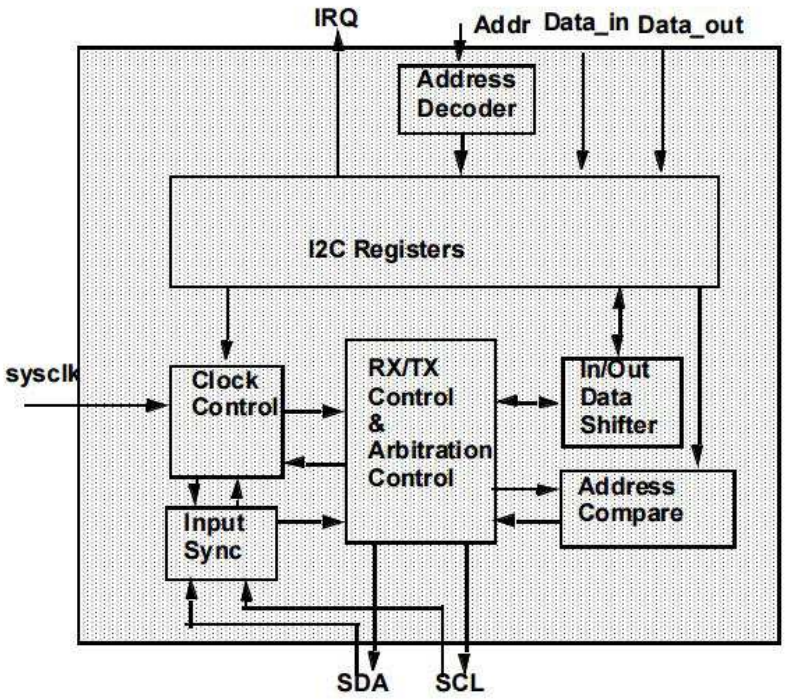


Figure 24–1: I2C Block Diagram

©Levetop Electronics Co., Ltd.

24.4 Memory Map and Registers (Base: 0x4015_0000)

There are five registers in the I2C memory map, as shown in **Table 24–1**.

Table 24–1: I2C Memory Map

Address Offset	Bits 7- 0
0x0000	I2C Status Register (I2CS)
0x0001	I2C Clock Prescaler Register (I2CP)
0x0002	I2C Control Register (I2CC)
0x0003	I2C Slave Address Register (I2CSA)
0x0004	I2C Port Control Register (I2CPCR)
0x0005	I2C slave high-speed indicator register
0x0006	I2C slave SDA hold time Register
0x0007	I2C Data Register (I2CD)
0x0008	Reversed
0x0009	Reversed
0x000a	I2C Port Direction Register (I2CDDR)
0x000b	I2C Port Data Register (I2CPDR)

24.4.1 I2C Status Register (I2CS)

Address Offset: 0x0000

	Bit7	6	5	4	3	2	1	Bit0
Read	AACK	DACK	RXTX	ARBL	BBUSY	AASLV	RC	TF
Write								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–2: I2CS Register

The I2CS register shows the status of I2C module.

TF - Transfer Complete Flag

Indicates there is data transmitted or received. For receiver, It is set by the falling edge of the ninth clock of a data (not address) byte received regardless of acknowledge bit detected or not. For master transmitter, It is set by the falling edge of the ninth clock of a data or address byte send regardless of acknowledge bit detected or not. For slave transmitter, it is set by the falling edge of the ninth clock of address or data byte transferred and acknowledging bit must be detected. If the IEN bit in I2CC is also set, an interrupt will be generated. Clear TF by reading I2CS with TF set and then accessing I2CD or master-transmit mode writing I2CC.

0 = meaningless,

1 = Data or address transfer complete.

©Levetop Electronics Co., Ltd.

RC - Receive Complete

Indicates it is time to configure the receiver. For master receiver, It is set by the falling edge of the ninth clock of data or address byte transferred regardless of acknowledge bit detected or not. For slave receiver, it is set by the falling edge of the ninth clock of address or data byte received and acknowledgement bit must be received. If the IEN bit in I2CC is also set, an interrupt will be generated. Clear RC by reading I2CS with RC set and then writing I2CC.

0 = meaningless,

1 = Indicates it is time to configure the receiver.

AASLV - Addressed as a slave

It is set by the falling edge of the eighth clock if I2C module is addressed as a slave and its own slave address matches the calling address received on SDL. It is clear by START or STOP bit detected

0 = Not as a slave.

1 = Addressed as a slave.

BBUSY - I2C bus busy

Show the bus status.

0 = Bus is idle. It is cleared from STOP bit.

1 = Bus is busy. It is set from START bit.

ARBL - Arbitration lost

Show the arbitration status of the bus. It will be set in the following cases during SCL high:

1. SDA is sampled low when the master drives high during START condition, address cycle, data-transmit cycle or STOP condition.
2. SDA is sampled low when the master drives high during the acknowledge bit of a data-receive cycle.
3. A start cycle is attempted when the bus is busy.

ARBL must be cleared by software by reading the I2CS register.

0 = Arbitration not lost.

1 = Arbitration lost.

RXTX - Receive or transmit.

Indicate the I2C module function as receiver or transmitter. It is valid of falling edge of the eighth clock.

0 = Receive, receive data.

1 = Transmit, transmit data.

DACK - Data acknowledge received.

Indicates whether the acknowledge bit is detected during address or data transfer. It is valid of rising edge of the ninth clock.

0 = No acknowledge bit received.

1 = Acknowledge bit received.

AACK - Address acknowledge error.

Indicates whether the acknowledge bit was detected or not by master during address phase. It is set by the rising edge of ninth clock of the address phase and clear by changing MSMOD from 1 to 0 or repeat start condition.

0 = No address acknowledge error.

1 = Address acknowledge error.

©Levetop Electronics Co., Ltd.

24.4.2 I2C Clock Prescalar Register (I2CP)

Address Offset: 0x0001

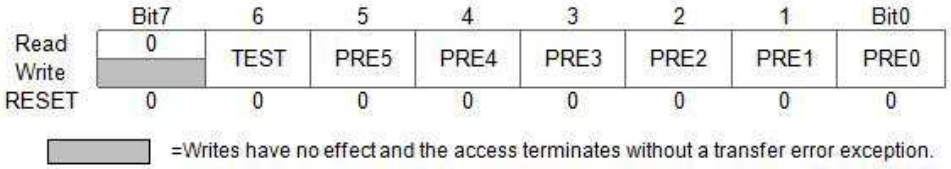


Figure 24–3: I2CP Register

I2CP is a prescaler to generate a bit-rate clock for the data transceiver. Due to potentially slow SCL and SDA rise and fall times, bus signals are sampled at the prescaler frequency. The serial bit clock frequency is equal to the OPB clock divided by the divider.

PRE[5:0] - Prescaler Divider Value

TEST - Clock Test Enable

It enables test mode for the I2C clock. In normal mode, its frequency is fopb/(396x(PRE[5:0]+1)+1). In test mode, the frequency is fopb/(8x(PRE[5:0]+1)+1).

1 = Clock Test Mode Enable

0 = Normal Mode

24.4.3 I2C Control Register (I2CC)

Address Offset: 0x0002

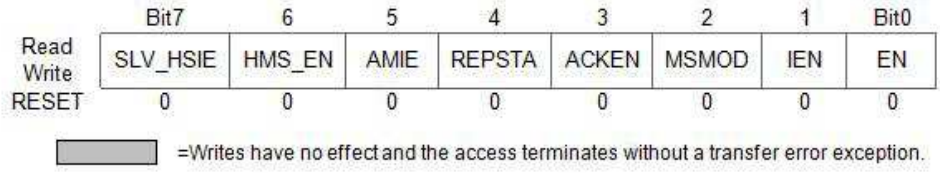


Figure 24–4: I2CC Register

I2CC is used to control the working of I2C module.

EN - I2C enable/disable control.

1 = module is enabled.

0 = module is disabled.

It enables/disables the module. Also controls the software reset of the entire I2C module. Setting the bit generates an internal reset to the module which gets asserted after 2 clock of setting the bit and remains asserted for 3 clocks. Thus reset gets negated after 5 clocks of setting EN bit.
If the module is enabled in the middle of a byte transfer, slave mode ignores the current I2C bus transfer and starts operating when the next start condition is detected. Master mode is not aware that the bus is busy; so initiating a start cycle may corrupt the current bus cycle, ultimately causing either the current master or the I2C module to lose arbitration, after which bus operation returns to normal.

IEN - I2C interrupt enable control.

It enables the I2C interrupt when it is set.

1 = I2C interrupt enabled.

0 = I2C interrupt disabled.

©Levetop Electronics Co., Ltd.

MSMOD - I2C master/slave mode selection control.

Changing MSMOD from 1 to 0 generates a STOP on the bus and selects slave mode. Changing MSMOD from 0 to 1 generates a START on the bus and selects master mode.

1 = Master mode.

0 = Slave mode.

ACKEN - Acknowledge enable control.

Specifies the value driven onto SDA during acknowledge cycles for both master and slave receivers. Note that ACKEN bit applies only when the I2C bus receives data byte. During receiving address, if the received address is matched with the slave address, the Acknowledge bit will be sent regardless of the ACKEN bit.

1 = An acknowledge signal is sent to the SDA at the ninth clock bit after receiving one byte of data.

0 = No acknowledge signal is sent to the SDA at the ninth clock bit after receiving one byte of data.

REPSTA - Repeat Start

In these case: 1) The master receiver wasn't acknowledged after sending slave address or data byte, 2) the master transmitter has send a address or data byte regardless of acknowledged or not , the master can repeat the START signal, followed by a new slave address. The repeat start bit will be send when configure slave address (by writing I2CD) after REPSTA bit set.

1 = Generate repeat start.

0 = No repeat start.

AMIE - Address Match Interrupt Enable

Select whether Slave address match will generate interrupt request if I2C interrupt enable control is set to 1. AMIE should be set before entering stop mode to wakeup the system when slave address matched and must be clear when normal work mode.

1 = Enable address match interrupt request.

0 = Disable address match interrupt request.

HSM_EN - High Speed Mode Enable

Select the High Speed mode or Fast/Standard mode operation for the module in master mode. The HSM_EN bit should be set to select the High Speed mode operation. After the transmission of the master code in F/S mode and the HS_I2C does not loose the arbitration, HSM_EN should be set by software, and if you prepare to clear MSMOD to transfer STOP, HSM_EN should be clear by software before after MSMOD.

0 = Fast/Standard mode operation (Default)

1 = High-Speed Mode operation

SLV_HSIE - Slave High-Speed Mode Interrupt Enable

Select whether Slave High-Speed Mode status will generate interrupt request if 2C interrupt enable control is set to 1.

1 = Enable Slave High-Speed Mode status request.

0 = Disable Slave High-Speed Mode status request.

©Levetop Electronics Co., Ltd.

24.4.4 I2C Slave Address Register (I2CSA)

Address Offset: 0x0003

	Bit7	6	5	4	3	2	1	Bit0
Read Write	ADDR7	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	0
RESET	0	1	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–5: I2CSA Register

The I2CSA stores the address when the I2C works as slave and responds to the address sent by a master.

ADDR[7:1] - Module Slave Address.
Slave address when the I2C module is in slave mode. (I2C is slave by default).

24.4.5 I2C Port Control Register (I2CPCR)

Address Offset: 0x0004

	Bit7	6	5	4	3	2	1	Bit0
Read Write	SDAPA	SCLPA	WOMI2C[1:0]		PDI2C[1:0]		PUI2C[1:0]	
RESET	0	0	0	0	0	0	1	1

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–6: I2CPCR Register

SDAPA - SDA Port Assignment Bit.
The read/write bit selects the SDA's function mode.
1 = Pin configured as GPIO.
0 = Pin configured as primary function.

SCLPA - SCL Port Assignment Bit.
The read/write bit selects the SCL's function mode.
1 = Pin configured as GPIO.
0 = Pin configured as primary function.

WOMI2C[1:0] — Wired-OR mode Bits
The read/write bits set the corresponding I2C pins to open-drain drive mode. WOMI2C[1] is for SDA pin and WOMI2C[0] is for SCL pin. These bits are available only in GPIO mode
1 = Open-drain when output.
0 = CMOS drive when output.

PDI2C[1:0] — Pull-down Enable Bits
The read/write bits enable the pull-downs of corresponding I2C pins. PDI2C[1] is for SDA pin and PDI2C[0] is for SCL pin. These bits are available both in GPIO and main function mode
1 = Pullup Enabled.
0 = Pullup Disabled.

PUI2C[1:0] — Pull-up Enable Bits
The read/write bits enable the pullups of corresponding I2C pins. PUI2C[1] is for SDA pin and PUI2C[0] is for SCL pin. These bits are available both in GPIO and main function mode
1 = Pullup Enabled.
0 = Pullup Disabled.

©Levetop Electronics Co., Ltd.

24.4.6 I2C Slave High-Speed Mode Indicator Register (I2CSHIR)

Address Offset: 0x0005

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	0	0	0	0	0	SLV_HS
Write								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–7: I2CSHIR Register

SLV_HS -Slave high speed mode.

When I2C is selected as slave, this bit indicates whether the module is selected for High-Speed mode or Fast/Standard mode data transfer. This bit is set on the ninth SCL rising edge of Master code and Not Acknowledgement bit is received. This bit must be cleared by writing 1 to it otherwise the SCL line will be force to LOW state.

0 = I2C is selected for Fast/Standard data transfer (Default).

1 = I2C is selected for High-Speed mode data transfer.

24.4.7 I2C Slave SDA Hold Time Register (I2CSHT)

Address Offset: 0x0006

	Bit7	6	5	4	3	2	1	Bit0
Read	SCL_FILTER_EN	SDA_FILTER_EN	SLVHT					
Write								
RESET	0	0	0	0	1	0	0	1

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–8: I2CSHT Register

SCL_FILTER_EN - SCL Filter enable.

If SCL filter is enabled, when the HS mode transfer is ongoing, the 10ns pulse occurred on SCL line will be filter out. If the F/S mode transfer is ongoing, the 50ns pulse occurred on SCL line will be filter on.

SDA_FILTER_EN - SDA Filter enable.

If SDA filter is enabled, when the HS mode transfer is ongoing, the 10ns pulse occurred on SDA line will be filter out. If the F/S mode transfer is ongoing, the 50ns pulse occurred on SDA line will be filter on.

SLVHT - slave SDA line hold time configuration.

When I2C work as slave output mode, the data will be changed after SCL falling edge and the value of internal SDA hold register equal to SLVHT. Writing value of 0 is not allowed.

©Levetop Electronics Co., Ltd.

24.4.8 I2C Data Register (I2CD)

Address Offset: 0x0007

	Bit7	6	5	4	3	2	1	Bit0
Read	R7	R6	R5	R4	R3	R2	R1	R0
Write	T7	T6	T5	T4	T3	T2	T1	T0
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 24–9: I2CD Register

The I2CD register holds the data to be transmitted (next byte) or data received. In master mode it also holds the slave address and transfer direction to be transmitted. Bits [7:1] form the slave address and bit [0] is the transfer direction (R/W). In master-receiver mode, reading I2CD allows a read to occur and initiates next byte data receiving. In master-transmit mode, writing I2CD will store the byte of the next transmit. In slave mode, the same function is available after it is addressed.

R[7:0] - Receive Bits [7:0]
T[7:0] - Transmit Bits [7:0]

24.4.9 I2C Port Direction Register (I2CDDR)

Address Offset: 0x000a

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	0	0	0		DDRI2C[1:0]	
Write								
RESET	0	0	0	0	0		0	0

Figure 24–10: I2CDDR Register

Read: Anytime

Write: Anytime

DDRI2C[1:0] — I2C Data Direction Bits
The read/write bits control the data direction of the I2C pins. These bits are available only in GPIO mode
1 = Corresponding pin configured as output
0 = Corresponding pin configured as input

©Levetop Electronics Co., Ltd.

24.4.10 I2C Port Data Register (I2CPDR)

Address Offset: 0x000b

	Bit7	6	5	4	3	2	1	Bit0
Read	0	0	0	0	0	0	PORTI2C[1:0]	
Write								
RESET	0	0	0	0	0	0	0	0

Figure 24–11: I2CDDR Register

Read: Anytime
Write: Anytime

PORTI2C[1:0] — I2C Port Data Bits

Writes to these bits set the output data for the corresponding I2C pins configured as GPIO. Reading these bits return the I2C pins' level.

©Levetop Electronics Co., Ltd.

24.5 Functional Description

The I2C module is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange, minimizing the interconnection between devices. Software can poll the I2C status flags or I2C operation can be interrupt driven. When a byte is received or to be transmitted, the TF bit in I2CS will be set and if the IEN bit in I2CC is also set, an interrupt will be generated.

If arbitration is lost during its transfer, the ARBL will be set and if the IEN bit in I2CC is also set, an interrupt will be generated. An interrupt can also be generated if there is no address acknowledge from the slave (AACK set).

The module can clear ACKEN if it does not want to receive next byte.

24.5.1 Master Mode

The I2C module may initial a transfer if the bus is free (the BBUSY bit of I2CS is clear) when it works as a master. Changing the MSMOD bit from 0 to 1 generates a START on the bus and selects master mode. The master controls the direction of transfer, namely the R/W bit. The first byte of a transfer sent by the master is the slave address, the following byte is data. Change the MSMOD bit from 1 to 0 to generate a STOP on the bus if no acknowledge is received after each ninth cycle of clock. The PRE[5:0] bits in I2CP control the bit-rate clock of I2C-bus.

By configuring slave address after setting the REPSTA bit, the master can repeat the START signal instead of signaling a STOP.

24.5.2 Slave Mode

If the MSMOD bit is cleared the module is a slave and can be addressed by other masters. When a winning master is trying to address it, it will release the SDA line and switch over immediately to its slave mode.

Note:
The I2C can't work in slave mode and master mode at the same time.

24.5.3 Protocol

The I2C communication protocol consists of six components: START, Data Source/Recipient, Data Direction, Slave Acknowledge, Data, Data Acknowledge and STOP. These are shown in **Figure 24–12** and described in the following text.

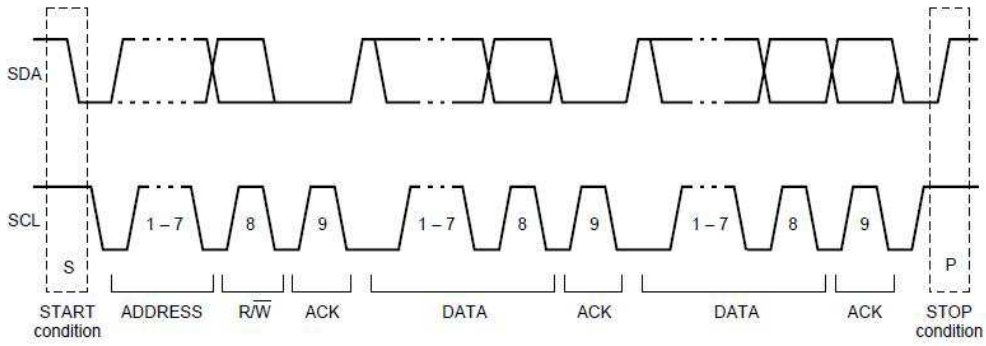


Figure 24–12: I2C Communication Protocol

©Levetop Electronics Co., Ltd.

1. START signal—When no other device is bus master (both SCL and SDA lines are at logic high), a device can initiate communication by sending a START signal (see A in **Figure 24–12**). A START signal is defined as a high-to-low transition of SDA while SCL is high. This signal denotes the beginning of a data transfer (each data transfer can be several bytes long) and awakens all slaves.

2. Slave address transmission—The master sends the slave address in the first byte after the START signal (B). After the seven-bit calling address, it sends the R/W bit (C), which tells the slave the data transfer direction.

Each slave must have a unique address. An I2C master must not transmit an address that is the same as its own slave address; it cannot be master and slave at the same time. The slave whose address matches that sent by the master pulls SDA low at the ninth clock (D) to return an acknowledge bit.

3. Data transfer—When successful slave addressing is achieved, the data transfer can proceed (E) on a byte-by-byte basis in the direction specified by the R/W bit sent by the calling master.

Data can be changed only while SCL is low and must be held stable while SCL is high, as **Figure 24–12** shows. SCL is pulsed once for each data bit, with the MSB being sent first. The receiving device must acknowledge each byte by pulling SDA low at the ninth clock; therefore, a data byte transfer takes nine clock pulses.

If it does not acknowledge the master, the slave receiver must leave SDA high. The master can then generate a STOP signal to abort the data transfer or generate a START signal (repeated start, shown in **Figure 24–13**) to start a new calling sequence.

If the master receiver does not acknowledge the slave transmitter after a byte transmission, it means end-of-data to the slave. The slave releases SDA for the master to generate a STOP or START signal.

4. STOP signal—The master can terminate communication by generating a STOP signal to free the bus. A STOP signal is defined as a low-to-high transition of SDA while SCL is at logical high (F). Note that a master can generate a STOP even if the slave has made an acknowledgment, at which point the slave must release the bus.

5. Instead of signaling a STOP, the master can repeat the START signal, followed by a calling command, (A in **Figure 24–13**). A repeated START occurs when a START signal is generated without first generating a STOP signal to end the communication. The master uses a repeated START to communicate with another slave or with the same slave in a different mode (transmit/receive mode), without releasing the bus.

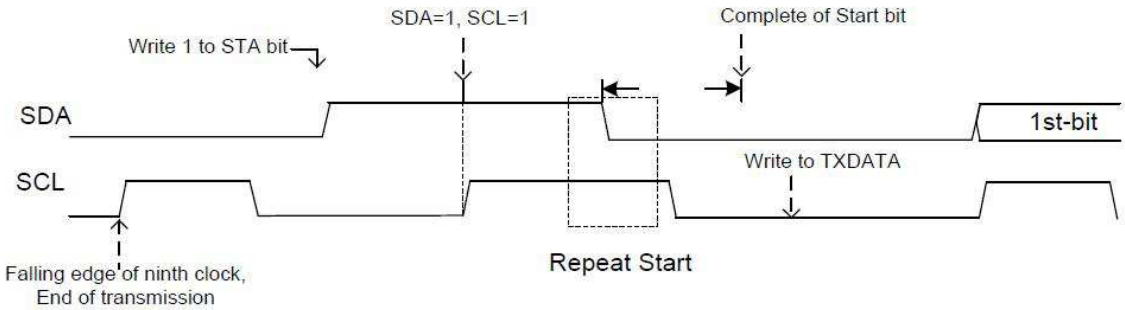


Figure 24–13: Repeat Start of I2C Protocol

©Levetop Electronics Co., Ltd.

24.5.4 Arbitration Procedure

If multiple devices simultaneously request the bus, the bus clock is determined by a synchronization procedure in which the low period equals the longest clock-low period among the devices and the high period equals the shortest. A data arbitration procedure determines the relative priority of competing devices. A device loses arbitration if it sends logic high while another sends logic low; it immediately switches to slave-receive mode and stops driving SDA.

A master that loses the arbitration can generate clock pulses until the end of the byte in which it loses the arbitration. It's possible that the winning master is trying to address it. The losing master must therefore switch over immediately to its slave mode.

In this case, the transition from master to slave mode does not generate a STOP condition. Meanwhile, hardware sets the ARBL bit of I2CSR to indicate loss of arbitration.

24.5.5 Clock Synchronization

Because wired-AND logic is used, a high-to-low transition on SCL affects all devices connected to the bus. Devices start counting their low period when the master drives SCL low. When a device clock goes low, it holds SCL low until the clock high state is reached. However, the low-to-high change in this device clock may not change the state of SCL if another device clock is still in its low period.

Therefore, the device with the longest low period holds the synchronized clock SCL low. Devices with shorter low periods enter a high wait state during this time (See **Figure 24–14)**. When all devices involved have counted off their low period, the synchronized clock SCL is released and pulled high.

There is then no difference between device clocks and the state of SCL, so all of the devices start counting their high periods. The first device to complete its high period pulls SCL low again.

24.5.6 Handshaking

The clock synchronization mechanism can be used as a handshake in data transfers. Slave devices can hold SCL low after completing one byte transfer (9 bits). In such a case, the clock mechanism halts the bus clock and forces the master clock into wait states until the slave releases SCL.

24.5.7 Clock Stretching

Slaves can use the clock synchronization mechanism to slow down the transfer bit rate. After the master has driven SCL low, the slave can drive SCL low for the required period and then release it. If the slave SCL low period is longer than the master SCL low period, the resulting SCL bus signal low period is stretched.

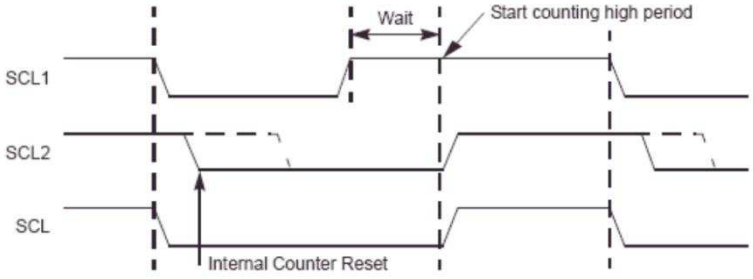


Figure 24–14: SCL Synchronization

©Levetop Electronics Co., Ltd.

24.5.8 High-Speed Mode Operation

The I2C module can transfer information at bit rates of up to 3.4 Mbits/s in HS-mode, yet it remains fully downward compatible with Fast or Standard-mode (F/S-mode) devices for bi-directional communication in a mixed-speed bus system. With the exception that arbitration and clock synchronization is not performed during the HS-mode transfer, the same serial bus protocol and data format is maintained as with the F/S-mode system.

Serial data transfer format in HS-mode meets the Standard-mode I2C-bus specification. HS-mode can only commence after the following conditions are met (all of which occur while in F/S-mode):

1. START condition (S)
2. 8-bit master code (00001XXX)
3. not-acknowledge bit (A)

(see **Figure 24–15** Data transfer format in HS mode) and (see **Figure 24–16** A Complete HS mode transfer) show this in more detail. The HS master code has two main functions:

1. It allows arbitration and synchronization between competing masters at F/S-mode speeds, resulting in one winning master.
2. It indicates the beginning of an HS-mode transfer.

HS-mode master codes are reserved 8-bit codes, which are not used for slave addressing or other purposes. Furthermore, as each master has its own unique master code, up to eight HS-mode masters can be present on the one I2C-bus system (although master code 0000 1000 should be reserved for test and diagnostic purposes). The master code for an I2C module is software programmable. Arbitration and clock synchronization only take place during the transmission of the master code and not-acknowledge bit (A), after which one winning master remains active. The master code indicates to other devices that an HS-mode transfer is to begin and the connected devices must meet the HS-mode specification. As no device is allowed to acknowledge the master code, the master code is followed by a not-acknowledge (A). After the not-acknowledge bit (A), and the SCL line has been pulled-up to a HIGH level, the active master switches to HS-mode and enables (at time tH, refer to (see **Figure 24–16** A Complete HS mode transfer)) the current-source pull-up circuit for the SCL signal. As other devices can delay the serial transfer before tH by stretching the LOW period of the SCL signal, the active master will enable its current-source pull-up circuit when all devices have released the SCL line and the SCL signal has reached a HIGH level, thus speeding up the last part of the rise time of the SCL signal. The active master then sends a repeated START condition (Sr)

Followed by a 7-bit slave address with a R/W bit address, and receives an acknowledge bit (A) from the selected slave.

After a repeated START condition and after each acknowledge bit (A) or not-acknowledge bit (A), the active master disables its current-source pull-up circuit. This enables other devices to delay the serial transfer by stretching the LOW period of the SCL signal. The active master re-enables its current-source pull-up circuit again when all devices have released and the SCL signal reaches a HIGH level, and so speeds up the last part of the SCL signal s rise time.

Data transfer continues in Hs-mode after the next repeated START (Sr), and only switches back to F/S-mode after a STOP condition (P). To reduce the overhead of the master code, it s possible that a master links a number of HS-mode transfers, separated by repeated START conditions (Sr).

©Levetop Electronics Co., Ltd.

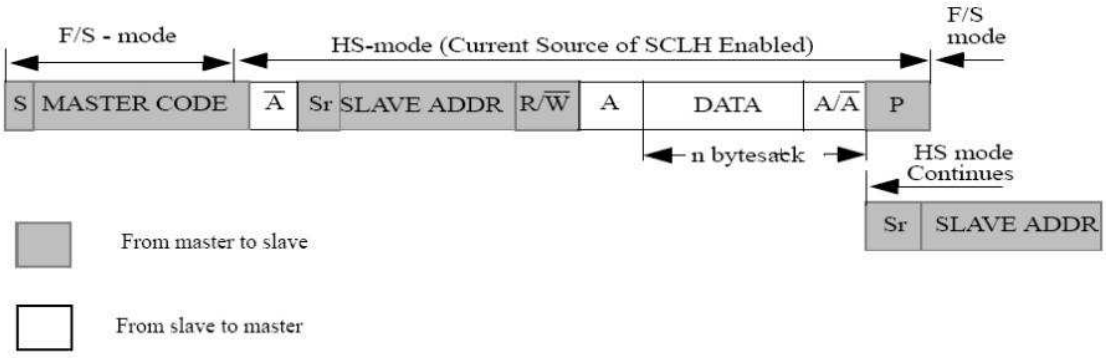


Figure 24–15: Data Transfer Format in HS Mode

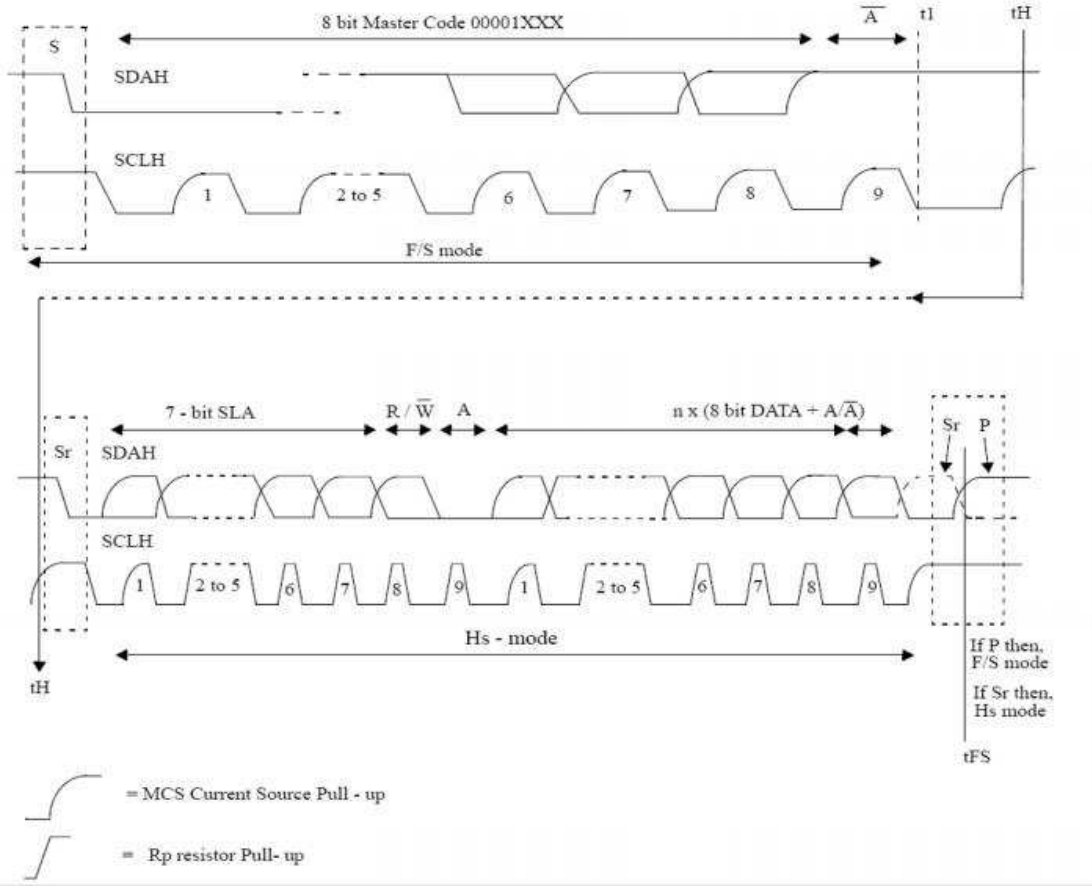


Figure 24–16: A Complete HS Mode Transfer

©Levetop Electronics Co., Ltd.

24.5.9 Software Transaction Flow Diagrams

1. Initialization.

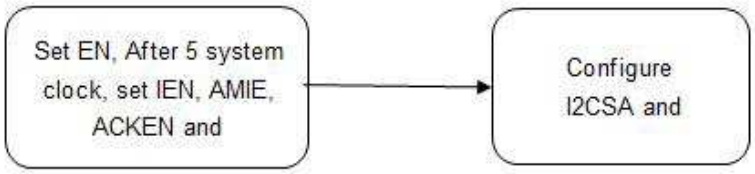


Figure 24–17: Slave Mode Initialization

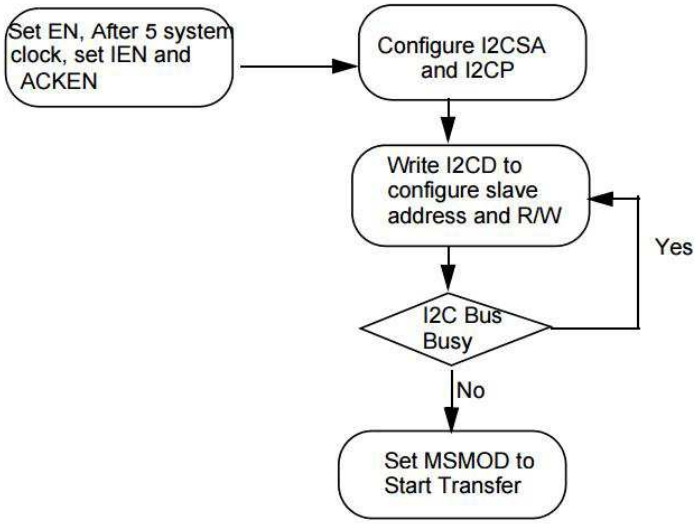


Figure 24–18: Master Mode Initialization

©Levetop Electronics Co., Ltd.

2. Interrupt transaction

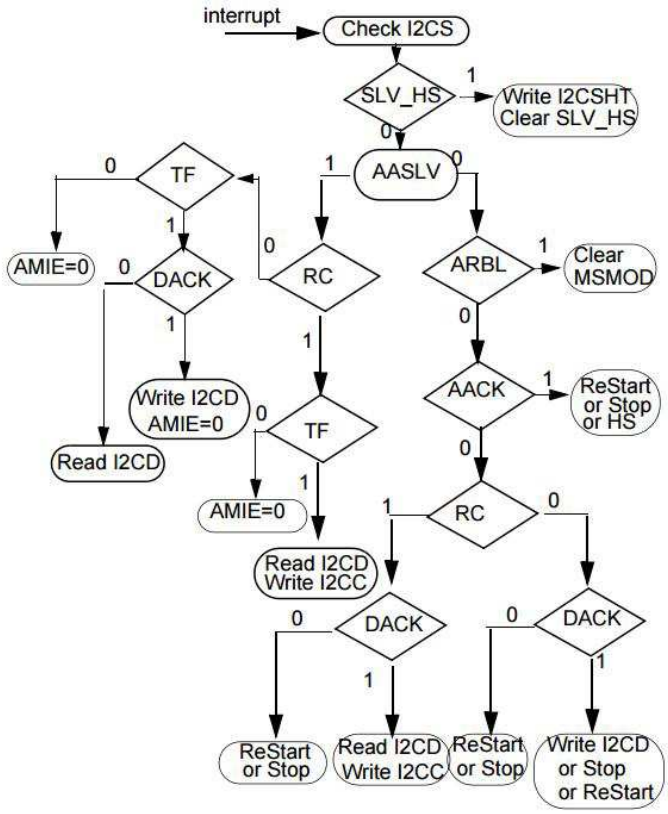


Figure 24–19: Interrupt Transaction

If there is an interruption comes from I2C, first check I2CS to confirm whether the I2C is in the master or slave mode.

First check SLV_HS status, if SLV_HS is set, then write I2CSHT to configure High-Speed Timing and write 1 to I2CSHIR to clear SLV_HS, otherwise:

In the slave mode, if RC has been set, means I2C is in the slave-receiver mode then check TF, if TF has also been set, means data (not address)has been received, then read I2CD and write I2CC to clear RC and TF to receive next byte data. If TF has not been set, means only slave address has been received, then write I2CC to clear RC and AMIE then receive next byte data. In the slave mode, if RC has not been set, then check TF, if TF has been set, then check DACK, if DACK has also been set, means I2C is in the slave-transmit mode, then write I2CD to clear TF, write I2CC to clear AMIE and transmit next byte data. If DACK has not been settled, then read I2CD for last received byte to clear TF.

In the slave mode, if both the TF and RC are not settled, then write I2CC to clear AMIE.

In the master mode, if ARBL has been set, means arbitration lost has been occurred, then write I2CC to clear MSMOD. If ARBL has not been settled then check AACK, if AACK has been settled, means address acknowledge error, then write I2CC to repeat start or stop.
In the master mode, if ARBL and AACK have not been settled, then check RC, if RC has been settled, means I2C is in the mast-receiver mode, then check DACK, if DACK has also been settled, then read I2CD to clear RC and write I2CC to choice whether acknowledge the data. If DACK has not been settled, write I2CC to repeat start or stop.

In the master mode, if ARBL and AACK have not been settled, then check RC, if RC has not been settled, means I2C is in the master-transmit mode, then check DACK, if DACK has been settled, write I2CD to clear TF and transmit next byte data or write I2CC to repeat start or stop. If DACK has not been settled, then write I2CC to repeat start or stop.

©Levetop Electronics Co., Ltd.

25. Pulse Width Modulator (PWM)

25.1 Introduction

There are 4 PWM-Timers enclosed. The 4 PWM-Timers has 2 Pre-scale, 2 clock divider, 4 clock selectors, 4 16-bit counters, 4 16-bit comparators, 2 Dead-Zone generators. Each can be used as a timer and issues interrupt independently.

Each two PWM-Timers share the same pre-scale. Clock divider provides each timer with 5 clock sources (1, 1/2, 1/4, 1/8, 1/16). The 16-bit counter in each timer receive clock signal from clock selector and can be used to handle one PWM period. The 16-bit comparator compares number in counter with threshold number in register loaded previously to generate PWM duty cycle. The clock signal from clock divider is called PWM clock. Dead-Zone generator utilize PWM clock as clock source. Once Dead-Zone generator is enabled, output of two PWM-Timers are blocked. Two output pin are all used as Dead-Zone generator output signal to control off-chip power device. The value of comparator is used for pulse width modulation. The counter control logic changes the output level when down-counter value matches the value of compare register.

Each PWM-Timer includes a capture channel. The Capture 0 and PWM 0 share a timer that included in PWM 0; and the Capture 1 and PWM 1 share another timer, and etc. Therefore user must setup the PWM-Timer before turn on Capture feature. After enabling capture feature, the capture always latched PWM-counter to CRLR when input channel has a rising transition and latched PWM-counter to CFLR when input channel has a falling transition. Capture channel 0 interrupt is programmable by setting CCR0[1] (Rising latch Interrupt enable) and CCR0[2] (Falling latch Interrupt enable) to decide the condition of interrupt occur. Capture channel1&2&3 has the same feature. Whenever Capture issues Interrupt 0/1/2/3, the PWM counter 0/1/2/3 will be reload at this moment. The maximal capture frequency should be decided by interrupt process time. If interrupt process time is T0, then the capture channel input signal should not change in T0, the maximal capture frequency is 1/T0.

There are only four interrupts from PWM to interrupt controller (INTC). PWM 0 and Capture 0 share the same interrupt; PWM1 and Capture 1 share the same interrupt and so on. Therefore, PWM function and Capture function in the same channel cannot be used at the same time.

25.2 Features

The Pulse Width Modulator includes these distinctive features:

- Programmable period
- Programmable duty cycle \
- Two Dead-Zone generator
- Capture function
- Pins can be configured as general-purpose I/O

©Levetop Electronics Co., Ltd.

25.3 Block Diagram

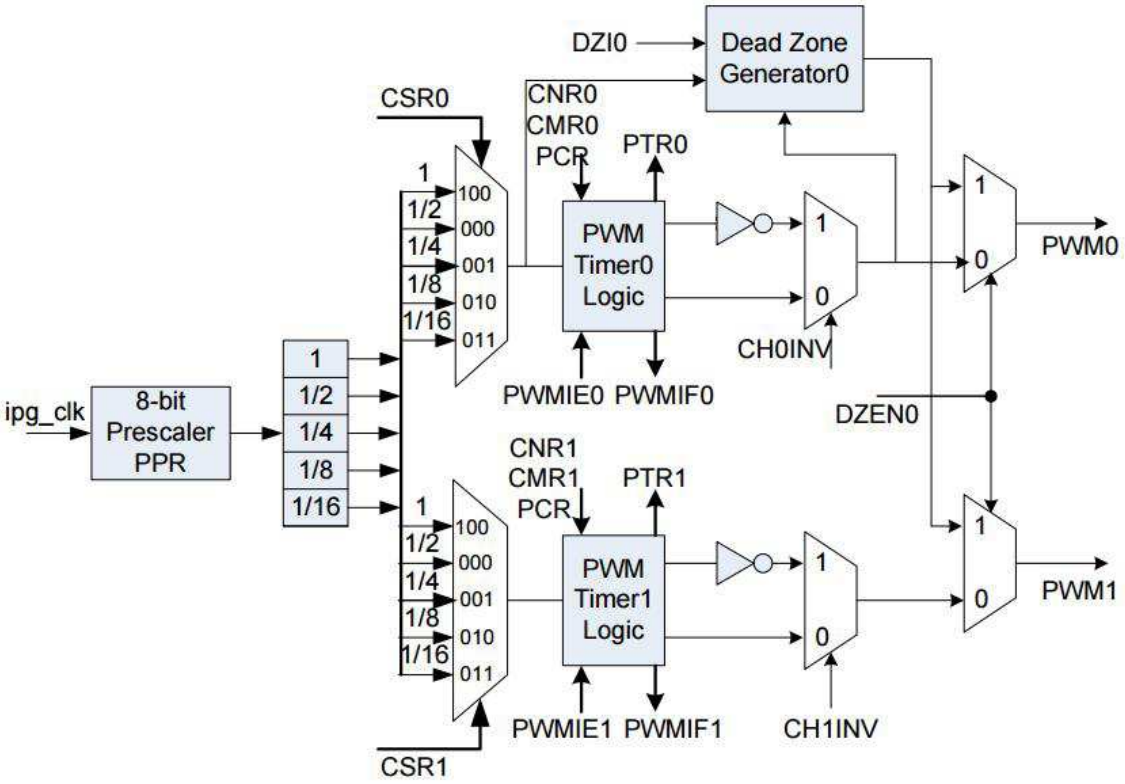


Figure 25–1: PWM Block Diagram

25.4 Signal Description

Table 25–1: PWM Signal Description

Name	I/O	Width	Reset State	Description
PWM0	I/O	1	0	PWM0 pin
PWM1	I/O	1	0	PWM1 pin
PWM2	I/O	1	0	PWM2 pin
PWM3	I/O	1	0	PWM3 pin

PWMx are used as a general-purpose input/output, and also used as the PWM send output or capture input.

In default state, it is used as general-purpose input port.

©Levetop Electronics Co., Ltd.

25.5 Memory Map and Registers

(Base: 0x400d_0000, 0x400e_0000)
This subsection describes the memory map and register structure.

25.5.1 Memory Map

Table 25–2: Module Memory Map

Address	Bit[15:8]	Bit[7:0]	Access(1)
0x0000	PWM Pre-scale Register (PPR)		S/U
0x0004	PWM Clock Select Register (PCSR)		S/U
0x0008	PWM Control Register (PCR)		S/U
0x000C	PWM Counter Register0 (PCNR0)		S/U
0x0010	PWM Comparator Register0 (PCMR0)		S/U
0x0014	PWM Timer Register0 (PTR0)		S/U
0x0018	PWM Counter Register1 (PCNR1)		S/U
0x001C	PWM Comparator Register1 (PCMR1)		S/U
0x0020	PWM Timer Register1 (PTR1)		S/U
0x0024	PWM Counter Register2 (PCNR2)		S/U
0x0028	PWM Comparator Register2 (PCMR2)		S/U
0x002C	PWM Timer Register2 (PTR2)		S/U
0x0030	PWM Counter Register3 (PCNR3)		S/U
0x0034	PWM Comparator Register3 (PCMR3)		S/U
0x0038	PWM Timer Register3 (PTR3)		S/U
0x003C	PWM Interrupt Enable Register (PIER)		S/U
0x0040	PWM Interrupt Flag Register (PIFR)		S/U
0x0044	PWM Capture Control Register0 (PCCR0)		S/U
0x0048	PWM Capture Control Register1 (PCCR1)		S/U
0x004C	PWM Capture Rising Latch Register0 (PCRLR0)		S/U
0x0050	PWM Capture Falling Latch Register0 (PCFLR0)		S/U
0x0054	PWM Capture Rising Latch Register1 (PCRLR1)		S/U
0x0058	PWM Capture Falling Latch Register1 (PCFLR1)		S/U
0x005C	PWM Capture Rising Latch Register2 (PCRLR2)		S/U
0x0060	PWM Capture Falling Latch Register2 (PCFLR2)		S/U
0x0064	PWM Capture Rising Latch Register3 (PCRLR3)		S/U
0x0068	PWM Capture Falling Latch Register3 (PCFLR3)		S/U
0x006C	PWM Port Control Register (PPCR)		S/U

©Levetop Electronics Co., Ltd.

25.5.2 Registers

25.5.2.1 PWM Pre-Scale Register (PPR)

The register (PPR) is used to set prescaler and set dead zone length.

Address: 0x0000 and 0x0003

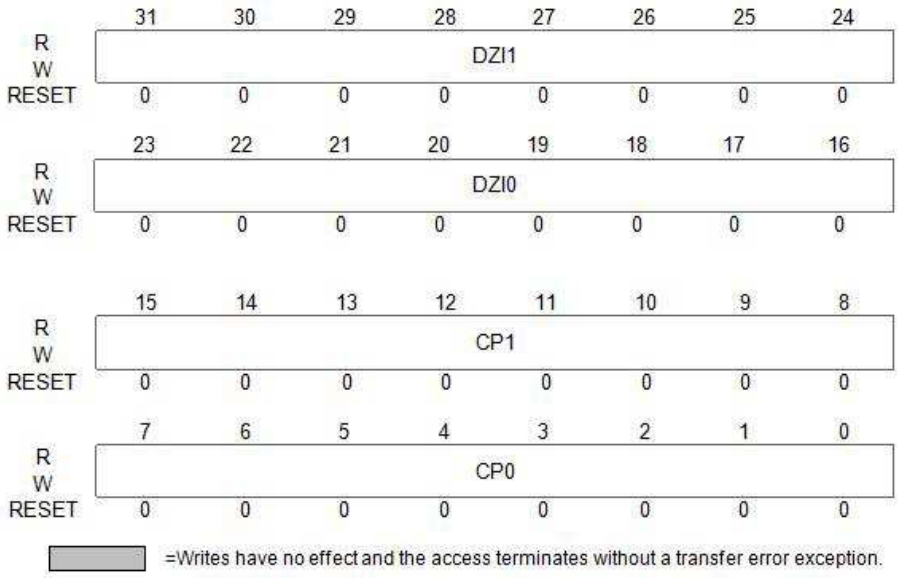


Figure 25–2: PWM Pre-Scale Register (PPR)

DZI1[7:0] — Dead zone interval register 1 for PWM2 and PWM3
These 8-bit determine dead zone length. The 1 unit time of dead zone length is received from clock selector 1.

DZI0[7:0] — Dead zone interval register 0 for PWM0 and PWM1
These 8-bit determine dead zone length. The 1 unit time of dead zone length is received from clock selector 0.

CP1[7:0] — Clock pre-scale 1 for PWM Timer 2 & 3
Clock input is divided by (CP1 + 1) before it is fed to the counter of PWM Timer 2 & 3.If CP1=0, then the pre-scale 1 output clock will be stopped.

CP0[7:0] — Clock pre-scale 0 for PWM Timer 0 & 1
Clock input is divided by (CP0 + 1) before it is fed to the counter of PWM Timer 0 & 1.If CP0=0, then the pre-scale 0 output clock will be stopped.

©Levetop Electronics Co., Ltd.

25.5.2.2 PWM Clock Select Register (PCSR)

Clock divider provides each timer with 5 clock sources (1, 1/2, 1/4, 1/8, 1/16). Each timer receives its own clock signal from clock divider which receives clock from 8-bit pre-scale.

Address: 0x0004 and 0x0007

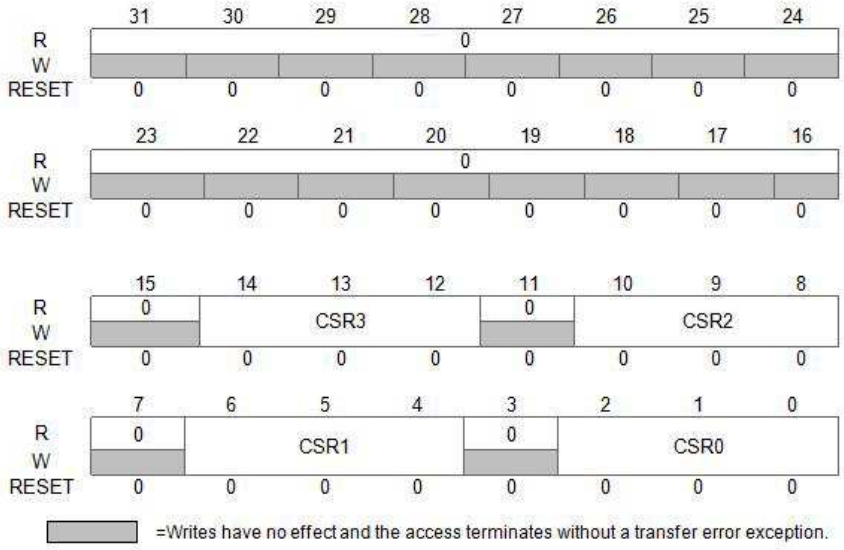


Figure 25–3: PWM Clock Select Register (PCSR)

CSR3[2:0] — Timer 3 Clock Source Selection
Select clock input for timer 3.

Table 25–3: CSR3 Description

CSR3[2:0]	Input Clock Divided by
100~111	1
011	16
010	8
001	4
000	2

CSR2[2:0] — Timer 2 Clock Source Selection

Select clock input for timer 2.Same as CSR3.

CSR1[2:0] — Timer 1 Clock Source Selection

Select clock input for timer 1.Same as CSR3.

CSR0[2:0] — Timer 0 Clock Source Selection

Select clock input for timer 0.Same as CSR3.

©Levetop Electronics Co., Ltd.

25.5.2.3 PWM Control Register (PCR)

This register is the PWM control register.

Address: 0x0008 and 0x000b

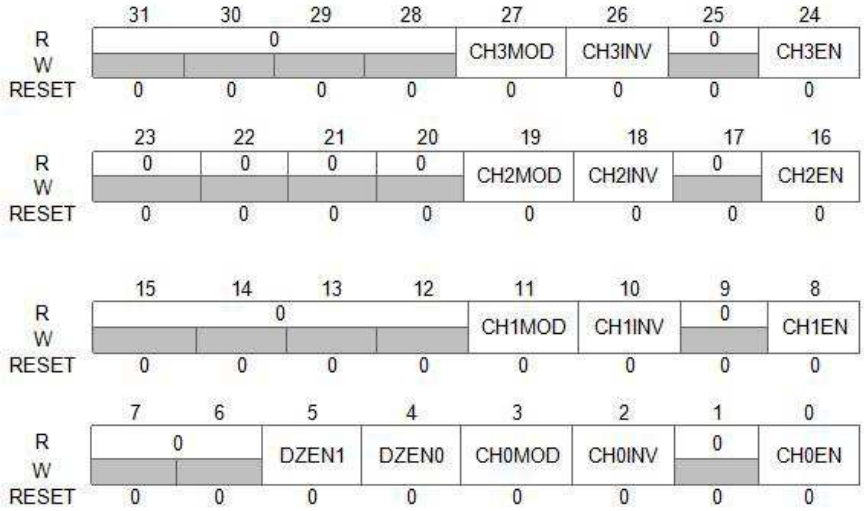


Figure 25–4: PWM Control Register (PCR)

CH3MOD — Timer 3 Auto-load/One-Shot Mode
1 = Auto-load Mode
0 = One-Shot Mode

Note:
If there is a rising or falling transition at this bit, it will cause CNR3 and CMR3 be clear

CH3INV — Timer 3 Inverter ON/OFF
1 = Inverter ON
0 = Inverter OFF

CH3EN — Timer 3 Enable/Disable
1 = Enable
0 = Disable

CH2MOD — Timer 2 Auto-load/One-Shot Mode
1 = Auto-load Mode
0 = One-Shot Mode

Note:
If there is a rising or falling transition at this bit, it will cause CNR2 and CMR2 be clear

CH2INV — Timer 2 Inverter ON/OFF
1 = Inverter ON
0 = Inverter OFF

CH2EN — Timer 2 Enable/Disable
1 = Enable
0 = Disable

CH1MOD — Timer 1 Auto-load/One-Shot Mode
1 = Auto-load Mode
0 = One-Shot Mode

©Levetop Electronics Co., Ltd.

Note:
If there is a rising or falling transition at this bit, it will cause CNR1 and CMR1 be clear.

CH1INV — Timer 1 Inverter ON/OFF
1 = Inverter ON
0 = Inverter OFF

CH1EN — Timer 1 Enable/Disable
1 = Enable
0 = Disable

CH0MOD — Timer 0 Auto-load/One-Shot Mode
1 = Auto-load Mode
0 = One-Shot Mode

Note:
If there is a rising or falling transition at this bit, it will cause CNR0 and CMR0 be clear.

CH0INV — Timer 0 Inverter ON/OFF
1 = Inverter ON
0 = Inverter OFF

CH0EN — Timer 0 Enable/Disable
1 = Enable
0 = Disable

DZEN1 — Dead-Zone 1 Generator Enable/Disable
1 = Enable
0 = Disable

Note:
When DZEN1 is enable, CH3EN should be set disable. Because channel3 and channel2 outputs both decided by channel2.

DZEN0— Dead-Zone 0 Generator Enable/Disable
1 = Enable
0 = Disable

Note:
When DZEN0 is enable, CH1EN should be set disable. Because channel1 and channel0 outputs both decided by channel0.

©Levetop Electronics Co., Ltd.

25.5.2.4 PWM Counter Register (PCNR0/1/2/3)

These registers control the period of the PWM by defining the number of the count_pulse in the period.

Address: 0x000c and 0x000f

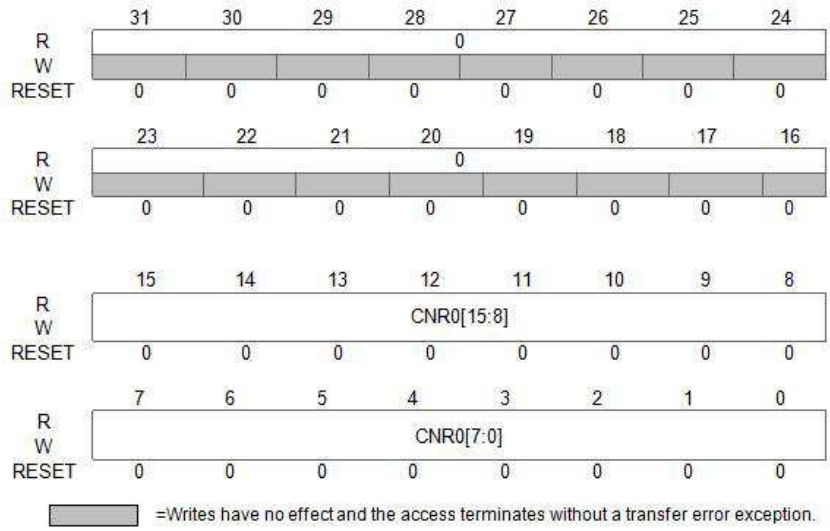


Figure 25–5: PWM Counter Register (PCNR0)

Address: 0x0018 and 0x001b

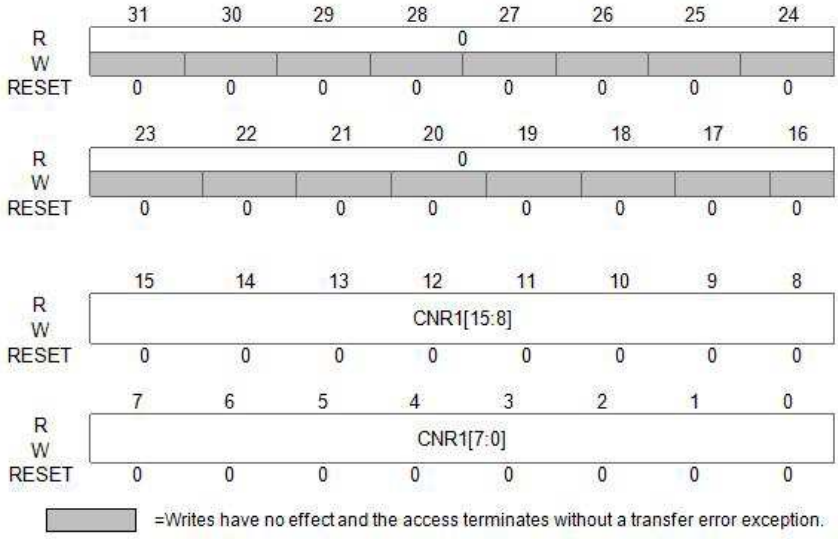


Figure 25–6: PWM Counter Register (PCNR1)

©Levetop Electronics Co., Ltd.

Address: 0x0024 and 0x0027

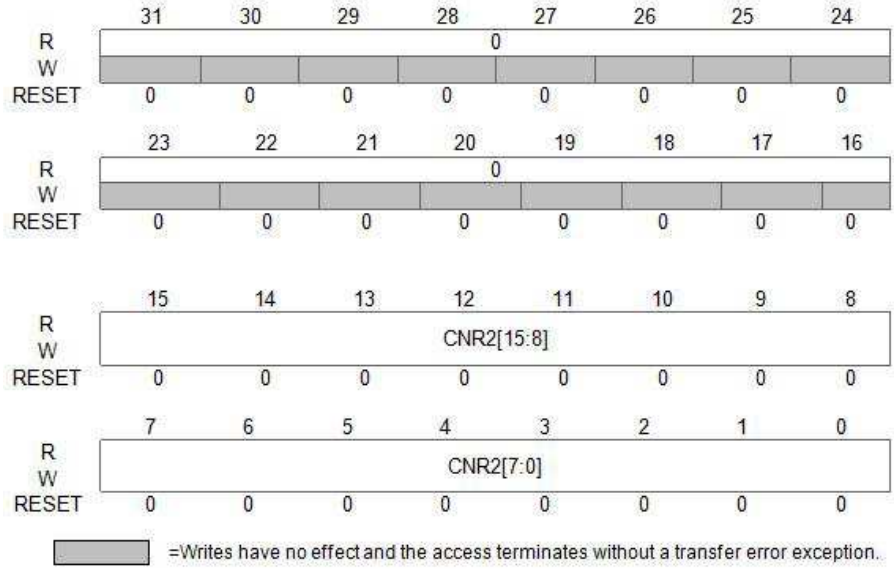


Figure 25–7: PWM Counter Register (PCNR2)

Address: 0x0030 and 0x0033

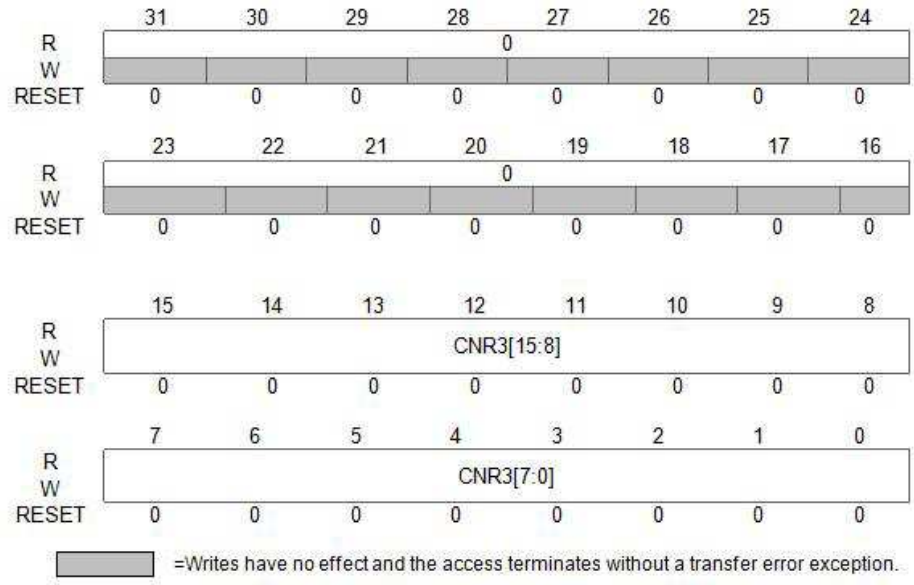


Figure 25–8: PWM Counter Register (PCNR3)

CNR[15:0] — PWM Counter/Timer Loaded Value
Inserted data range: 65535~0 (Unit : 1 PWM clock cycle)

Note 1:
One PWM cycle width = CNR + 1. If CNR equal zero, PWM counter/timer will be stopped.

Note 2:
Programmer can feel free to write a data to CNR at any time, and it will take effect in next PWM Counter cycle.

©Levetop Electronics Co., Ltd.

25.5.2.5 PWM Comparator Register (PCMR0/1/2/3)

These registers define the width of the pulse. When the counter matches the value in this register, the output is reset for the duration of the period.

Address: 0x0010 and 0x0013

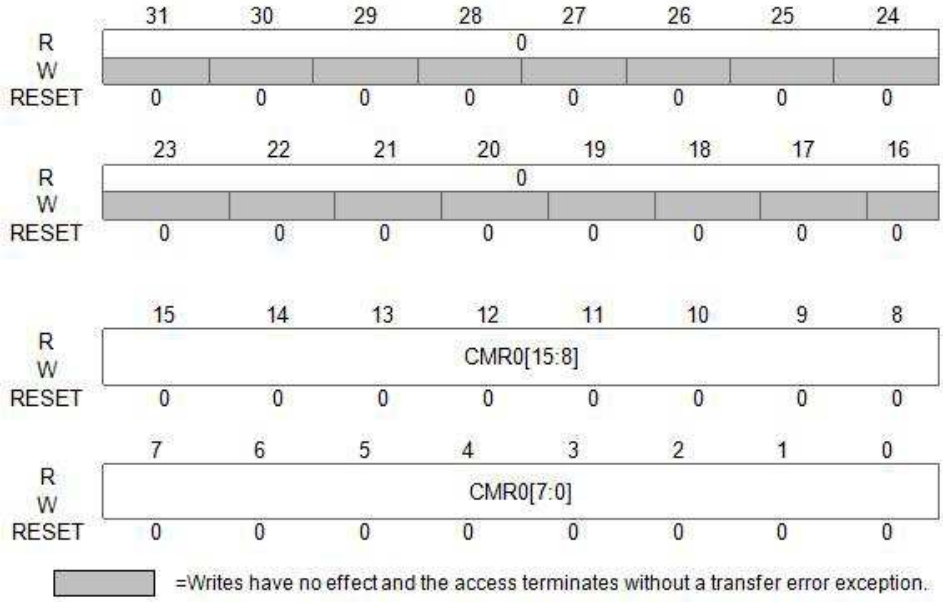


Figure 25–9: PWM Comparator Register (PCMR0)

Address: 0x001c and 0x001f

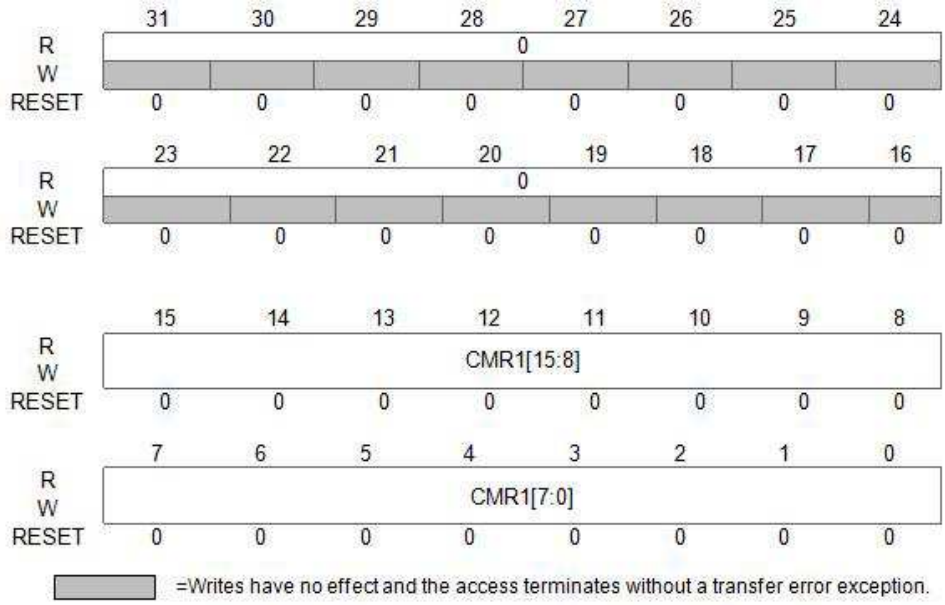


Figure 25–10: PWM Comparator Register (PCMR1)

©Levetop Electronics Co., Ltd.

Address: 0x0028 and 0x002b

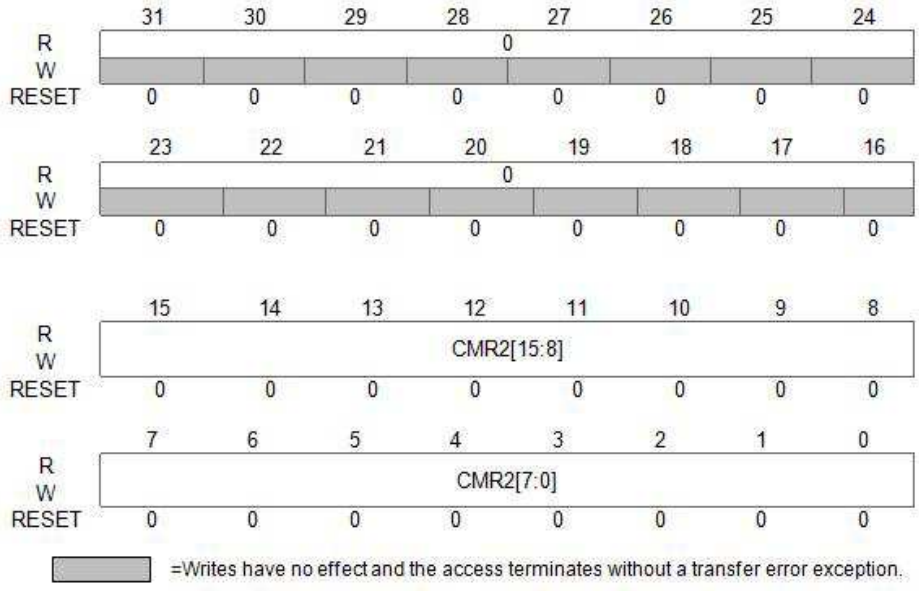


Figure 25–11: PWM Comparator Register (PCMR2)

Address: 0x0034 and 0x0037

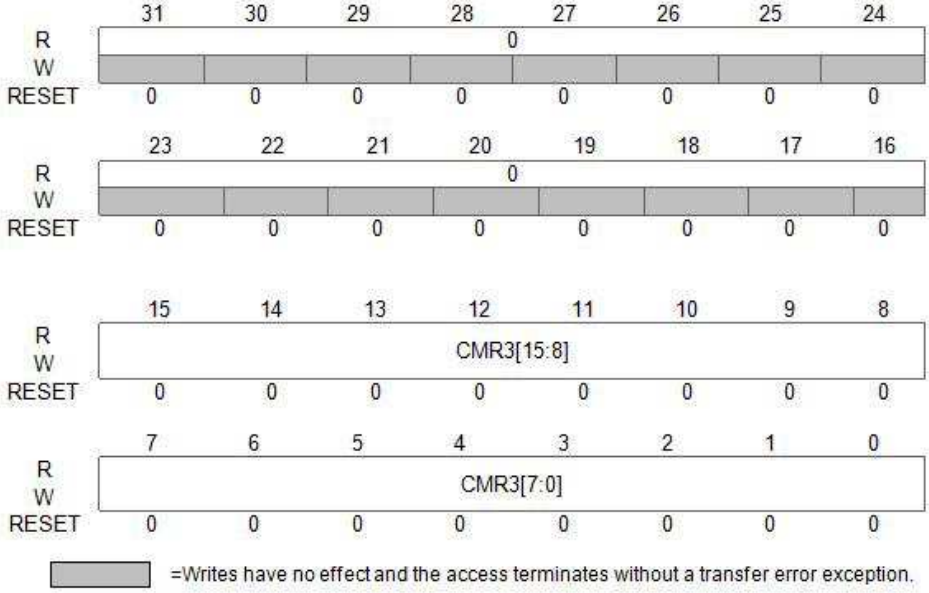


Figure 25–12: PWM Comparator Register (PCMR3)

CMR[15:0] — PWM Comparator Register
Inserted data range: 65535~0 (Unit : 1 PWM clock cycle)

CMR are used to determine PWM output duty ratio.

Assumption: PWM output initial: high
CMR >= CNR: PWM output is always high
CMR < CNR: PWM output high = (CMR + 1) unit
CMR = 0: PWM output high = 1 unit

©Levetop Electronics Co., Ltd.

Note 1:
PWM duty = CMR + 1. If CMR equal zero, PWM duty = 1

Note 2:
Programmer can feel free to write a data to CMR at any time, and it will take effect in next PWM Counter cycle.

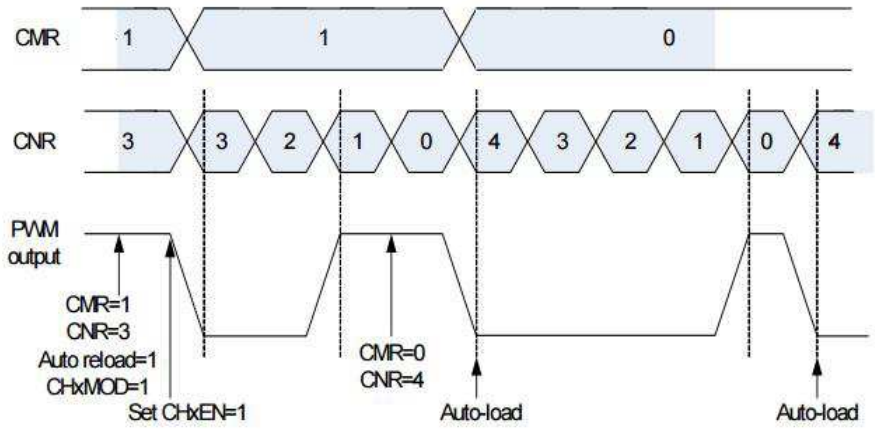


Figure 25–13: PWM Output

Modulate PWM Controller output duty ratio (CNR=150)

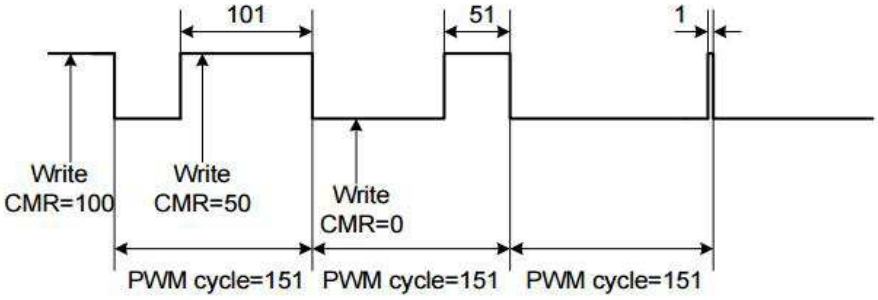


Figure 25–14: PWM Duty Ratio

©Levetop Electronics Co., Ltd.

25.5.2.6 PWM Timer Register (PTR0/1/2/3)

The read-only PWM timer registers hold the current count value. It can be read at any time without disturbing the counter.

Address: 0x0014 and 0x0017

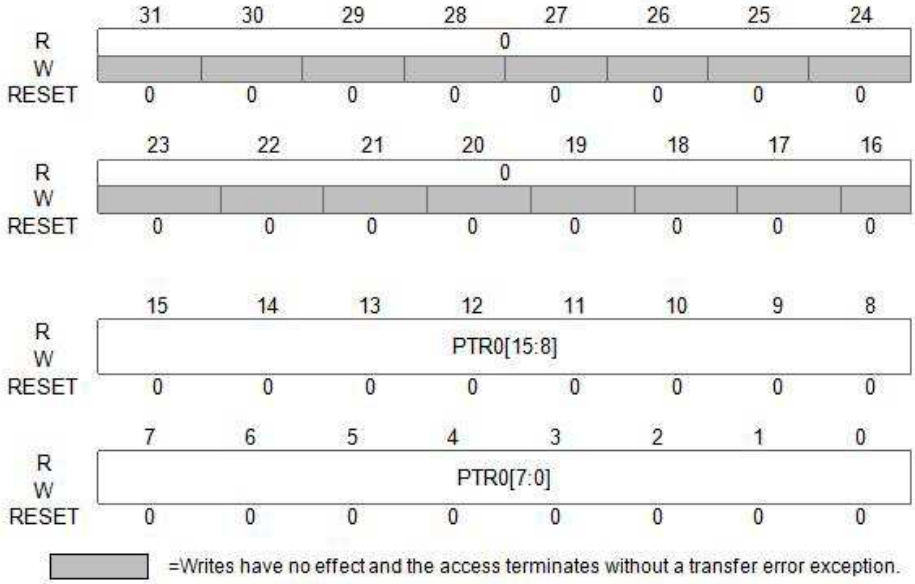


Figure 25–15: PWM Timer Register (PTR0)

Address: 0x0020 and 0x0023

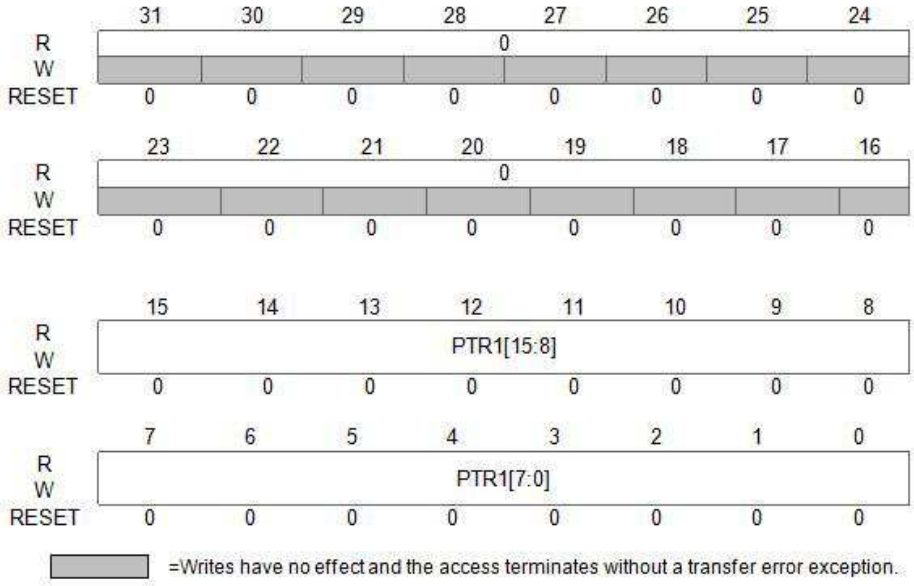


Figure 25–16: PWM Timer Register (PTR1)

©Levetop Electronics Co., Ltd.

Address: 0x002c and 0x002f

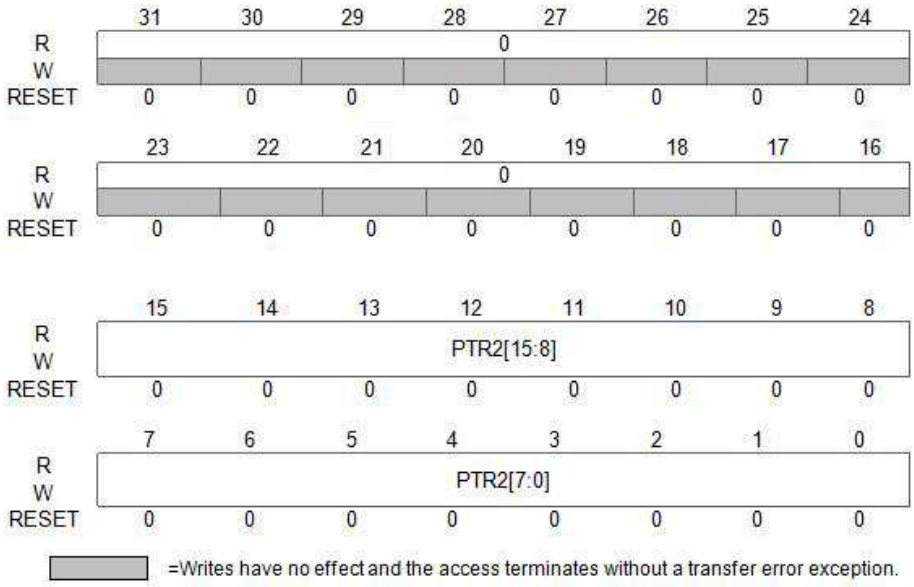


Figure 25–17: PWM Timer Register (PTR2)

Address: 0x0038 and 0x003b

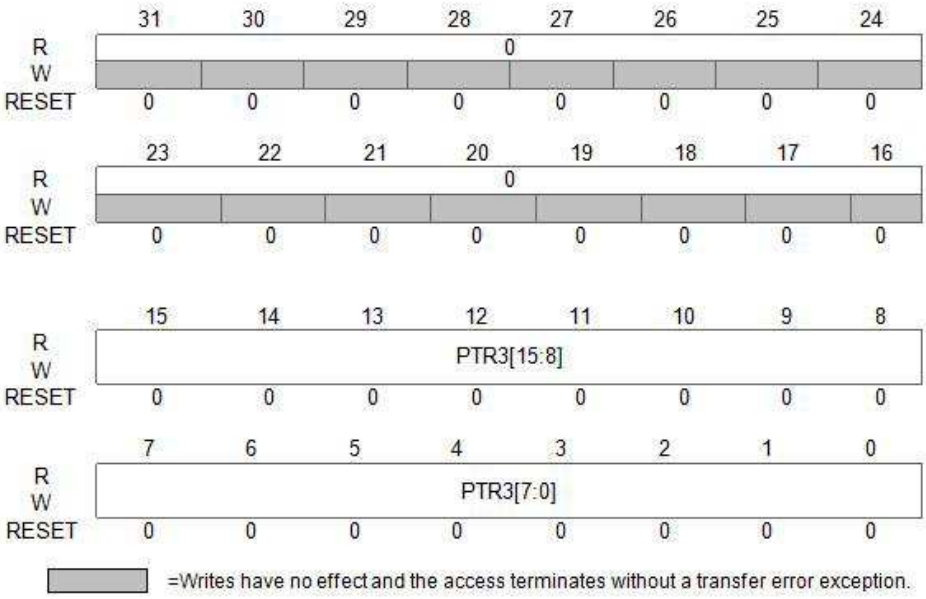


Figure 25–18: PWM Timer Register (PTR3)

PTR[15:0] — PWM Timer value

The read-only PTR bits holds the current count value. User can monitor PTR to know current value in 16-bit down counter.

©Levetop Electronics Co., Ltd.

25.5.2.7 PWM Interrupt Enable Register (PIER)

This register is used to enable PWM timer interrupt.

Address: 0x003c and 0x003f

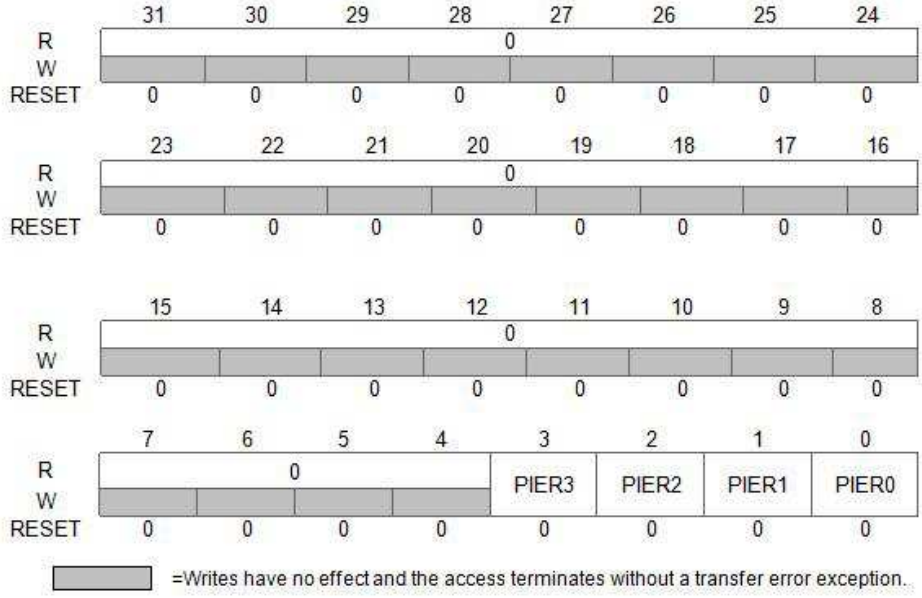


Figure 25–19: PWM Interrupt Enable Register (PIER)

PIER3 — PWM Timer 3 Interrupt Enable
1 = Enable
0 = Disable

PIER2 — PWM Timer 2 Interrupt Enable
1 = Enable
0 = Disable

PIER1 — PWM Timer 1 Interrupt Enable
1 = Enable
0 = Disable

PIER0 — PWM Timer 0 Interrupt Enable
1 = Enable
0 = Disable

©Levetop Electronics Co., Ltd.

25.5.2.8 PWM Interrupt Flag Register (PIFR)

This register is used to indicate PWM timer interrupt flag.

Address: 0x0040 and 0x0043

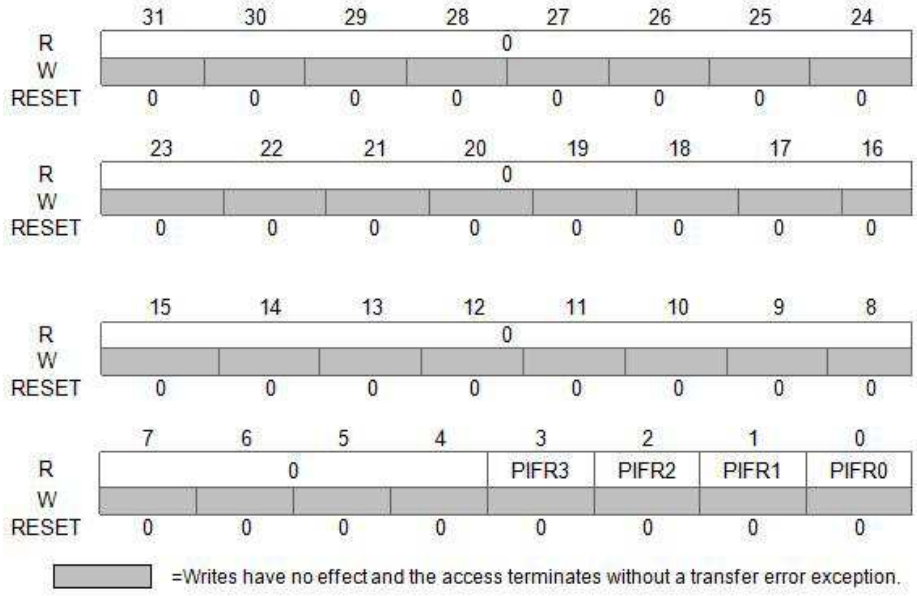


Figure 25–20: PWM Interrupt Flag Register (PIFR)

PIFR3 — PWM Timer 3 Interrupt Flag.
When PWM timer3 count to 0, and PIER3 =1, PIFR3 will be set to 1.Write 1 to this bit will clear PIFR3.
1 = Interrupt Flag on
0 = Interrupt Flag off

PIFR2 — PWM Timer 2 Interrupt Flag.
When PWM timer2 count to 0, and PIER2 =1, PIFR2 will be set to 1.Write 1 to this bit will clear PIFR2.
1 = Interrupt Flag on
0 = Interrupt Flag off

PIFR1 — PWM Timer 1 Interrupt Flag.
When PWM timer1 count to 0, and PIER1 =1,PIFR1 will be set to 1.Write 1 to this bit will clear PIFR1.
1 = Interrupt Flag on
0 = Interrupt Flag off

PIFR0 — PWM Timer 0 Interrupt Flag.
When PWM timer0 count to 0, and PIER0 =1, PIFR0 will be set to 1.Write 1 to this bit will clear PIFR0.
1 = Interrupt Flag on
0 = Interrupt Flag off

©Levetop Electronics Co., Ltd.

25.5.2.9 PWM Capture Control Register (PCCR0/1)

These registers are used to control capture function.

Address: 0x0044 and 0x0047

	31	30	29	28	27	26	25	24
R	0							
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	CFLRD1	CRLRD1	0	CAPIF1	CAPCH1EN	FL_IE1	RL_IE1	INV1
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0							
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	CFLRD0	CRLRD0	0	CAPIF0	CAPCH0EN	FL_IE0	RL_IE0	INV0
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 25–21: PWM Capture Control Register (PCCR0)

Address: 0x0048 and 0x004b

	31	30	29	28	27	26	25	24
R	0							
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	CFLRD3	CRLRD3	0	CAPIF3	CAPCH3EN	FL_IE3	RL_IE3	INV3
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0							
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	CFLRD2	CRLRD2	0	CAPIF2	CAPCH2EN	FL_IE2	RL_IE2	INV2
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 25–22: PWM Capture Control Register (PCCR1)

©Levetop Electronics Co., Ltd.

CFLRDx— Capture Falling Latch Register load flag
1 = When input channel x has a falling transition, CFLRx was updated and this bit was “1”.
0 = When input channel x doesn’t have a falling transition.
Write 1 clear this bit.

CRLRDx— Capture Rising Latch Register load flag
1 = When input channel x has a rising transition, CRLRx was updated and this bit was “1”.
0 = When input channel x doesn’t have a rising transition.
Write 1 clear this bit.

CAPIFx— Capture Channel x interrupt flag
1 = When input channel x has a falling transition, and FL&IEx bit enable, this interrupt flag set. When input channel x has a rising transition, and RL&IEx bit enable, this interrupt flag also set.
0 = Capture channel x interrupt flag not set.
Write 1 clear this bit.

CAPCHxEN— Capture Channel x Enable/Disable
1 = Enable
0 = Disable
When Enable, Capture latched the PMW-counter and saved to CRLR (Rising latch) and CFLR (Falling latch). When Disable, Capture do not update CRLR and CFLR, and disable Channel x Interrupt.

FL_IEx—Channel x Falling Interrupt Enable ON/OFF
1 = Enable
0 = Disable
When enable, if Capture detects Channel x has falling transition, Capture issues an Interrupt.

RL_IEx—Channel x Rising Interrupt Enable ON/OFF
1 = Enable
0 = Disable
When enable, if Capture detects Channel x has rising transition, Capture issues an Interrupt.

INVx—Channel x Inverter ON/OFF
1 = Inverter ON
0 = Inverter OFF

©Levetop Electronics Co., Ltd.

25.5.2.10 PWM Capture Rising Latch Register (PCRLR0/1/2/3)

These registers are used to latch the PWM counter when capture rising transition.

Address: 0x004c and 0x004f

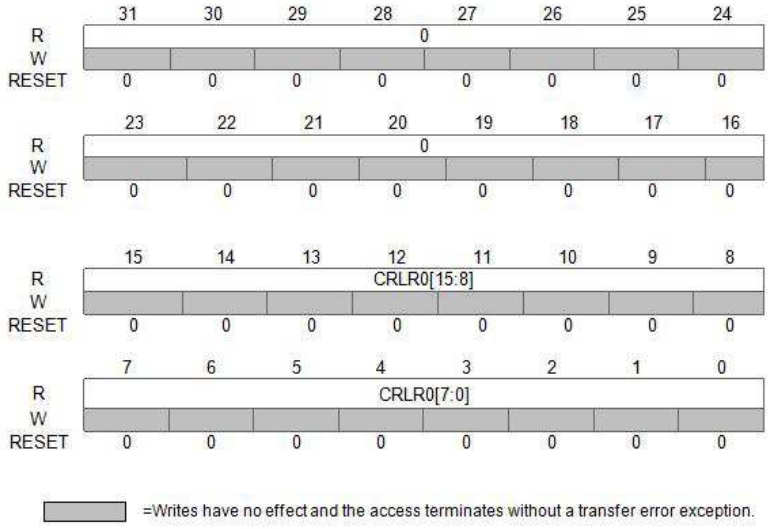


Figure 25–23: PWM Capture Rising Latch Register (PCRLR0)

Address: 0x0054 and 0x0057

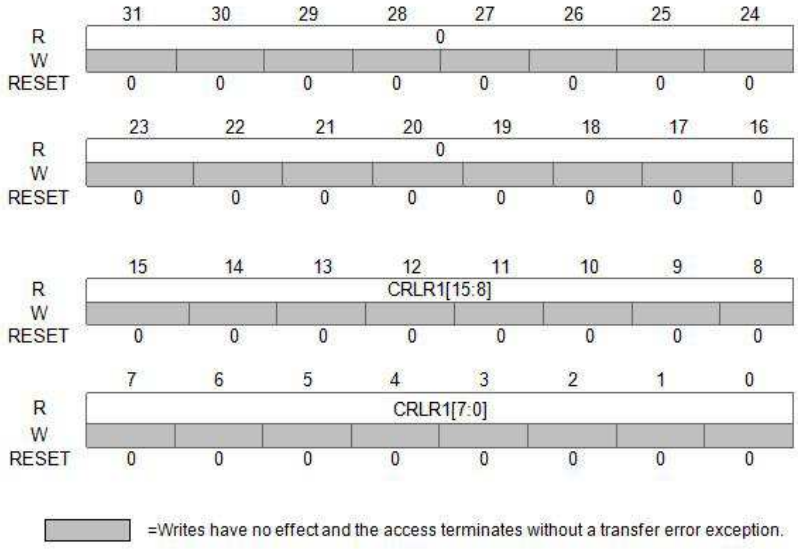


Figure 25–24: PWM Capture Rising Latch Register (PCRLR1)

©Levetop Electronics Co., Ltd.

Address: 0x005c and 0x005f

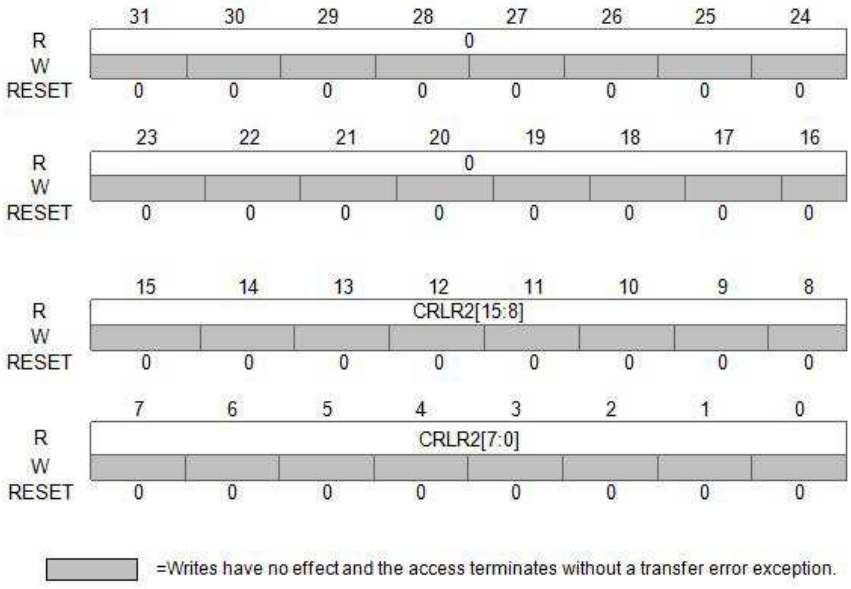


Figure 25–25: PWM Capture Rising Latch Register (PCRLR2)

Address: 0x0064 and 0x0067

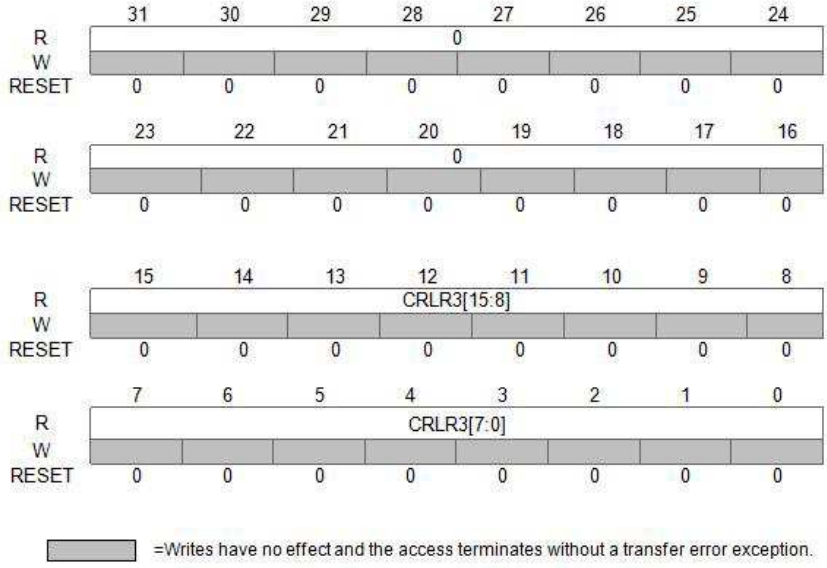


Figure 25–26: PWM Capture Rising Latch Register (PCRLR3)

CRLRx[15:0] — Capture Rising Latch Registerx
Latch the PWM counter when Channel x has rising transition.

©Levetop Electronics Co., Ltd.

25.5.2.11 PWM Capture Falling Latch Register (PCFLR0/1/2/3)

These registers are used to latch the PWM counter when capture falling transition.

Address: 0x0050 and 0x0053

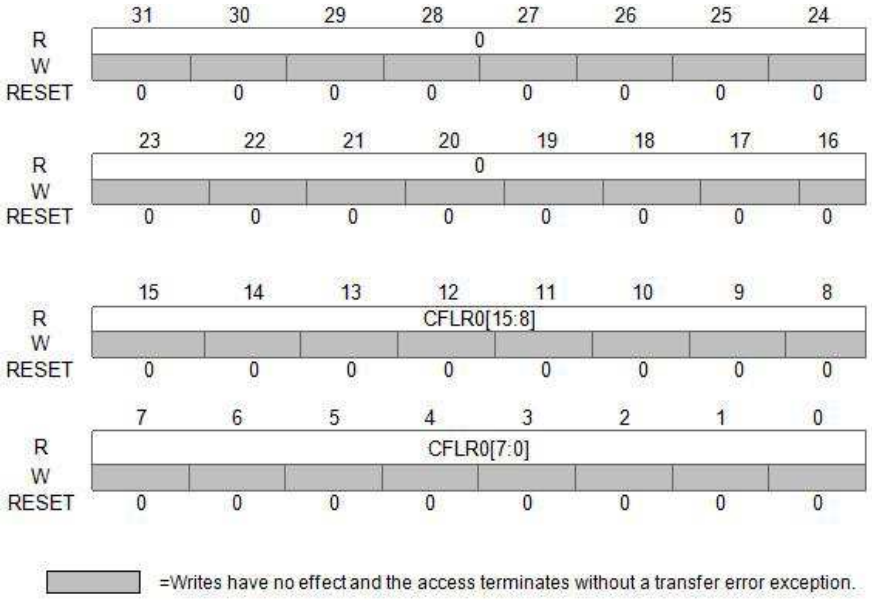


Figure 25–27: PWM Capture Falling Latch Register (PCFLR0)

Address: 0x0058 and 0x005b

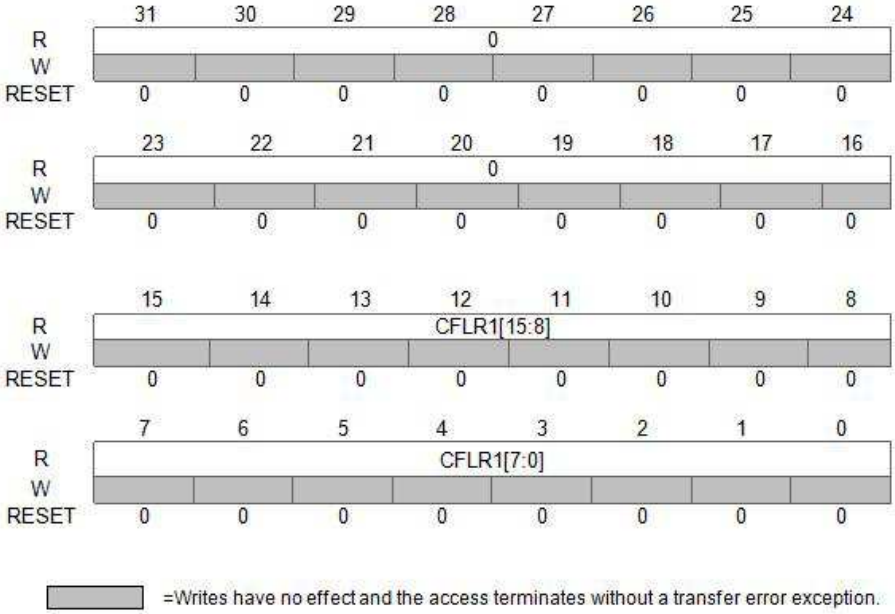


Figure 25–28: PWM Capture Falling Latch Register (PCFLR1)

©Levetop Electronics Co., Ltd.

Address: 0x0060 and 0x0063

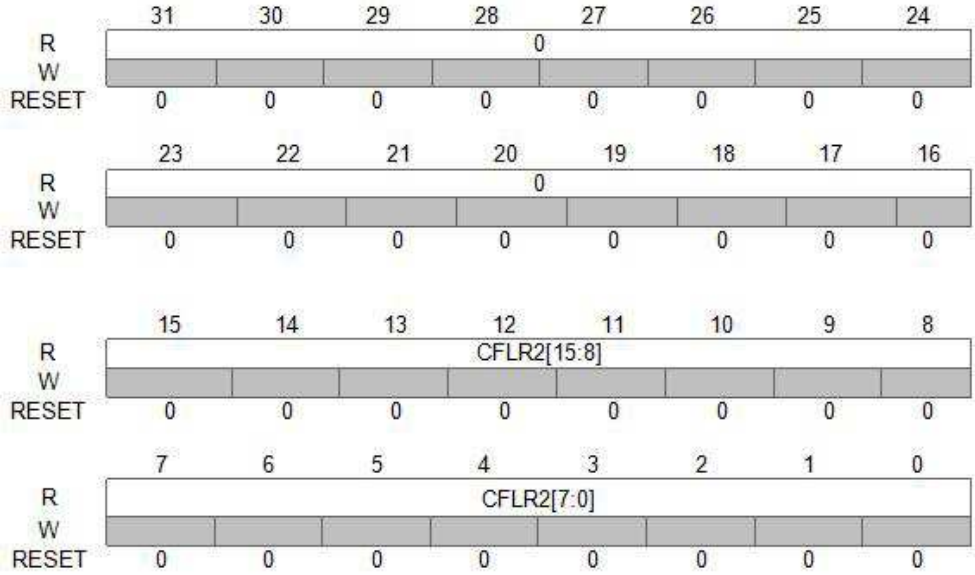


Figure 25–29: PWM Capture Falling Latch Register (PCFLR2)

Address: 0x0068 and 0x006b

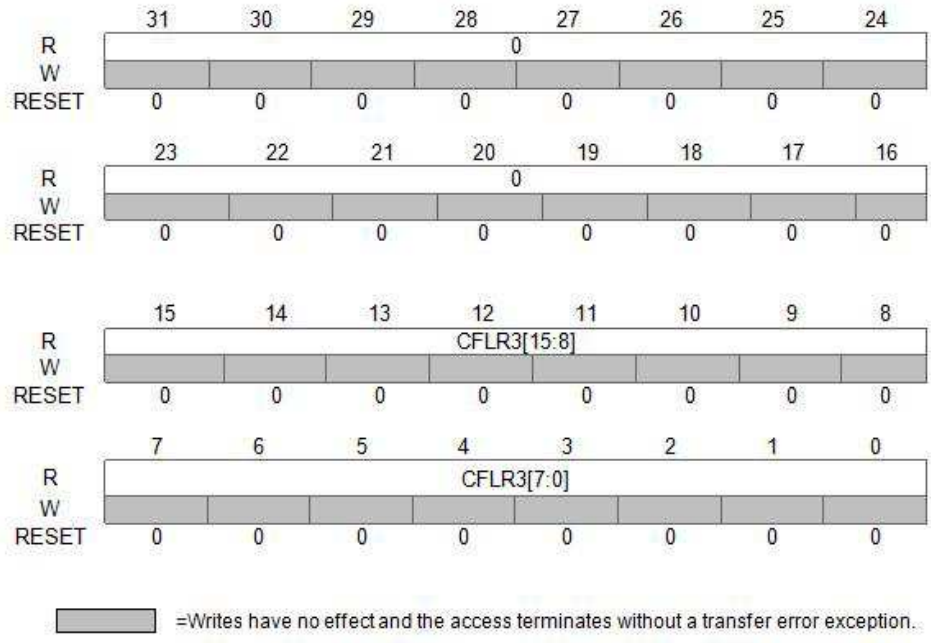


Figure 25–30: PWM Capture Falling Latch Register (PCFLR3)

CFLRx[15:0] — Capture Falling Latch Registerx
Latch the PWM counter when Channel x has falling transition.

©Levetop Electronics Co., Ltd.

25.5.2.12 PWM Port Control Register (PPCR)

The register (PPCR) is used to control PWMx pin direction and pin status.

Address: 0x006c and 0x006f

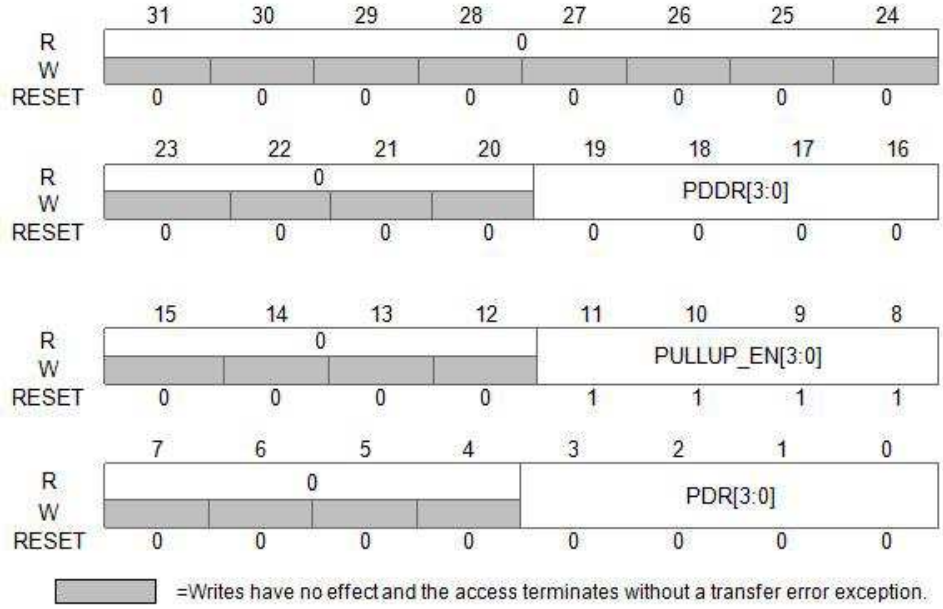


Figure 25–31: PWM Port Control Register (PPCR)

PDDR[3:0] — Port Data Direction Register

The PDDR[3:0] bits control the direction of PWM Pins. Reset clear PDDR[3:0].

1 = Corresponding pin configured as output

0 = Corresponding pin configured as input

PULLUP_EN[3:0]—Port Pull up Enable

The PULLUP_EN[3:0] bits control the pull-up characteristic of PWM Pins.

1 = Enable pull up

0 = Disable pull up

PDR[3:0] — Port Data Register

Data written to PDR[3:0] drives pins only when they are configured as general-purpose outputs. Reading an input (PDDR bit clear) returns the pin level.

©Levetop Electronics Co., Ltd.

25.6 Functional Descriptions

This subsection describes the PWM functional operation.

25.6.1 PWM Double Buffering and Automatic Reload

PWM-Timers have a double buffering function, enabling the reload value changed for next timer operation without stopping current timer operation. Although new timer value is set, current timer operation still operate successfully. The counter value can be written into CNR0~3 and current counter value can be read from PTR0~3. The auto-reload operation will copy from CNR0~3 to down-counter when down-counter reaches zero. If CNR0~3 are set as zero, counter will be halt when counter count to zero. If auto-reload bit is set as zero, counter will be stopped immediately.

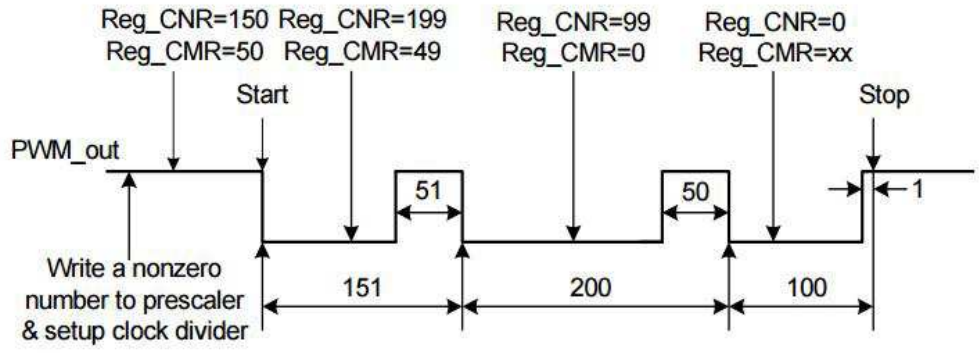


Figure 25–32: PWM Double Buffering Illustration

25.6.2 Modulate Duty Ratio

The double buffering function allows CMR written at any point in current cycle. The loaded value will take effect from next cycle.

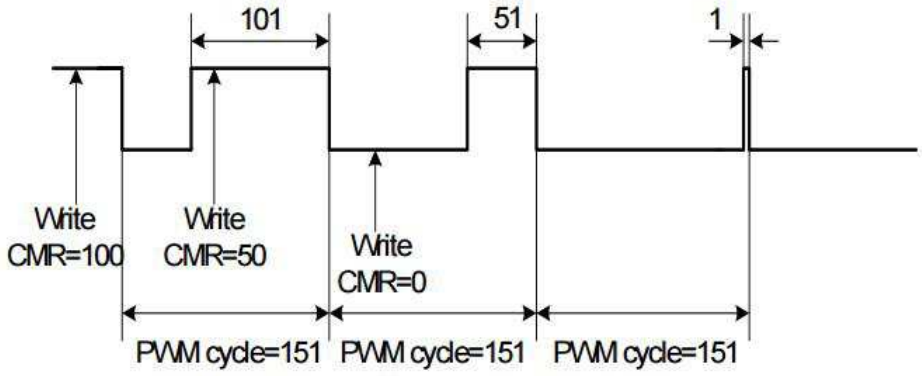


Figure 25–33: PWM Controller Output Duty Ratio

©Levetop Electronics Co., Ltd.

25.6.3 Dead-Zone Generator

PWM is implemented with Dead Zone generator. They are built for power device protection. This function enables generation of a programmable time gap at the rising of PWM output waveform. User can program PPR [31:24] and PPR [23:16] to determine the two Dead Zone interval respectively.

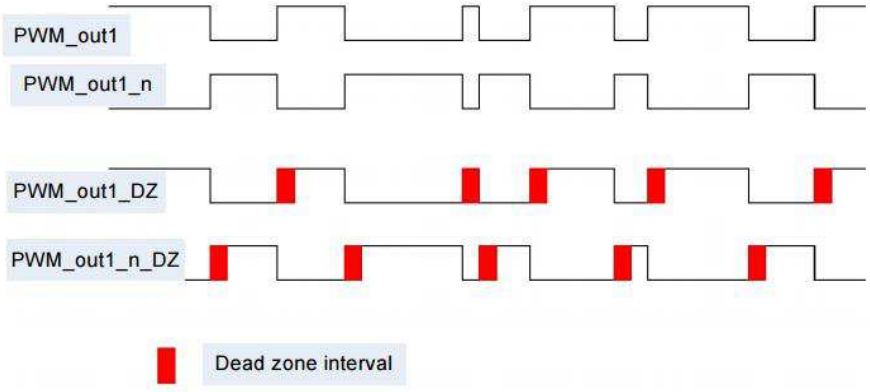


Figure 25–34: Dead Zone Generation Operation

25.6.4 PWM Timer Start Procedure

1. Setup clock selector (CSR)
2. Setup prescale & dead zone interval (PPR)
3. Setup inverter on/off, dead zone generator on/off, toggle mode /one-shot mode, and PWM timer off. (PCR)
4. Setup the comparator register (CMR)
5. Setup the counter register (CNR)
6. Setup the interrupt enable register (PIER)
7. Setup PWMx as output pin (PPCR) 8. Enable PWM timer (PCR)

25.6.5 PWM Timer Stop Procedure

Method 1:
Set 16-bit down counter (CNR) as 0, and monitor PTR. When PTR reaches to 0, disable PWM timer (PCR). (Recommended)

Method 2:
Set 16-bit down counter (CNR) as 0. When interrupt request happen, disable PWM timer (PCR). (Recommended)

Method 3:
Disable PWM timer directly (PCR). (Not recommended)

©Levetop Electronics Co., Ltd.

25.6.6 Capture Start Procedure

1. Setup clock selector (CSR)
2. Setup pre-scale (PPR)
3. Setup inverter on/off, dead zone generator on/off, auto-load mode/one-shot mode, and PWM timer off. (PCR)
4. Setup the counter register (CNR)
5. Setup the capture register (CCR)
6. Setup PWMx as input pin (PPCR)
7. Enable PWM timer (PCR)

25.6.7 Capture Basic Timer Operation

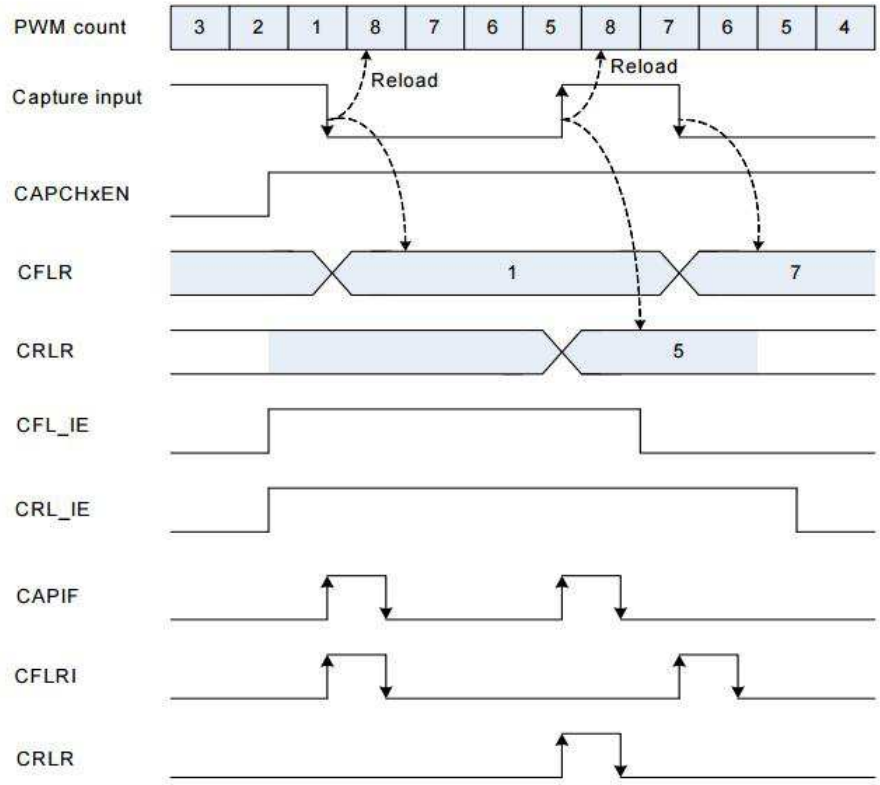


Figure 25–35: Capture Basic Timer Operation

At this case, the CNR is 8:

1. When CAPIFx set 1, the PWM counter CNRx will be reload.
2. The channel low pulse width is (CNR +1 – CRLR).
3. The channel high pulse width is (CNR +1 - CFLR).

©Levetop Electronics Co., Ltd.

26. Comparator Modules (COMP)

26.1 Introduction

The Comparator offers programmable response time, hysteresis, an analog input multiplexer, and two outputs that are optionally available at the Port pins: a synchronous “Filtered” output (CP), or an asynchronous “raw” output (CPA). The asynchronous CPA signal is available even when in when the system clock is not active. This allows the Comparator to operate and generate an output with the device in STOP mode.

26.2 Block Diagram

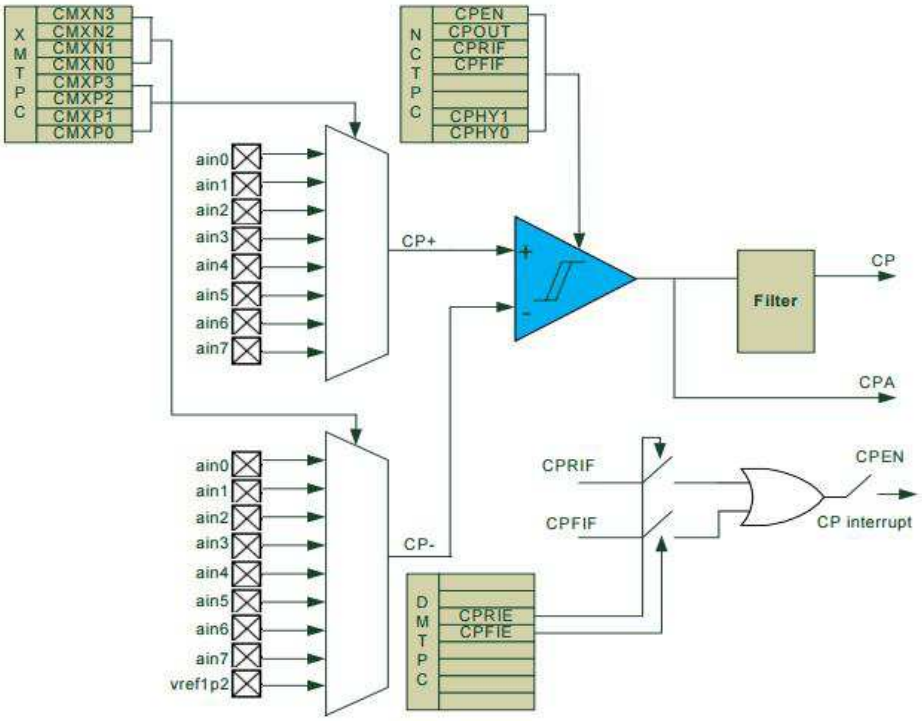


Figure 26–1: Comparator Block Diagram

©Levetop Electronics Co., Ltd.

26.3 Modes of Operation

This subsection describes the three low-power modes.

26.3.1 Wait Mode

In wait mode, the Comparator module can be continues to operate normally by setting CPEN bit and can be configured to exit the low-power mode by generating an interrupt request.

26.3.2 Doze Mode

In wait mode, the Comparator module can be continues to operate normally by setting CPEN bit and can be configured to exit the low-power mode by generating an interrupt request.

26.3.3 Stop Mode

In stop mode, the system clock is absent, and the Comparator module can be continues to operate normally by setting CPEN bit and can be configured to exit the low-power mode by generating an asynchronous “raw” output (CPA) wakeup signal.

©Levetop Electronics Co., Ltd.

26.4 Memory Map and Registers

This subsection describes the memory map and register structure for Comparator.

26.4.1 Memory Map (Base: 0x400a_0000, 0x400b_0000)

Refer to **Table 26–1** for a description of the memory map.
This device has two Comparator modules.

Table 26–1: Comparator Module Memory Map

Offset Address	Bits 7-0	Access(1)
0x3	Comparator Control Register(CPTCN)	S/U
0x2	Comparator0 Mode Selection Register(CPTMD)	S/U
0x1	Comparator MUX Selection Register(CPTMX)	S/U
0x0	Comparator Output Filter Selection Register(CPTFS)	S/U

Note:

1. S = CPU supervisor mode access only. S/U = CPU supervisor or user mode access. User mode accesses to supervisor only addresses have no effect and result in a cycle termination transfer error.

26.4.2 Registers

The Comparator programming model consists of these registers:

- CPTCN: Comparator Control Register.
- CPTMD: Comparator Mode Selection Register.
- CPTMX: Comparator MUX Selection Register.
- CPTFLS: Comparator Output Filter Selection Register

26.4.2.1 Comparator Control Register

Address: 0x3

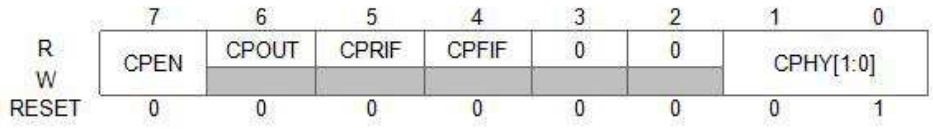


Figure 26–2: Comparator Control Register (CPTCN)

CPEN — Comparator Enable Bit
The read/write CPEN bit enables Comparator operation. When the Comparator is disabled, CPOUT is low state.
1 = Comparator enabled
0 = Comparator disabled

CPOUT— Comparator Output State Flag.
1 = Voltage on CP+ > CP–.
0 = Voltage on CP+ < CP–

©Levetop Electronics Co., Ltd.

CPRIF— Comparator Rising-Edge Flag. Must be cleared by software writing one to this bit.

1 = Comparator Rising Edge has occurred.

0 = No Comparator Rising Edge has occurred since this flag was last cleared.

CPFIF— Comparator Falling-Edge Flag. Must be cleared by software writing one to this bit.

1 = Comparator Falling-Edge has occurred.

0 = No Comp

CPHY[1:0]— Comparator Hysteresis Control Bits.

00: Hysteresis Disabled.

01: Hysteresis = 8 mV.

10: Hysteresis = 12 mV.

11: Hysteresis = 15 mV.

26.4.2.2 Comparator Mode Selection Register

Address: 0x2

	7	6	5	4	3	2	1	0
R	CPOEN	0	CPRIF	CPFIF	0	WKUP LEVEL	0	CPMD
W								
RESET	0	0	0	0	0	0	0	1

=Writes have no effect and the access terminates without a transfer error exception.

Figure 26–3: Comparator Mode Selection Register (CPTMD)

CPOEN— Comparator Output to pad control bit.

1 = Comparator output can be observed in PWM1[0] for COMP0 and PWM1[1] for COMP1.

0 = Comparator output is disabled.

CPRIE— Comparator Rising-Edge Interrupt Enable.

1 = Comparator Rising-edge interrupt enabled.

0 = Comparator Rising-edge interrupt disabled.

CPFIE— Comparator Falling-Edge Interrupt Enable.

1 = Comparator Falling-edge interrupt enabled.

0 = Comparator Falling-edge interrupt disabled.

WKUPLEVEL— Comparator Wake Up level control bit.

1 = Voltage on CP+ > CP– will generate an wakeup request during stop mode.

0 = Voltage on CP+ < CP– will generate an wakeup request during stop mode.

CPMD— Comparator Mode Select. These bits select the response time for Comparator.

Table 26–2: Comparator Mode Selection

Mode	CPMD	Response Time	Power Consumption
Low-speed	0	500ns	1.5uA
High-speed	1	100ns	25uA

©Levetop Electronics Co., Ltd.

26.4.2.3 Comparator MUX Selection Register

Address: 0x1

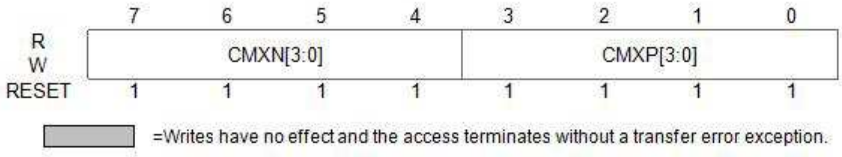


Figure 26–4: Comparator MUX Selection Register (CPTMX)

CMXN[3:0]— Comparator Negative Input MUX Select. These bits select which Port pin is used as the Comparator negative input.

Table 26–3: Comparator Negative Input MUX Selection

CMXN[3]	CMXN[2]	CMXN[1]	CMXN[0]	Negative Input
0	0	0	0	ain0
0	0	0	1	ain1
0	0	1	0	ain2
0	0	1	1	ain3
0	1	0	0	ain4
0	1	0	1	ain5
0	1	1	0	ain6
0	1	1	1	ain7
1	0	0	0	vref1p2
Other value				None

CMXP[3:0]— Comparator Positive Input MUX Select. These bits select which Port pin is used as the Comparator positive input.

Table 26–4: Comparator Positive Input MUX Selection

CMXN[3]	CMXN[2]	CMXN[1]	CMXN[0]	Positive Input
0	0	0	0	ain0
0	0	0	1	ain1
0	0	1	0	ain2
0	0	1	1	ain3
0	1	0	0	ain4
0	1	0	1	ain5
0	1	1	0	ain6
0	1	1	1	ain7
1	x	x	x	None

©Levetop Electronics Co., Ltd.

26.4.2.4 Comparator Output Filter Selection Register

Address: 0x0

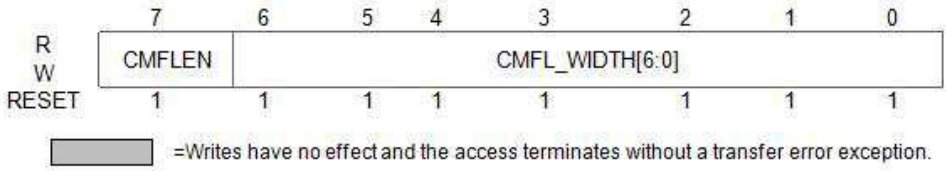


Figure 26–5: Comparator Output Filter Selection Register (CPTFLS)

CMFLEN — Comparator Output Digital Filter Enable

1 = Comparator Output digital filter enable and the CPMRIF and CPMFIF is generated by filted output;

0 = Comparator Output digital filter disable and the CPMRIF and CPMFIF is generated by raw output;

CMFL_WIDTH[6:0] — Comparator Output Digital Filter Pulse Width Selection.

CMFL_WIDTH[6:0] determine the width of input pulse will be filtered. If the input pulse width less than (CMFL_WIDTH[6:0]+2) * Period of fips, the pulse will be filtered.

©Levetop Electronics Co., Ltd.

26.5 Function Description

The Comparator inputs are selected in the CPTMX register. The CMXP3–CMXP0 bits select the Comparator positive input; the CMXN3–CMXN0 bits select the Comparator negative input. Important Note About Comparator Inputs: The Port pins selected as comparator inputs should be configured as analog inputs in their associated Port configuration register and the input, output, pullup and pulldown control should be disabled.

The Comparator output can be polled in software, used as an interrupt source, and/or routed to a Port pin. When routed to a Port pin, the Comparator output is available asynchronous or synchronous to the system clock; the asynchronous output is available even in STOP mode (with no system clock active). When disabled, the Comparator output defaults to the logic low state, and its supply current falls to less than 100nA.

The Comparator hysteresis is software-programmable via its Comparator Control register CPTCN. The user can program the amount of hysteresis voltage. The Comparator hysteresis is programmed using Bits1–0 in the Comparator Control Register CPTCN.

Comparator interrupts can be generated on both rising-edge and falling-edge output transitions. The CPFIF flag is set to logic 1 upon a Comparator falling-edge occurrence, and the CPRIF flag is set to logic 1 upon the Comparator rising-edge occurrence. Once set, these bits remain set until cleared by software. The Comparator rising-edge interrupt mask is enabled by setting CPRIE to a logic1. The Comparator falling-edge interrupt mask is enabled by setting CPFIE to a logic1

The output state of the Comparator can be obtained at any time by reading the CPOUT bit. The Comparator is enabled by setting the CPEN bit to logic 1, and is disabled by clearing this bit to logic 0.

Note that false rising edges and falling edges can be detected when the comparator is first powered on or if changes are made to the hysteresis control bits. Therefore, it is recommended that the rising-edge and falling-edge flags be explicitly cleared to logic 0 a short time after the comparator is enabled or its mode bits have been changed. This least start-up time of the comparator is more than 5us.

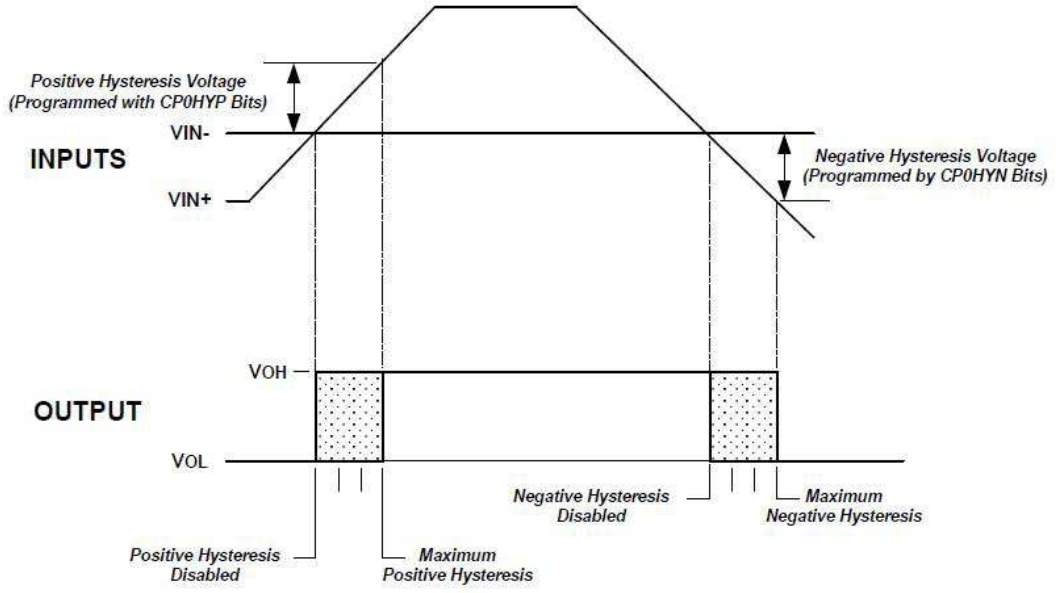


Figure 26–6: Comparator Hysteresis Plot

©Levetop Electronics Co., Ltd.

27. Analog-to-Digital Converter (ADC)

27.1 Introduction

The 12-bit ADC is a successive approximation analog-to-digital converter. It has up to 9 channels allowing it to measure signals from 8 external and 2 internal sources. A/D conversion of the various channels can be performed in single, continuous, scan or discontinuous mode. The results of the ADC are stored in a 12bit x 8depth FIFO, and the data format can be left-aligned or right-aligned.

The analog watchdog feature allows the application to detect if the input voltage goes outside the user-defined higher or lower thresholds.

An efficient low power mode is implemented to allow very low consumption at low frequency.

27.2 ADC Main Features

- High performance
  - 12-bit, 10-bit, 8-bit or 6-bit configurable resolution
  - ADC conversion time: 1.0μs for 12-bit resolution (1 MHz), 0.88μs conversion time for 10 bit resolution, faster conversion times can be obtained by lowering resolution.
  - Programmable sampling time
  - Data alignment with built-in data coherency
  - DMA support
- Low power
  - Application can reduce PLCK frequency for low power operation while still keeping optimum ADC performance. For example, 1.0μs conversion time is kept, whatever the frequency of PCLK.
  - Wait mode: prevents ADC overrun in applications with low frequency PLCK
  - Auto off mode: ADC is automatically powered off except during the active conversion phase. This dramatically reduces the power consumption of the ADC.
- Analog input channels
  - 8 external analog inputs
  - 1 channel for internal reference voltage
  - 1 channel for internal temperature sensor
- Start-of-conversion can be initiated:
  - By software
  - By hardware triggers with configurable polarity
- Conversion modes
  - Can convert a single channel or can scan a sequence of channels.
  - Single mode converts selected inputs once per trigger
  - Continuous mode converts selected inputs continuously
  - Discontinuous mode
- Interrupt generation at the end of sampling, end of conversion, end of sequence conversion, and in case of analog watchdog or overrun events.
- Analog watchdog
- Single-ended and differential-input configurations
- Converter uses an internal reference or an external reference

©Levetop Electronics Co., Ltd.

27.3 ADC Functional Description

Figure 27–1shows the ADC block diagram

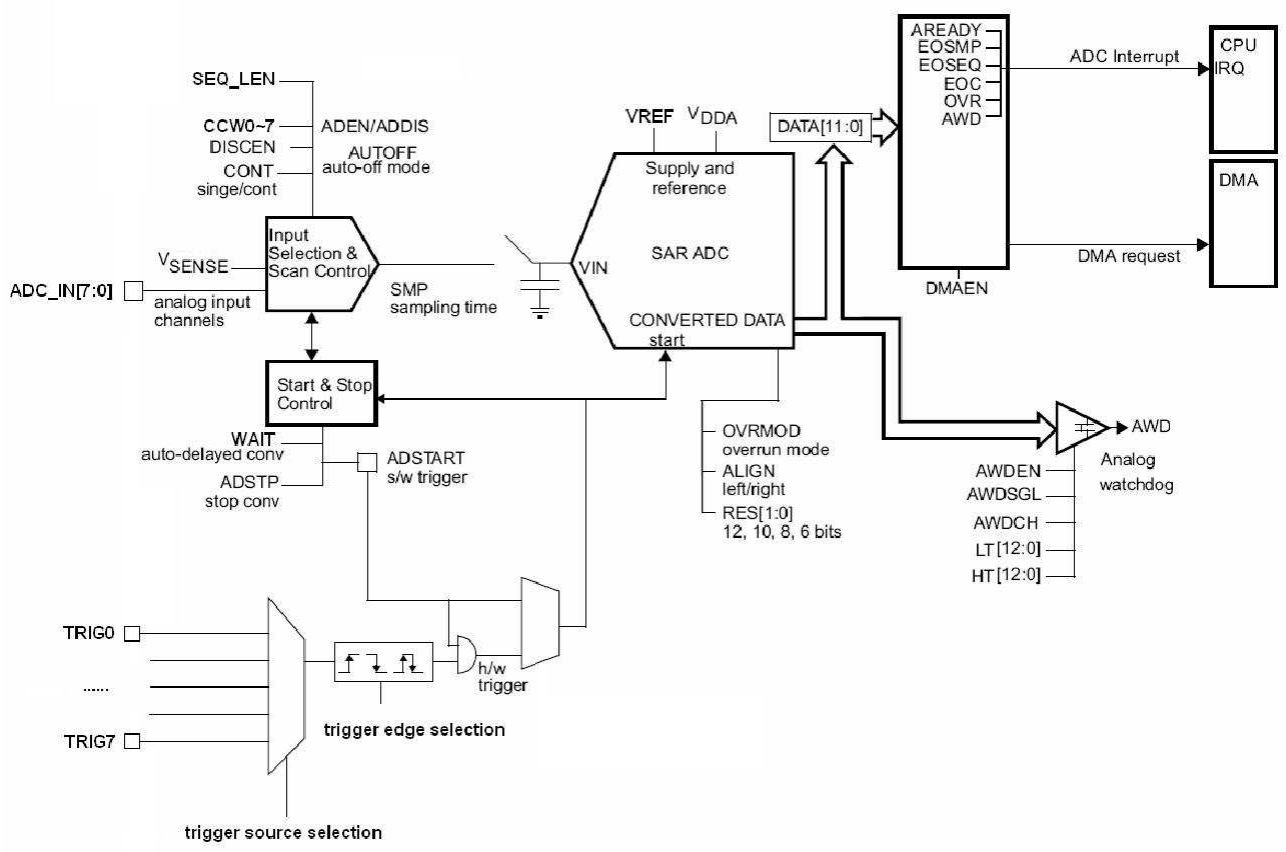


Figure 27–1: ADC Block Diagram

27.3.1 ADC On-Off Control (ADEN, ADDIS, ADRDY)

At MCU power-up, the ADC is disabled and put in power-down mode (ADEN=0). As shown in **Figure 27–2**, the ADC needs a stabilization time of tSTAB (~2.0μs) before it starts converting accurately.

Two control bits are used to enable or disable the ADC:

- Set ADEN=1 to enable the ADC. The ADRDY flag is set as soon as the ADC is ready for operation.
- Set ADDIS=1 to disable the ADC and put the ADC in power down mode. The ADEN and ADDIS bits are then automatically cleared by hardware as soon as the ADC is fully disabled.

Conversion can then start either by setting ADSTART=1 or when an external trigger event occurs if triggers are enabled.

Follow this procedure to enable the ADC:

- Set ADEN=1 in the ADC_CR register.
- Wait until ADRDY=1 in the ADC_ISR register (ADRDY is set after the ADC startup time). This can be handled by interrupt if the interrupt is enabled by setting the ADRDYIE bit in the ADC_IER register.

©Levetop Electronics Co., Ltd.

Follow this procedure to disable the ADC:

- Check that ADSTART=0 in the ADC_CR register to ensure that no conversion is ongoing. If required, stop any ongoing conversion by writing 1 to the ADSTP bit in the ADC_CR register and waiting until this bit is read at 0.
- Set ADDIS=1 in the ADC_CR register.
- If required by the application, wait until ADEN=0 in the ADC_CR register, indicating that the ADC is fully disabled (ADDIS is automatically reset once ADEN=0).

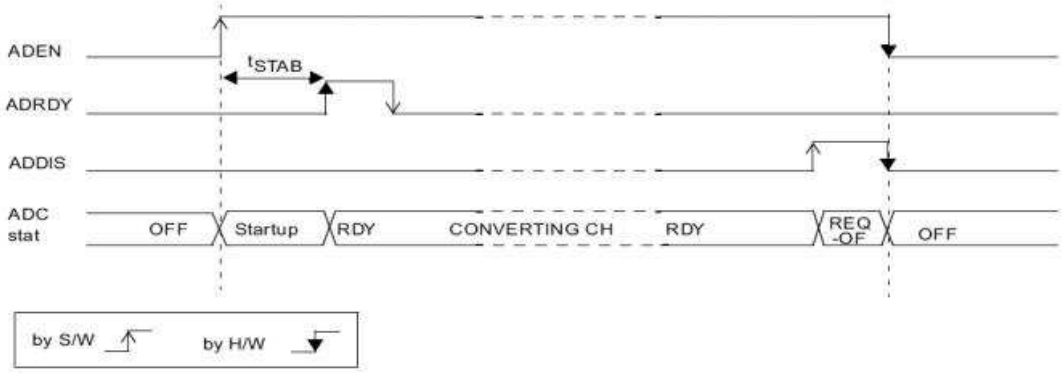


Figure 27–2: Enabling/disabling The ADC

Note:

In auto-off mode (AUTOFF=1) the power-on/off phases are performed automatically, by hardware and the ADRDY flag is not set.

27.3.2 ADC Clock

The ADC has a dual clock-domain architecture, as show in **Figure 27–3**, this has the advantage of reaching the maximum ADC clock frequency whatever the IPG clock scheme selected.

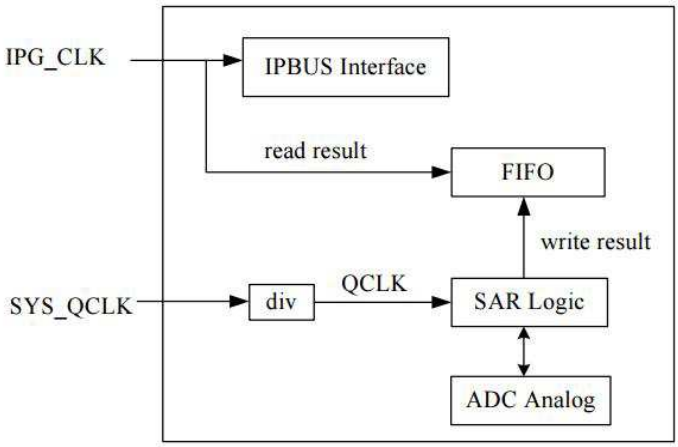


Figure 27–3: ADC Clock Scheme

©Levetop Electronics Co., Ltd.

27.3.3 Configuring the ADC

Software must write to the ADEN bit in the ADC_CR register if the ADC is disabled (ADEN must be 0).

Software must only write to the ADSTART and ADDIS bits in the ADC_CR register only if the ADC is enabled and there is no pending request to disable the ADC (ADEN = 1 and ADDIS = 0).

For all the other control bits in the ADC_IER, ADC_CFGRi, ADC_SMPR, ADC_TR, ADC_CHSELRi and ADC_WDG registers, software must only write to the configuration control bits if there is no conversion ongoing (ADSTART = 0).

Software must only write to the ADSTP bit in the ADC_CR register if the ADC is enabled (and possibly converting) and there is no pending request to disable the ADC (ADSTART = 1 and ADDIS = 0) .

Note:

There is no hardware protection preventing software from making write operations forbidden by the above rules. If such a forbidden write access occurs, the ADC may enter an undefined state. To recover correct operation in this case, the ADC must be disabled(ADDIS = 1).

27.3.4 Channel Selection (CCWi)

There are up to 10 multiplexed channels:

- 8 analog inputs from pins (ADC_IN0...ADC_IN7)
- 2 internal analog input (VREFINT & Temperature Sensor)

It is possible to convert a single channel or to automatically scan a sequence of channels.

The sequence of the channels to be converted is organized as CCW[0] ,then CCW[1], ... ,then CCW[7]. The CCWi are programmed in ADC_CHSELRi. The sequence length is programmed in SEQ_LEN[2:0] of ADC_CFGR2. For example, if sequence length is set as 3, then the sequence is organized as CCW[0], then CCW[1], then CCW[2].
The channel decode is shown in **Table 27–1.**

Table 27–1: Channel Decode

CCWi[3:0]	channel select
4’b0000	ADC_IN0
4’b0001	ADC_IN1
4’b0010	ADC_IN2
4’b0011	ADC_IN3
4’b0100	ADC_IN4
4’b0101	ADC_IN5
4’b0110	ADC_IN6
4’b0111	ADC_IN7
4’b1110	VREFINT
4’b1111	Temperature Sensor

©Levetop Electronics Co., Ltd.

27.3.5 Programmable Sampling Time (SMP)

Before starting a conversion, the ADC needs to establish a direct connection between the voltage source to be measured and the embedded sampling capacitor of the ADC. This sampling time must be enough for the input voltage source to charge the sample and hold capacitor to the input voltage level.

Having a programmable sampling time allows to trim the conversion speed according to the input resistance of the input voltage source. The ADC samples the input voltage for a number of ADC clock cycles that can be modified using the SMP[3:0] bits in the ADC_SMPR register. This programmable sampling time is common to all channels.

The ADC indicates the end of the sampling phase by setting the EOSMP flag.

27.3.6 Single Conversion Mode (CONT=0)

In Single conversion mode, the ADC converts the sequence once. This mode is selected when CONT=0 in the ADC_CFGR1 register. Conversion is started by either:

- Setting the ADSTART bit in the ADC_CR register
- Hardware trigger event

Inside the sequence, after each conversion is complete:

- The converted data are stored in the FIFO
- The EOC (end of conversion) flag is set
- An interrupt is generated if the EOCIE bit is set

After the sequence of conversions is complete:

- The EOSEQ (end of sequence) flag is set
- An interrupt is generated if the EOSEQIE bit is set

Then the ADC stops until a new external trigger event occurs or the ADSTART bit is set again.

Note:
To convert a single channel once, program a sequence with a length of 1.

©Levetop Electronics Co., Ltd.

27.3.7 Continuous Conversion Mode (CONT=1)

In continuous conversion mode, when a software or hardware trigger event occurs, the ADC performs a sequence of conversions, converting the sequence once and then automatically re-starts and continuously performs the same sequence of conversions. This mode is selected when CONT=1 in the ADC_CFGR1 register. Conversion is started by either:

- Setting the ADSTART bit in the ADC_CR register
- Hardware trigger event

Inside the sequence, after each conversion is complete:

- The converted data are stored in the FIFO
- The EOC (end of conversion) flag is set
- An interrupt is generated if the EOCIE bit is set

After the sequence of conversions is complete:

- The EOSEQ (end of sequence) flag is set
- An interrupt is generated if the EOSEQIE bit is set

Then, a new sequence restarts immediately and the ADC continuously repeats the conversion sequence.

Note:

It is not possible to have both discontinuous mode and continuous mode enabled: it is forbidden to set both bits DISCEN=1 and CONT=1.

27.3.8 Starting Conversions (ADSTART)

Software starts ADC conversions by setting ADSTART=1.

When ADSTART is set, the conversion:

- Starts immediately if TRIGMODE = 0x0 (software trigger)
- At the next active edge of the selected hardware trigger if TRIGMODE ≠ 0x0

The ADSTART bit is also used to indicate whether an ADC operation is currently ongoing. It is possible to re-configure the ADC while ADSTART=0, indicating that the ADC is idle.

The ADSTART bit is cleared by hardware:

- In single mode with software trigger
  - At any end of conversion sequence (EOSEQ=1)
- In discontinuous mode with software trigger
  - At any end of conversion
- In all cases
  - After execution of the ADSTP procedure invoked by software

Note:

- In continuous mode (CONT=1), the ADSTART bit is not cleared by hardware when the EOSEQ flag is set because the sequence is automatically relaunched.
- When hardware trigger is selected in single mode, ADSTART is not cleared by hardware when the EOSEQ flag is set. This avoids the need for software having to set the ADSTART bit again and ensures the next trigger event is not missed.

©Levetop Electronics Co., Ltd.

27.3.9 Timings

The elapsed time between the start of a conversion and the end of conversion is the sum of the configured sampling time plus the successive approximation time depending on data resolution:

tADC = tSMPL + tSAR = [4|min + 12|12bit] x tQCLK = 1μs |min (for fQCLK = 16 MHz)

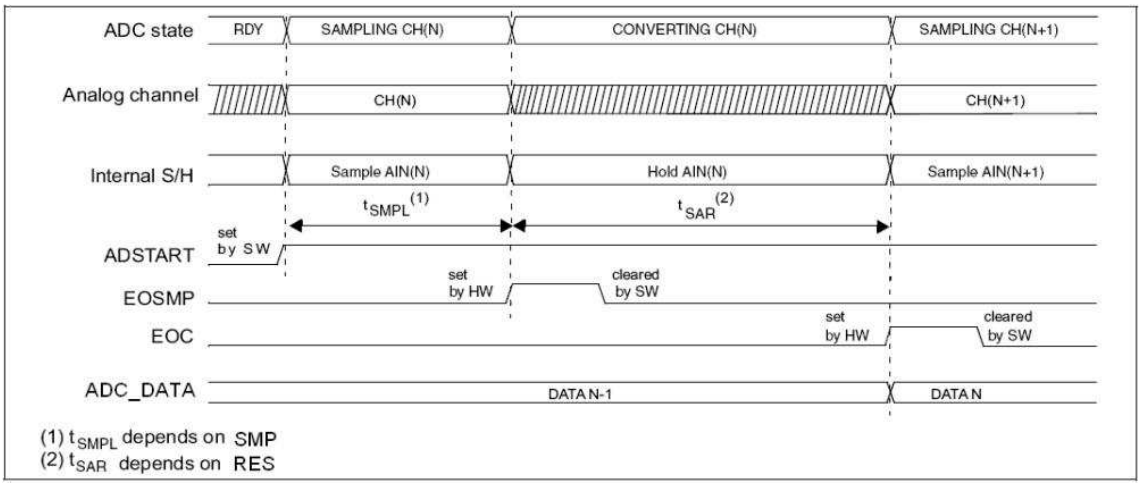


Figure 27–4: Analog To Digital Conversion Time

27.3.10 Stopping An Ongoing Conversion (ADSTP)

The software can decide to stop any ongoing conversions by setting ADSTP=1 in the ADC_CR register.

This will reset the ADC operation and the ADC will be idle, ready for a new operation.

When the ADSTP bit is set by software, any ongoing conversion is aborted and the result is discarded (FIFO is not updated with the current conversion). The scan sequence is also aborted and reset (meaning that restarting the ADC would restart a new sequence).

Once this procedure is complete, the ADSTP and ADSTART bits are both cleared by hardware and the software must wait until ADSTART=0 before starting new conversions.

Note:
The flags in QADC_ISR are not cleared by STOP command, and the data in FIFO are not lost.

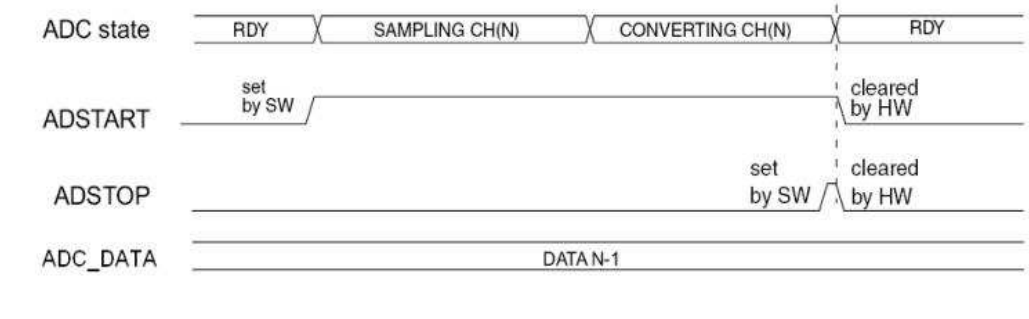


Figure 27–5: Stopping An Ongoing Conversion

©Levetop Electronics Co., Ltd.

27.4 Conversion On External Trigger and Trigger Polarity

A conversion or a sequence of conversion can be triggered either by software or by an external event. If the TRIGMODE control bits are not equal to “0”, then external events are able to trigger a conversion with the selected polarity. The trigger selection is effective once software has set bit ADSTART=1.
Any hardware triggers which occur while a conversion is ongoing are ignored.

If bit ADSTART=0, any hardware triggers which occur are ignored.

Table 27–2 provides the correspondence between the TRIGMODE values and the trigger polarity.

Table 27–2: Configuring The Trigger Polarity

TRIGMODE[2:0]	Source
3’b000	Trigger detection disabled, software trigger
3’b001	Detection on rising edge
3’b010	Detection on falling edge
3’b011	Detection on both rising and falling edges
3’b100	Detection on high level voltage
3’b101	Detection on low level voltage
3’b110	Detection on PIT 0
3’b111	Detection on PWM 0

Note:
The polarity of the external trigger can be changed only when the ADC is not converting (ADSTART= 0).

The TRIGSCR control bits are used to select which of 8 possible events can trigger conversions. **Table 27–3** gives the possible external trigger for regular conversion. Software source trigger events can be generated by setting the ADSTART bit in the ADC_CR register.

Table 27–3: External Triggers

TRIGSCR[2:0]	Name	Source
3’b000	TRG0	
3’b001	TRG1	
3’b010	TRG2	
3’b011	TRG3	
3’b100	TRG4	
3’b101	TRG5	
3’b110	TRG6	
3’b111	TRG7	

Note:
The trigger selection can be changed only when the ADC is not converting (ADSTART= 0).

©Levetop Electronics Co., Ltd.

27.4.1 Discontinuous Mode (DISCEN)

This mode is enabled by setting the DISCEN bit in the ADC_CFGR1 register.
In this mode (DISCEN=1), a hardware or software trigger event is required to start each conversion defined in the sequence. On the contrary, if DISCEN=0, a single hardware or software trigger event successively starts all the conversions defined in the sequence.
Example:

- DISCEN=1, channels to be converted = 0, 3, 7, 10
  - 1st trigger: channel 0 is converted and an EOC event is generated
  - 2nd trigger: channel 3 is converted and an EOC event is generated
  - 3rd trigger: channel 7 is converted and an EOC event is generated
  - 4th trigger: channel 10 is converted and both EOC and EOSEQ events are generated.
  - 5th trigger: channel 0 is converted an EOC event is generated
  - 6th trigger: channel 3 is converted and an EOC event is generated
  - ...
- DISCEN=0, channels to be converted = 0, 3, 7, 10
  - 1st trigger: the complete sequence is converted: channel 0, then 3, 7 and 10. Each conversion generates an EOC event and the last one also generates an EOSEQ event.
  - Any subsequent trigger events will restart the complete sequence.

27.4.2 Programmable Resolution (RES) - Fast Conversion Mode

It is possible to obtain faster conversion times (tSAR) by reducing the ADC resolution. The resolution can be configured to be either 12, 10, 8, or 6 bits by programming the RES[1:0] bits in the ADC_CFGR1 register. Lower resolution allows faster conversion times for applications where high data precision is not required.

Note:
The RES[1:0] bit must only be changed when the ADEN bit is reset.
The result of the conversion is always 13 bits wide and any unused LSB bits are read as zeroes.
Lower resolution reduces the conversion time needed for the successive approximation steps.

27.4.3 End of Conversion, End of Sampling Phase (EOC, EOSMP Flags)

The ADC indicates each end of conversion (EOC) event.

The ADC sets the EOC flag in the ADC_ISR register as soon as a new conversion data result is available. An interrupt can be generated if the EOCIE bit is set in the ADC_IER register. The EOC flag is cleared by software either by writing 1 to it, or by reading the FIFO.

The ADC also indicates the end of sampling phase by setting the EOSMP flag in the ADC_ISR register. The EOSMP flag is cleared by software by writing1 to it. An interrupt can be generated if the EOSMPIE bit is set in the ADC_IER register.

27.4.4 End of Conversion Sequence (EOSEQ Flag)

The ADC notifies the application of each end of sequence (EOSEQ) event.

The ADC sets the EOSEQ flag in the ADC_ISR register as soon as the last data result of a conversion sequence is available in the FIFO. An interrupt can be generated if the EOSEQIE bit is set in the ADC_IER register. The EOSEQ flag is cleared by software by writing 1 to it.

©Levetop Electronics Co., Ltd.

27.4.5 Example Timing Diagrams

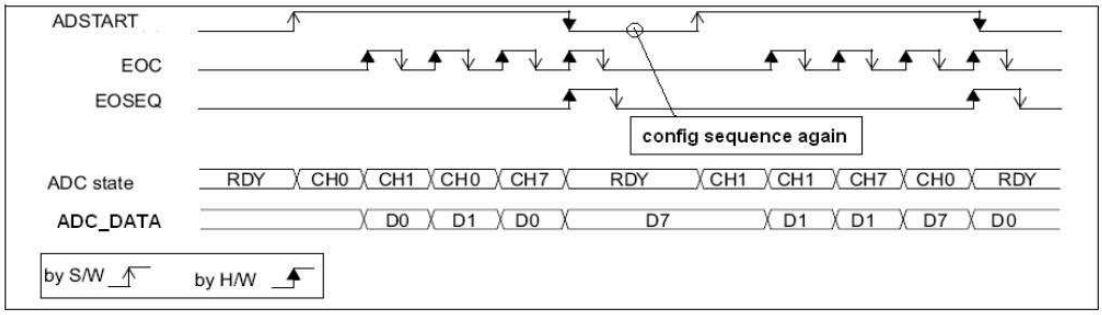


Figure 27–6: Single Conversions of A Sequence, Software Trigger

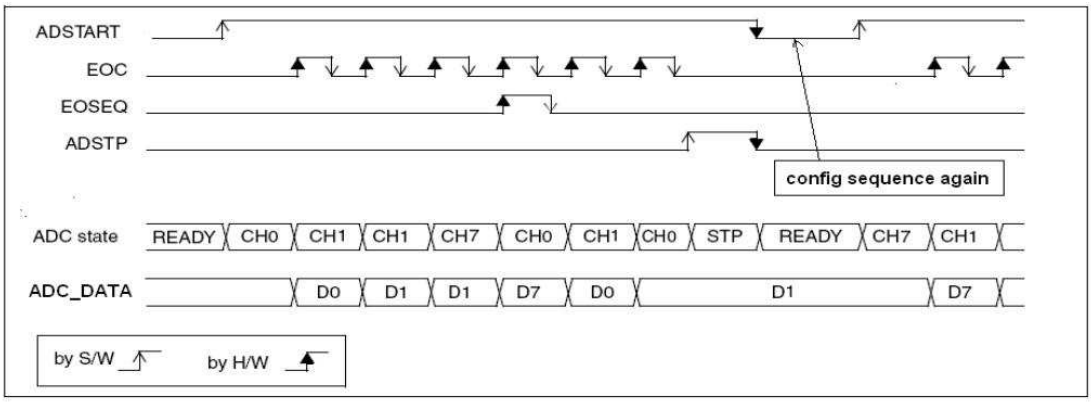


Figure 27–7: Continuous Conversion of A Sequence, Software Trigger

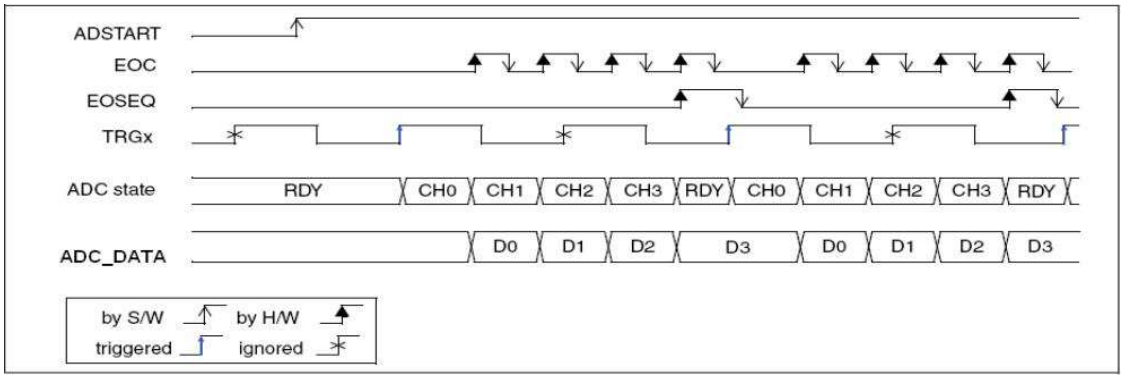


Figure 27–8: Single Conversions of A Sequence, Hardware Trigger

©Levetop Electronics Co., Ltd.

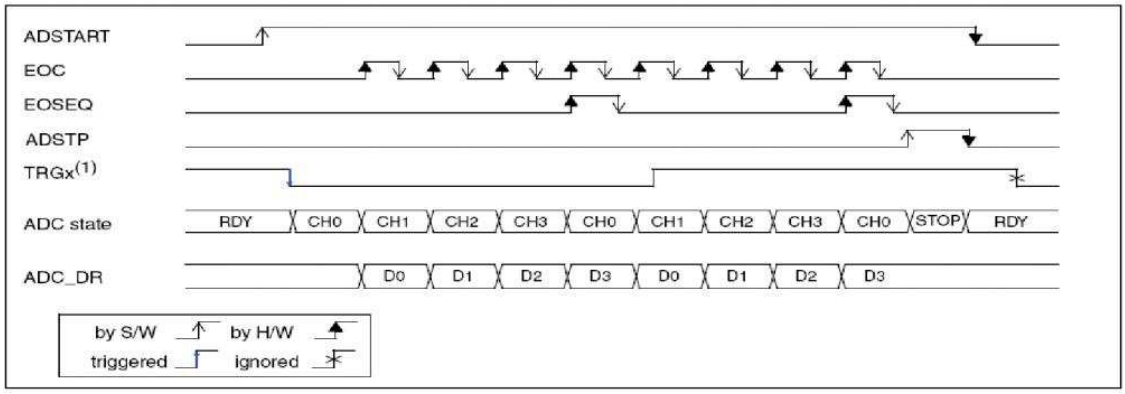


Figure 27–9: Continuous Conversions of A Sequence, Hardware Trigger

©Levetop Electronics Co., Ltd.

27.5 Data management

27.5.1 Data FIFO & Data Alignment (ADC_FIFO, ALIGN)

At the end of each conversion (when an EOC event occurs), the result of the converted data is stored in the ADC_FIFO which is 13-bit wide x 8 depth.

The format of the read out data depends on the configured data alignment and resolution.

The ALIGN bit in the ADC_CFGR1 register selects the alignment of the data stored after conversion. Data can be right-aligned (ALIGN=0) or left-aligned (ALIGN=1) as shown in **Figure 27–10**.

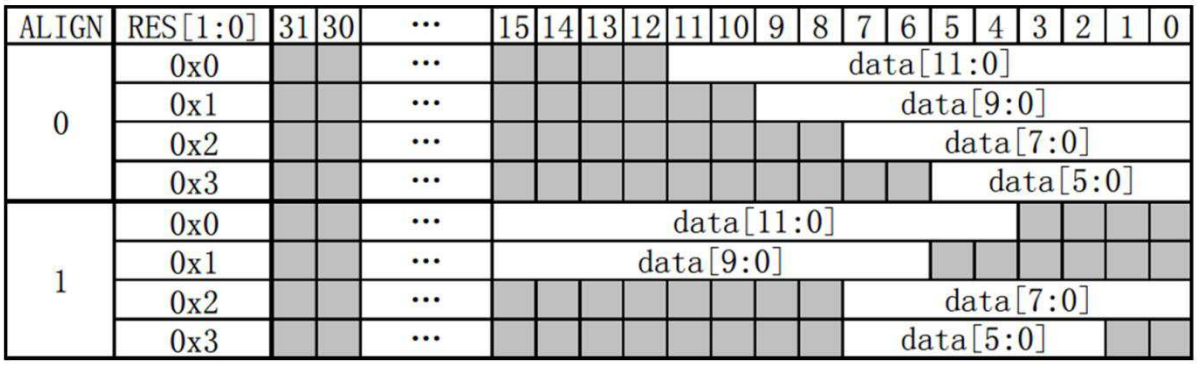


Figure 27–10: Data Alignment and Resolution

The FIFO supports byte, half-word and word read, but the address offset should be always 0x4c. For different data alignments and resolutions, users should note:

- If read by word, then the data format is as data[31:0] as **Figure 27–10** shows
- if read by half word, then the data format is as data[15:0] as **Figure 27–10** shows
- If read by byte when the data is longer than 8bit, then the high byte is first read out, the low byte should read again.
- If read by byte when the data is no longer than 8bit, then the data format is as data[7:0] as **Figure 27–10** shows

©Levetop Electronics Co., Ltd.

27.5.2 ADC Overrun (OVR, OVRMOD)

The overrun flag (OVR) indicates a data overrun event, when the converted data was not read in time by the CPU or the DMA, before the FIFO is full.

The OVR flag is set in the ADC_ISR register if the FULL flag is still at '1' at the time when a new conversion completes. An interrupt can be generated if the OVRIE bit is set in the ADC_IER register.

When an overrun condition occurs, the ADC keeps operating and can continue to convert unless the software decides to stop and reset the sequence by setting the ADSTP bit in the ADC_CR register.

The OVR flag is cleared by software by writing 1 to it.

It is possible to configure if the data is preserved or overwritten when an overrun event occurs by programming the OVRMOD bit in the ADC_CFGR1 register:

- OVRMOD=0
  - An overrun event preserves the data register from being overwritten: the old data is maintained and the new conversion is discarded. If OVR remains at 1, further conversions can be performed but the resulting data is discarded.
- OVRMOD=1
  - The data register is overwritten with the last conversion result. If OVR remains at 1, further conversions can be performed and the FIFO always contains the data from the latest conversion.

27.5.3 Managing A Sequence of Data Converted Without Using The DMA

If the conversions are slow enough, the conversion sequence can be handled by software. In this case the software can use the EOC flag and its associated interrupt to handle each data result. Each time a conversion is complete, the EOC bit is set in the ADC_ISR register and the FIFO register can be read.

Software can also use FIFO EMPTY flag to handle each data result. If EMPTY is not "0", this means FIFO has new data.

The OVRMOD bit in the ADC_CFGR1 register should be configured to 0 to manage overrun events as an error.

27.5.4 Managing Converted Data Without Using The DMA Without Overrun

It may be useful to let the ADC convert one or more channels without reading the data after each conversion. In this case, the OVRMOD bit must be configured at 1 and the OVR flag should be ignored by the software. When OVRMOD=1, an overrun event does not prevent the ADC from continuing to convert and the FIFO always contains the latest conversion data.

©Levetop Electronics Co., Ltd.

27.5.5 Managing Converted Data Using DMA

Once the data number in the FIFO is not empty and bit DMAEN is set, QADC will send request to the DMA. This allows the transfer of the converted data from the FIFO to the destination location selected by the software.

Despite this, if an overrun occurs (OVR=1) because the DMA could not serve the DMA transfer request in time, the ADC stops generating DMA requests and the data corresponding to the new conversion is not transferred by the DMA. Which means that all the data transferred to the RAM can be considered as valid.

Depending on the configuration of OVRMOD bit, the data is either preserved or overwritten.

The DMA transfer requests are blocked until the software clears the OVR bit.

27.6 Low Power Features

27.6.1 Wait Mode Conversion

Wait mode conversion can be used to simplify the software as well as optimizing the performance of applications clocked at low frequency where there might be a risk of ADC overrun occurring. When the WAIT bit is set to 1 in the ADC_CFGR1 register, a new conversion can start only if the FIFO is not full.

This is a way to automatically adapt the speed of the ADC to the speed of the system that reads the data.

Note:

Any hardware triggers which occur while a conversion is ongoing or during the wait time preceding the read access are ignored.

27.6.2 Auto-Off Mode (AUTOFF)

The ADC has an automatic power management feature which is called auto-off mode, and is enabled by setting AUTOFF=1 in the ADC_CFGR1 register.

When AUTOFF=1, the ADC is always powered off when not converting and automatically wakes-up when a conversion is started (by software or hardware trigger). A startup-time is automatically inserted between the trigger event which starts the conversion and the sampling time of the ADC. The ADC is then automatically disabled once the sequence of conversions is complete.

Auto-off mode can cause a dramatic reduction in the power consumption of applications which need relatively few conversions or when conversion requests are timed far enough apart (for example with a low frequency hardware trigger) to justify the extra power and extra time used for switching the ADC on and off.

Auto-off mode can be combined with the wait mode conversion (WAIT=1) for applications clocked at low frequency. This combination can provide significant power savings if the ADC is automatically powered-off during the wait phase and restarted as soon as the FIFO is read by the application

©Levetop Electronics Co., Ltd.

27.7 Analog Window Watchdog

The AWD analog watchdog feature is enabled by setting the AWDEN bit in the ADC_CFGR1 register. It is used to monitor that either one selected channel or all enabled channels remain within a configured voltage range (window) as shown in **Figure 27–11.**

The AWD analog watchdog status bit is set if the analog voltage converted by the ADC is below a lower threshold or above a higher threshold. These thresholds are programmed in the ADC_TR register. An interrupt can be enabled by setting the AWDIE bit in the ADC_IER register.

The AWD flag is cleared by software by writing 1 to it.

When converting a data with a resolution of less than 12-bit (according to bits RES[1:0]), the LSB of the programmed thresholds must be kept cleared because the internal comparison is always performed on the full 12-bit raw converted data (left aligned).

Table 27–4 shows how to configure the AWDSGL and AWDEN bits in the ADC_CFGR1 register to enable the analog watchdog on one or more channels.

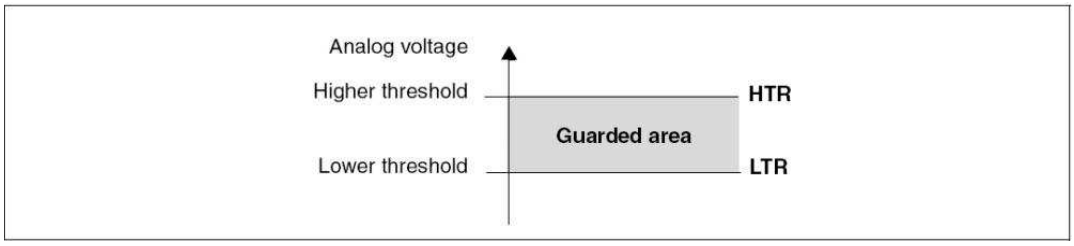


Figure 27–11: Analog Watchdog Guarded Area

Table 27–4: Analog Watchdog Channel Selection

Channels Guarded by the Analog Watchdog	AWDSGL bit	AWDEN bit
None	x	0
All channels	0	1
Single(1) channel	1	1

Note:
1. Selected by the AWDCH.

©Levetop Electronics Co., Ltd.

27.8 Temperature Sensor

The temperature sensor is internally connected to the ADC1_IN15 input channel which is used to convert the sensor's output voltage to a digital value.

27.9 ADC Interrupts

An interrupt can be generated by any of the following events:

- ADC power-up, when the ADC is ready (ADRDY flag)
- End of any conversion (EOC flag)
- End of a sequence of conversions (EOSEQ flag)
- When an analog watchdog detection occurs (AWD flag)
- When the end of sampling phase occurs (EOSMP flag)
- When a data overrun occurs (OVR flag) Separate interrupt enable bits are available for flexibility.

Table 27–5: ADC Interrupts

Interrupt event	Event flag	Enable control bit
ADC ready	ADRDY	ADRDYIE
End of conversion	EOC	EOCIE
End of sequence of conversions	EOSEQ	EOSEQIE
Analog watching status bit is set	AWD	AWDIE
End of sampling phase	EOSMP	EOSMPIE
Overrun	OVR	OVRIE

©Levetop Electronics Co., Ltd.

27.10 Memory Map and Registers

This subsection describes the memory map and register structure.

27.10.1 Memory Map (Base: 0x4011_0000)

Refer to **Table 27–6** for a description of the QADC memory map.

Table 27–6: QADC Memory Map

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	ADC_ISR																									AWD	Empty	FULL	OVR	EOSEQ	EOC	EOSMP	ADRDY
0x04	ADC_IER																									AWDIE			OVRIE	EOSEQIE	EOCIE	EOSMPIE	ADRDYIE
0x08	ADC_CR																													ADSTP	ADSTART	ADDIS	ADEN
0x0c	ADC_CFGR1	DIFF	OVRMOD				SEQ_LEN [2:0]			DISCEN	AUTOFF	WAIT	CONT		TRIGSCR [2:0]					TRIGMODE [2:0]		ALIGN	RES [1:0]										DMAEN
0x10	ADC_CFGR2																					QPR [3:0]				STCNT [7:0]							
0x14	ADC_SMPR																									SMP [7:0]							
0x18	ADC_WDG																									AWDEN	AWDSGL			AWDCH [3:0]			
0x1c	ADC_TR					HT [11:0]																LT [11:0]											
0x2c	ADC_CHSELR1					CCW3 [3:0]								CCW2 [3:0]								CCW1 [3:0]								CCW0 [3:0]			
0x30	ADC_CHSELR2					CCW7 [3:0]								CCW6 [3:0]								CCW5 [3:0]								CCW4 [3:0]			
0x4c	ADC_FIFO																	DATA [15:0]															

Notes:

1. All the registers are CPU supervisor or user mode accessible.
2. The dark bits are reserved, and must be kept at reset value.

©Levetop Electronics Co., Ltd.

27.10.2 Registers

27.10.2.1 ADC Interrupt and Status Register (ADC_ISR)

Address Offset: 0x00

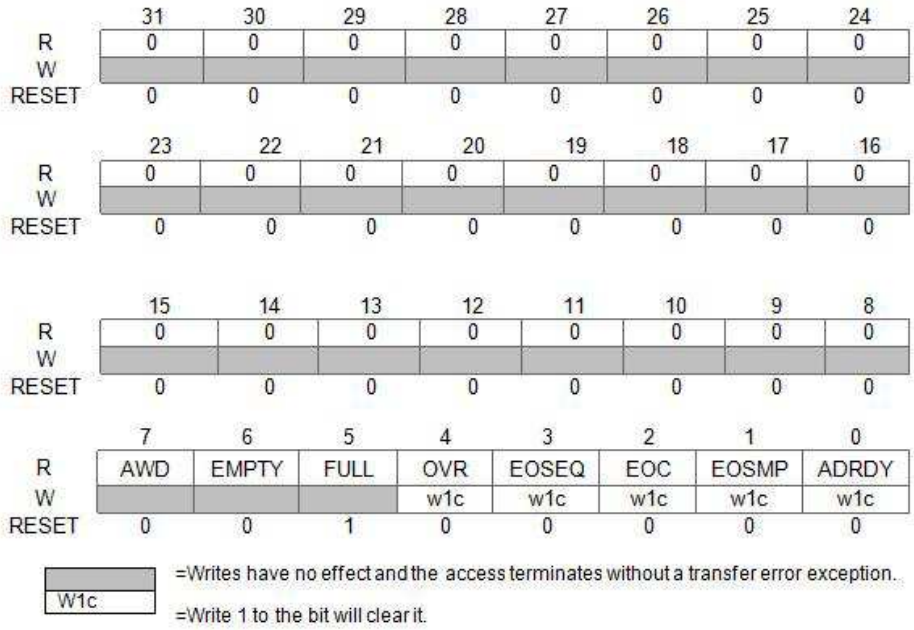


Figure 27–12: ADC Interrupt and Status Register (ADC_ISR)

Read: Anytime

Write: Anytime

AWD — Analog watchdog flag
This bit is set by hardware when the converted voltage crosses the values programmed in the ADC_TR register. It is cleared by software writing 1 to it.
1 = Analog watchdog event occurred
0 = No analog watchdog event occurred (or the flag event was already acknowledged and cleared by software)

EMPTY — FIFO empty status
This bit is set by hardware when the FIFO is empty. It is cleared by hardware when FIFO is not empty.
1 = FIFO is empty
0 = FIFO is not empty

FULL — FIFO full status
This bit is set by hardware when the FIFO is full. It is cleared by hardware when FIFO is not full.
1 = FIFO is full
0 = FIFO is not full

OVR — ADC overrun
This bit is set by hardware when an overrun occurs, meaning that a new conversion has complete while the FULL flag was already set. It is cleared by software writing 1 to it.
1 = Overrun has occurred
0 = No overrun occurred (or the flag event was already acknowledged and cleared by software)

©Levetop Electronics Co., Ltd.

EOSEQ — End of sequence flag

This bit is set by hardware at the end of the conversion of a sequence. It is cleared by software writing 1 to it.

1 = Conversion sequence complete

0=Conversion sequence not complete (or the flag event was already acknowledged and cleared by software)

EOC — End of conversion flag

This bit is set by hardware at the end of each conversion of a channel when a new data result is available in the ADC_FIFO register. It is cleared by software writing 1 to it or by reading the ADC_FIFO register.

1 = Channel conversion complete

0 = Channel conversion not complete (or the flag event was already acknowledged and cleared by software)

EOSMP — End of sampling flag

This bit is set by hardware during the conversion, at the end of the sampling phase.

1 = End of sampling phase reached

0 =Not at the end of the sampling phase (or the flag event was already acknowledged and cleared by software)

ADRDY — ADC ready

This bit is set by hardware after the ADC has been enabled (bit ADEN=1) and when the ADC reaches a state where it is ready to accept conversion requests. It is cleared by software writing 1 to it.

1 = ADC is ready to start conversion

0=ADC not yet ready to start conversion (or the flag event was already acknowledged and cleared by software)

27.10.2.2 ADC Interrupt Enable Register (ADC_IER)

Address Offset: 0x04

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	AWDIE	0	0	OVRIE	EOSEQIE	EOCIE	EOSMPIE	ADRDYIE
W								
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 27–13: ADC Interrupt Enable Register (ADC_IER)

©Levetop Electronics Co., Ltd.

Read: Anytime

Write: Before ADC start

AWDIE — Analog watchdog interrupt enable
This bit is set and cleared by software to enable/disable the analog watchdog interrupt.
1 = Analog watchdog interrupt enabled
0 = Analog watchdog interrupt disabled

OVRIE — Overrun interrupt enable
This bit is set and cleared by software to enable/disable the overrun interrupt.
1 = Overrun interrupt enabled. An interrupt is generated when the OVR bit is set.
0 = Overrun interrupt disabled

EOSEQIE — End of conversion sequence interrupt enable
This bit is set and cleared by software to enable/disable the end of sequence of conversions interrupt.
1 = EOSEQ interrupt enabled. An interrupt is generated when the EOSEQ bit is set.
0 = EOSEQ interrupt disabled

EOCIE — End of conversion interrupt enable
This bit is set and cleared by software to enable/disable the end of conversion interrupt.
1 = EOC interrupt enabled. An interrupt is generated when the EOC bit is set.
0 = EOC interrupt disabled

EOSMPIE — End of sampling flag interrupt enable
This bit is set and cleared by software to enable/disable the end of the sampling phase interrupt.
1 = EOSMP interrupt enabled. An interrupt is generated when the EOSMP bit is set.
0 = ADRDY interrupt disabled.

27.10.2.3 ADC Control Register (ADC_CR)

Address Offset: 0x08

	31	30	29	28	27	26	25	24
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8
R	0	0	0	0	0	0	0	0
W								
RESET	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	0	0	0	0	ADSTP	ADSTART	ADDIS	ADEN
W					rs	rs	rs	rs
RESET	0	0	0	0	0	0	0	0

=Writes have no effect and the access terminates without a transfer error exception.

Figure 27–14: ADC Control Register (ADC_CR)

©Levetop Electronics Co., Ltd.

Read: Anytime

Write: See each bit description

ADSTP — ADC stop conversion command
This bit is set by software to stop and discard an ongoing conversion (ADSTP Command). It is cleared by hardware when the conversion is effectively discarded and the ADC is ready to accept a new start conversion command.

1 = Write 1 to stop the ADC. Read 1 means that an ADSTP command is in progress.

0 = No ADC stop conversion command ongoing

Note:
Software is allowed to set ADSTP only when ADSTART=1 and ADDIS=0 (ADC is enabled and may be converting and there is no pending request to disable the ADC)

ADSTART — ADC start conversion command
This bit is set by software to start ADC conversion. Depending on the TRIGMODE configuration bits, a conversion either starts immediately (software trigger configuration) or once a hardware trigger event occurs (hardware trigger configuration). It is cleared by hardware:

- In single conversion mode when software trigger is selected: at the assertion of the End of Conversion Sequence (EOSEQ) flag.
- In discontinued conversion mode when software trigger is selected: at the assertion of the End of Conversion (EOC) flag.
- In all cases: after the execution of the ADSTP command, at the same time as the ADSTP bit is cleared by hardware.

1 = Write 1 to start the ADC. Read 1 means that the ADC is operating and may be converting.

0 = No ADC conversion is ongoing.

Note:
Software is allowed to set ADSTART only when ADEN=1 and ADDIS=0 (ADC is enabled and there is no pending request to disable the ADC)

ADDIS — ADC disable command
This bit is set by software to disable the ADC (ADDIS command) and put it into power-down state (OFF state). It is cleared by hardware once the ADC is effectively disabled (ADEN is also cleared by hardware at this time).
1 = Write 1 to disable the ADC. Read 1 means that an ADDIS command is in progress.
0 = No ADDIS command ongoing

Note:
Software is allowed to set ADDIS only when ADEN=1 and ADSTART=0 (which ensures that no conversion is ongoing)

ADEN — ADC enable command
This bit is set by software to enable the ADC. The ADC will be effectively ready to operate once the ADRDY flag has been set. It is cleared by hardware when the ADC is disabled, after the execution of the ADDIS command.
1 = Write 1 to enable the ADC.
0 = ADC is disabled (OFF state).

Note:
Software is allowed to set ADEN only when all bits of ADC_CR registers are 0 (ADSTP=0, ADSTART=0, ADDIS=0 and ADEN=0)

©Levetop Electronics Co., Ltd.

27.10.2.4 ADC Configuration Register 1 (ADC_CFGR1)

Address Offset: 0x0c

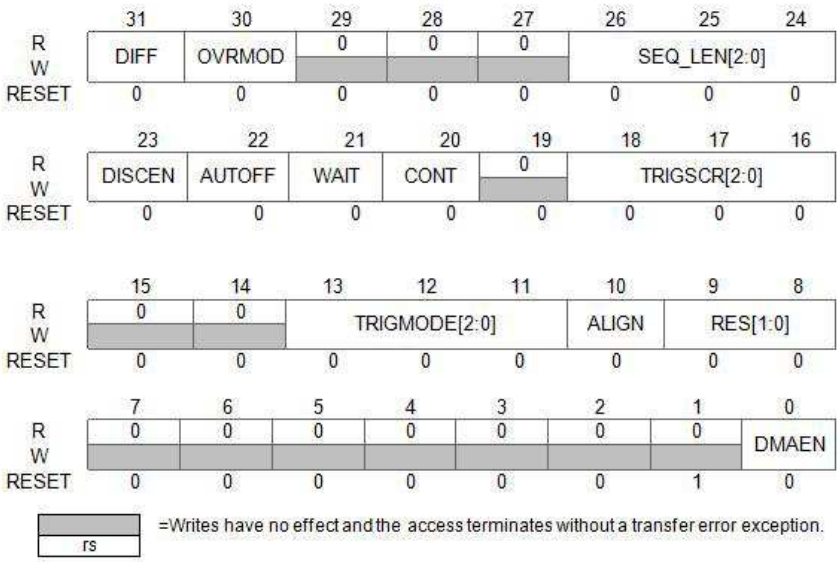


Figure 27–15: ADC Configuration Register 1 (ADC_CFGR1)

Read: Anytime

Write: Before ADC start

DIFF — Select differential-input
This bit determines that if the input is single-ended or differential-input.
1 = The analog input is differentially sampled.
0 = The analog input is single sampled

OVRMOD — Overrun management mode
This bit is set and cleared by software and configure the way data overruns are managed.
1 = ADC_DR register is overwritten with the last conversion result when an overrun is detected.
0 = ADC_DR register is preserved with the old data when an overrun is detected.

SEQ_LEN[2:0] — Sequence length
These bits define the length of sequence. The sequence length=SEQ_LEN+1. For example, SEQ_LEN=7 means sequence length is 8; SEQ_LEN=0 means sequence length is 1.

DISCEN — Discontinuous mode
This bit is set and cleared by software to enable/disable discontinuous mode.
1 = Discontinuous mode enabled
0 = Discontinuous mode disabled

AUTOFF — Auto-off mode
This bit is set and cleared by software to enable/disable auto-off mode.
1 = Auto-off mode enabled
0 = Auto-off mode disabled

©Levetop Electronics Co., Ltd.

WAIT — Wait conversion mode
This bit is set and cleared by software to enable/disable wait conversion mode.
1 = Wait conversion mode on
0 = Wait conversion mode off
CONT — Single / continuous conversion mode
This bit is set and cleared by software. If it is set, conversion takes place continuously until it is cleared.
1 = Continuous conversion mode
0 = Single conversion mode

TRIGSCR[2:0] — External trigger source
These bits are used to select which of 8 possible events can trigger conversions. See **Table 27–3**.

TRIGMOD[2:0] — Trigger mode select
These bits are used to select software trigger mode or external trigger polarity, see **Table 27–2**

ALIGN — Data alignment
This bit is set and cleared by software to select right or left alignment. Refer to **Figure 27–10**.
1 = Right alignment
0 = Left alignment

RES[1:0] — Data resolution
These bits are written by software to select the resolution of the conversion. Refer to **Figure 27–10.**

DMAEN — Direct memory access enable
This bit is set and cleared by software to enable the generation of DMA requests. This allows to use the DMA controller to manage automatically the converted data.
1 = DMA enabled
0 = DMA disabled

27.10.2.5 ADC Configuration Register 2 (ADC_CFGR2)

Address Offset: 0x10

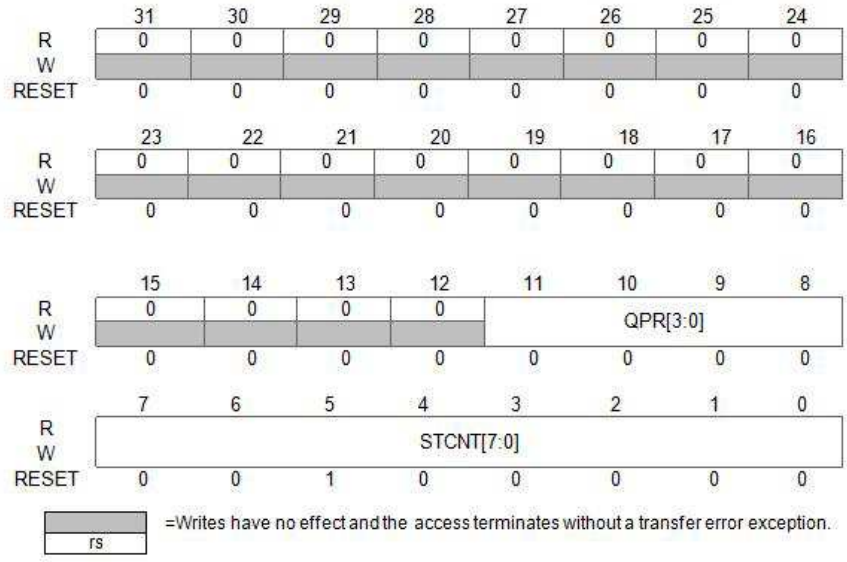


Figure 27–16: ADC Configuration Register 2 (ADC_CFGR2)

©Levetop Electronics Co., Ltd.

Read: Anytime

Write: Before ADC enable

QPR[3:0] — Prescaler Clock Divider Bits
These bits select the system clock divisor to generate the QADC clock as follows:

$$F_{QCLK} = F_{sys_QCLK} / (QPR[3:0] + 1)$$

Where:

$$0 <= QPR[3:0] <= 15.$$

STCNT[7:0] — ADC startup counter bits
The ADC needs a stabilization time of tSTAB(~2us) before it starts converting accurately. This time is calculated by counting QCLK cycles until the internal counter reaches the STCNT[7:0]. So user should set these bits before ADC enable. For example, if the QCLK = 16MHz, then should set the STCNT[7:0] = 2000/(1000/16)=32.

27.10.2.6 ADC Sampling Time Register (ADC_SMPR)

Address Offset: 0x08

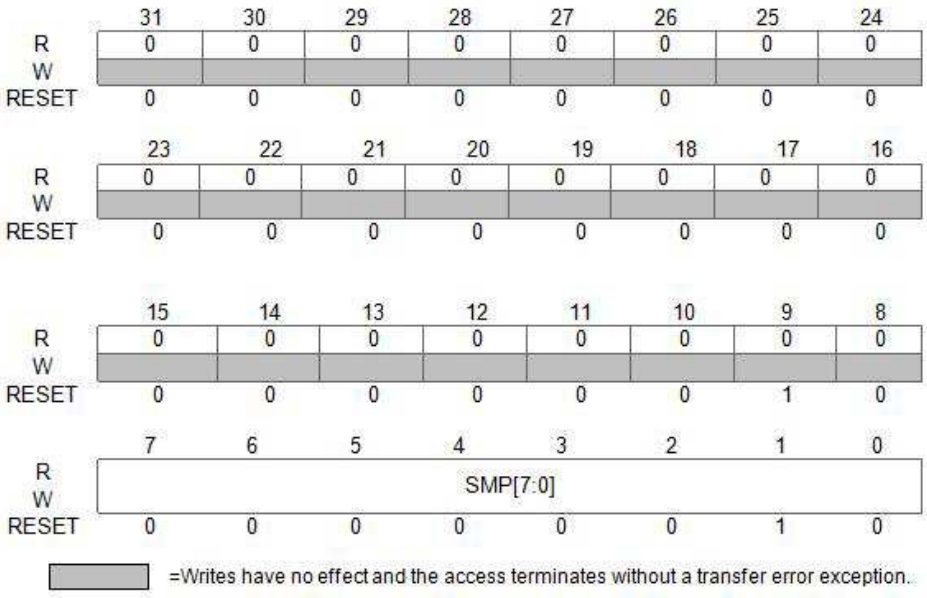


Figure 27–17: ADC Sampling Time Register (ADC_SMPR)

Read: Anytime

Write: Before ADC start

SMP[7:0] — Sampling time selection
These bits are written by software to select the sampling time that applies to all channels. The sample time is calculated as (SMP[7:0]+2) QCLKs Example : SMP[7:0] = 0x2 means the sample time is 4 QCLKs

©Levetop Electronics Co., Ltd.

27.10.2.7 ADC Watchdog Register (ADC_WDG)

Address Offset: 0x18

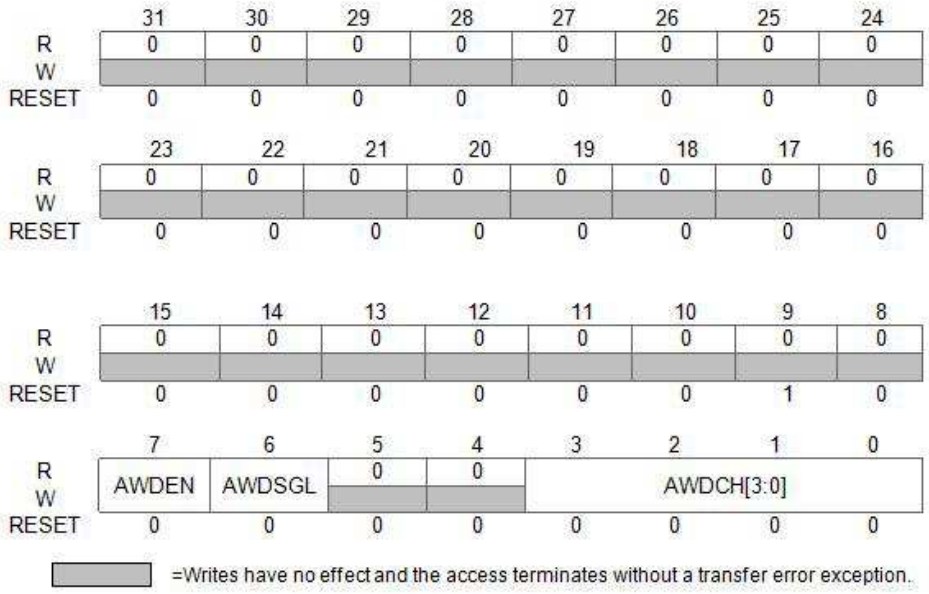


Figure 27–18: ADC Watch Dog Register (ADC_WDG)

Read: Anytime

Write: Before ADC start

AWDEN — Analog watchdog enable
This bit is set and cleared by software.
1 = Analog watchdog enabled
0 = Analog watchdog disabled

AWDSGL — Enable the watchdog on a single channel or on all channels
This bit is set and cleared by software to enable the analog watchdog on the channel identified by the AWDCH[4:0] bits or on all the channels
1 = Analog watchdog enabled on a single channel
0 = Analog watchdog enabled on all channels

AWDCH[3:0] — Analog watchdog channel selection
These bits are set and cleared by software. They select the input channel to be guarded by the analog watchdog.

- 0000: ADC analog input Channel 0 monitored by AWD
- 0001: ADC analog input Channel 1 monitored by AWD
- :
- :
- :
- 0111: ADC analog input Channel 7 monitored by AWD
- 1111: Temperature sensor monitored by AWD
- other values: Reserved, must not be used

©Levetop Electronics Co., Ltd.

27.10.2.8 ADC Watchdog Threshold Register (ADC_TR)

Address Offset: 0x1c

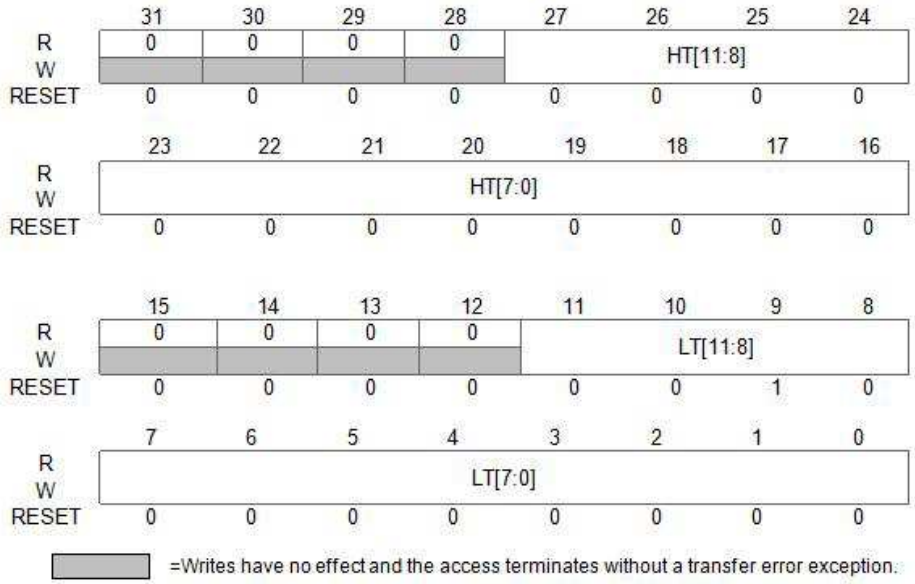


Figure 27–19: ADC Watchdog Threshold Register (ADC_TR)

Read: Anytime

Write: Before ADC start

HT[11:0] — Analog watchdog higher threshold
These bits are written by software to define the higher threshold for the analog watchdog.

LT[11:0] — Analog watchdog lower threshold
These bits are written by software to define the lower threshold for the analog watchdog.

©Levetop Electronics Co., Ltd.

27.10.2.9 ADC Channel Selection Register (ADC_CHSELR1, ADC_CHSELR2)

Address Offset: 0x2c

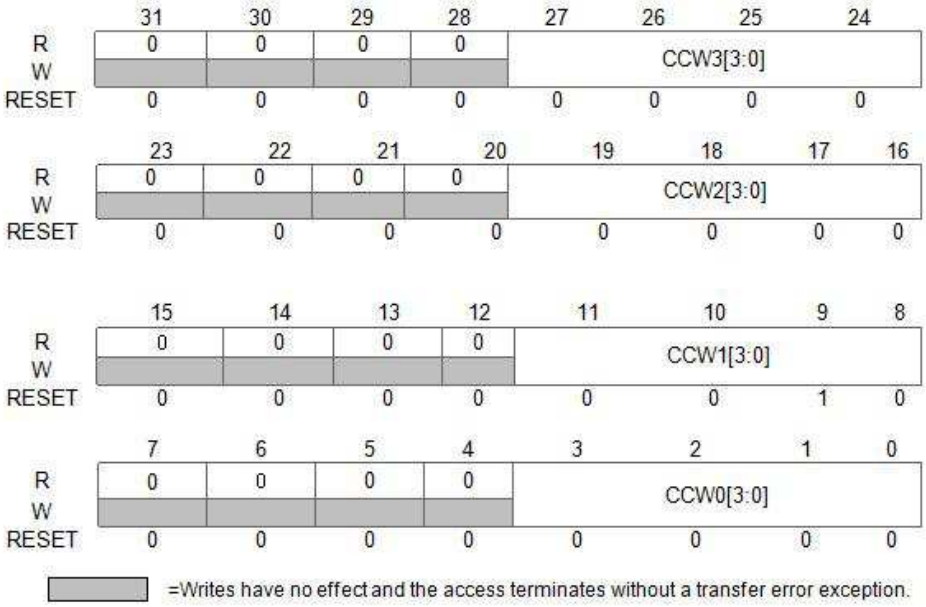


Figure 27–20: ADC Channel Selection Register 1(ADC_CHSELR1)

Address Offset: 0x30

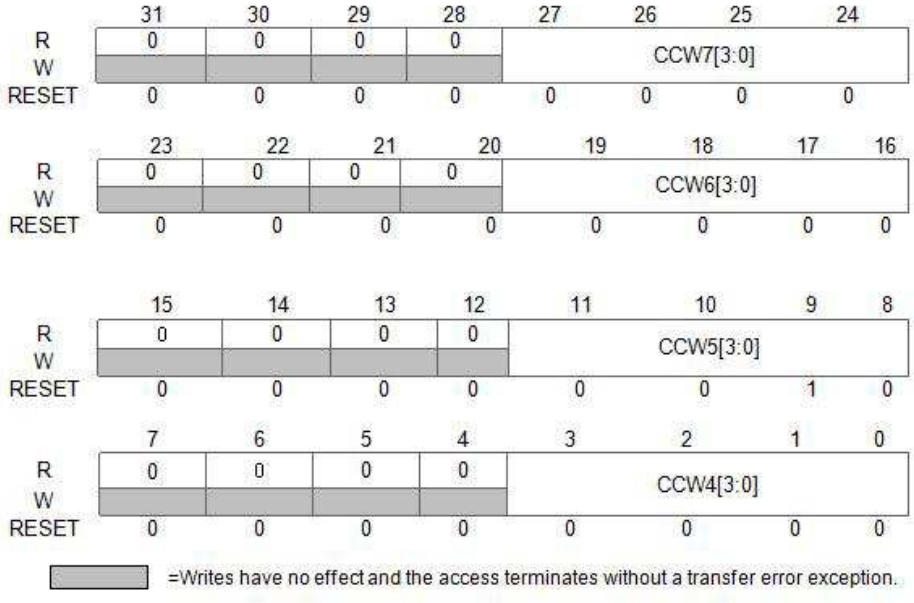


Figure 27–21: ADC Channel Selection Register 2(ADC_CHSELR2)

Read: Anytime

Write: Before ADC start

CCWi[3:0] — The number “I” conversion select channel, refer to **Table 27–1**.

©Levetop Electronics Co., Ltd.

27.10.2.10 ADC FIFO Access Register (ADC_FIFO)

Address Offset: 0x4c

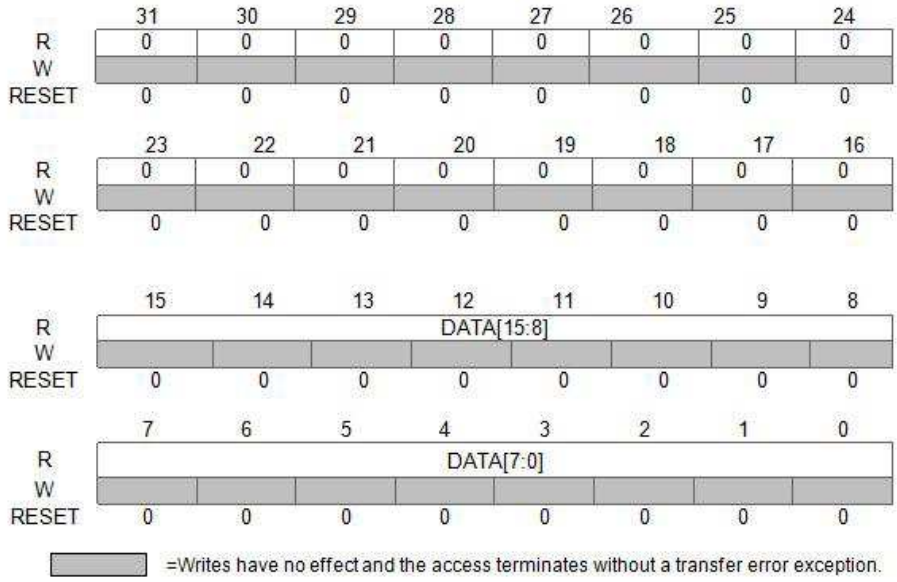


Figure 27–22: ADC FIFO Access Register (ADC_FIFO)

Read: Anytime

Write: Never

DATA[15:0]: Converted data
Refer to **Figure 27–10.**

©Levetop Electronics Co., Ltd.

28. Mechanical Specifications

28.1 LT32U02(QFN48) Mechanical Drawing

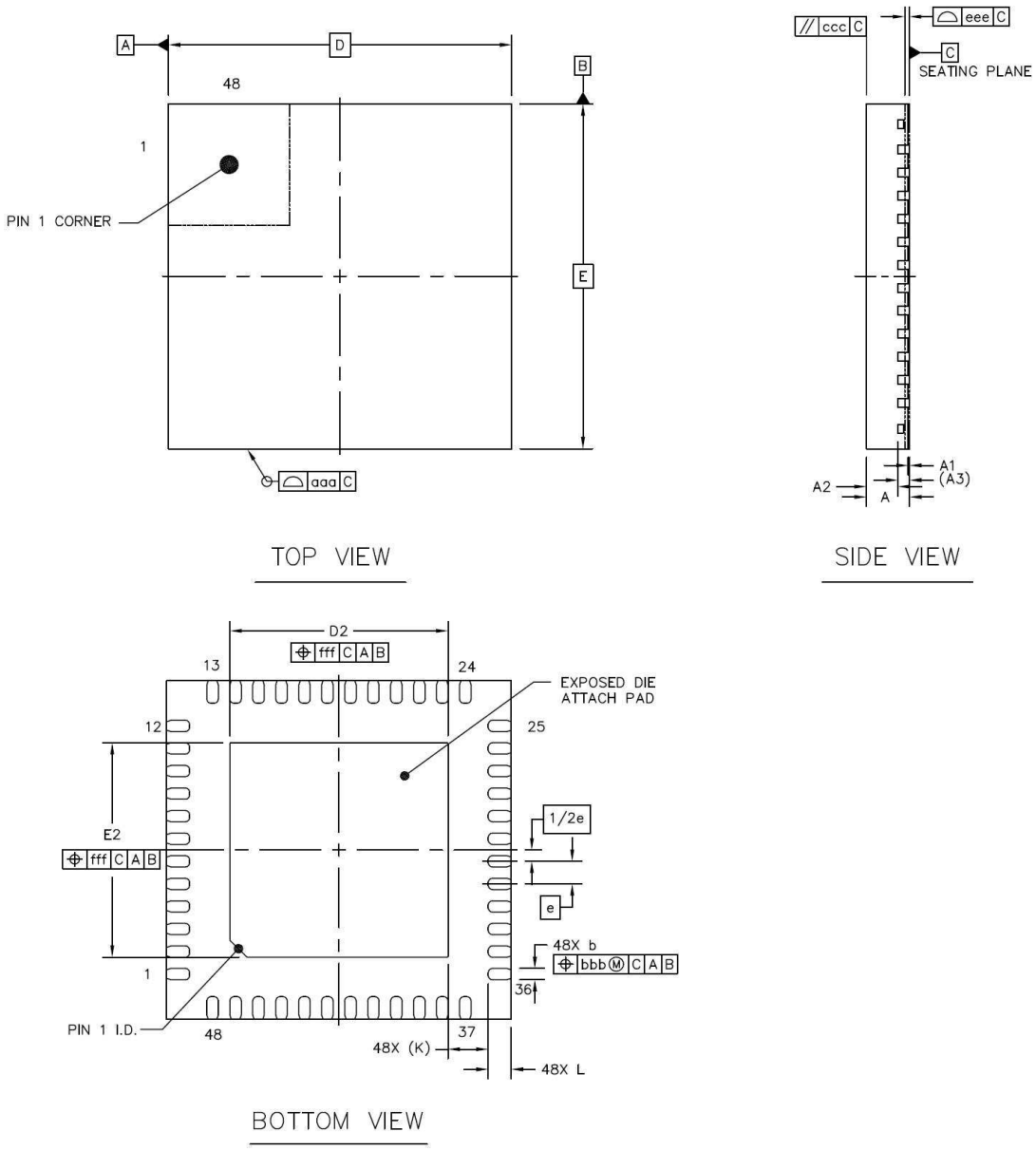


Figure 28–1: LT32U02 Dimension Diagram

©Levetop Electronics Co., Ltd.

Table 28–1: LT32U02 Dimension Table

		SYMBOL	MIN	NOM	MAX
TOTAL THICKNESS		A	0.7	0.75	0.8
STAND OFF		A1	0	0.02	0.05
MOLD THICKNESS		A2	---	0.55	---
L/F THICKNESS		A3		0.203 REF	
LEAD WIDTH		b	0.15	0.2	0.25
BODY SIZE	X	D		6 BSC	
	Y	E		6 BSC	
LEAD PITCH		e		0.4 BSC	
EP SIZE	X	D2	3.7	3.8	3.9
	Y	E2	3.7	3.8	3.9
LEAD LENGTH		L	0.3	0.4	0.5
LEAD TIP TO EXPOSED PAD EDGE		K		0.7 REF	
PACKAGE EDGE TOLERANCE		aaa		0.1	
MOLD FLATNESS		ccc		0.1	
COPLANARITY		eee		0.08	
LEAD OFFSET		bbb		0.07	
EXPOSED PAD OFFSET		fff		0.1	

©Levetop Electronics Co., Ltd.

28.2 LT32A02(QFN32) Mechanical Drawing

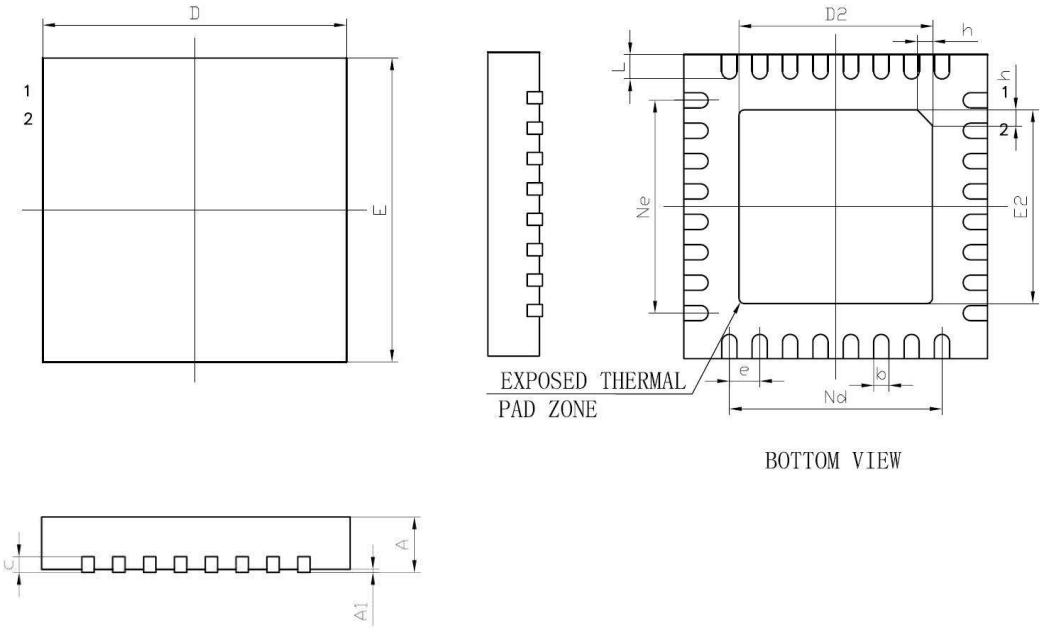


Figure 28–2: LT32A02 Dimension Diagram

Table 28–2: LT32A02 Dimension Table

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	—	0.02	0.05
b	0.18	0.25	0.30
c	0.18	0.20	0.25
D	4.90	5.00	5.10
D2	3.10	3.20	3.30
e	0.50BSC		
Ne	3.50BSC		
Nd	3.50BSC		
E	4.90	5.00	5.10
E2	3.10	3.20	3.30
L	0.35	0.40	0.45
h	0.25	0.30	0.35
L/F载体尺寸	146X146		

©Levetop Electronics Co., Ltd.

29. Absolute Maximum Ratings

Maximum ratings are the extreme limits to which the MCU can be exposed without permanently damaging it. See **Table 29–1**. The MCU contains circuitry to protect the inputs against damage from high static voltages; however, do not apply voltages higher than those shown in the table. Connect unused inputs to the appropriate voltage level, either V_{SSH} or V_{DDH}. This device is not guaranteed to operate properly at the maximum ratings. Refer to “DC Electrical Specifications” for guaranteed operating conditions.

Table 29–1: Absolute Maximum Ratings

Num	Rating	Symbol	Value	Unit
1	Supply Voltage	V_{ddh}	-0.5 to +5.5	V
2	Supply Voltage	V_{ddh1}	-0.5 to +5.5	V
3	Input Voltage[1]	AVDD	-0.5 to +5.5	V
4	Operating Temperature Range	TOPT	-40 to +85	^{O}C

30. DC Electrical Specifications

Table 30–1: DC Electrical Specifications

Parameter	Symbol	Min	Typical	Max	Unit
Supply Voltage	V_{ddh}	2.0 (LT32A02) 2.8 (LT32U02)	3.3/5.0	5.5	V
Input High Voltage	V_{IH}	0.65*V_{ddh}	-	V_{ddh}	V
Input Low Voltage	V_{IL}	V_{SS}	-	0.35*V_{ddh}	V
Output High Voltage	V_{OH}	0.8*V_{ddh}	—	—	V
Output Low Voltage	V_{OL}	—	—	0.1*V_{ddh}	V
Pull-Up Resistor	R_{PU}	20		100	Kohm
Pull-Down Resistor	R_{PU}	20		100	Kohm
Low Level Output Current @ V_{OL}=0.1*V_{ddh}	I_{OL}			6	mA
High Level Output Current @ V_{OH}=0.8*V_{ddh}	I_{OH}			6	mA

This document describes the functionality of the C0 microprocessor, which is based on M*Core instruction set/architecture and designed for extremely low-power and cost-sensitive embedded control applications.

For the smaller size and power dissipation, C0 is built on a new 3-stage pipeline von Neumann architecture.

C0 also integrates an EIC(embedded interrupt controller) to reduce system area.

©Levetop Electronics Co., Ltd.

Table 30–2: Power Consumption

Parameter		Min	Typical	Max	Unit
VDD = 3.3V, System Clock = 72MHZ					
3.3V LDO Enable	ADC, Cache → On	--	20	--	mA
	ADC, Cache → Off	--	16	--	
	USB → On	--	23	--	
3.3V LDO Disable	ADC, Cache → On	--	20	--	mA
	ADC, Cache → Off	--	16	--	
	USB → On	--	18	--	
Sleep Mode (Ext. Interrupt Enable)		--	5	--	uA
Sleep Mode (WDT1 Interrupt Enable)		--	7	--	uA
VDD = 5.0V, System Clock = 72MHZ					
3.3V LDO Enable	ADC, Cache → On	--	21	--	mA
	ADC, Cache → Off	--	17	--	
	USB → On	--	24	--	
3.3V LDO Disable	ADC, Cache → On	--	21	--	mA
	ADC, Cache → Off	--	17	--	
	USB → On	--	19	--	
Sleep Mode (Ext. Interrupt Enable)		--	6	--	uA
Sleep Mode (WDT1 Interrupt Enable)		--	12	--	uA

©Levetop Electronics Co., Ltd.

31. Revision

Table 31–1: Revision

Version	Date	Description
V1.0	2018/08/08	Preliminary Release

32. Copyright

This document is the copyright of Levetop Electronics Co., Ltd. No part of this document may be reproduced or duplicated in any form or by any means without the prior permission of Levetop. The information appearing in this Data Sheet is believed to be accurate at the time of publication. However, Levetop assumes no responsibility arising from the use of the specifications described. The applications mentioned herein are used solely for the purpose of illustration and Levetop makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Levetop's products are not authorized for use as critical components in life support devices or systems. Levetop reserves the right to alter its products without prior notification. For the most up-to-date information, please visit our web site at http://www.levetop.cn.