Confidentiality level: External use

UM3241F / UM32417F User Manual

V1.0.0

Unicmicro

Unicmicro (Guangzhou) Co., Ltd.

http://www.unicmicro.com/

Copyright & Disclaimer

The information herein is the exclusive property of Unicmicro (Guangzhou) Co., Ltd. (hereinafter referred to as Unicmicro). No part of this document may be copied, transmitted or transcribed without prior written permission of Unicmicro. Unicmicro makes no warranty, either express, implied or statutory, regarding the information set forth herein, and shall not be liable for any direct or indirect loss in connection with or arising out of the performance or use of this document or any information on the products herein. In addition, the product specifications and information given in this document are for reference only, and are subject to change without prior notice.

1. The information regarding circuit, software, etc. provided in this document is intended only to illustrate examples of operation and application of semiconductor products. Users are solely responsible for applying such information in the design of their devices. Unicmicro shall not be liable to recipient or any third party for any damages arising out of the furnishing, performance or use of the aforementioned information.
2. In the course of preparing the information herein, Unicmicro has tried to exercise reasonable care, however, does not guarantee that such information is accurate. Unicmicro assumes no liability for any consequences resulting from errors or omissions in the information herein.
3. Unicmicro will not take responsibility for any infringement of patents, copyrights or other intellectual property rights of third parties resulting from the use of Unicmicro products or technical information herein. Nothing contained in this document shall be construed as granting any license or authorization, express, implied or otherwise, to the patents, copyrights or other intellectual property rights of Unicmicro or others.
4. The application of Unicmicro products referred in this document shall be within the range specified by Unicmicro, especially within the ranges of maxing ratings, supply operating voltage, thermal radiation characteristics, installation conditions, and other product characteristics. Unicmicro shall not be responsible for any failure or loss arising from the use of its products outside the above specified ranges.
5. Although Unicmicro has been committed to improving the quality and reliability of its products, semiconductor products with their own characteristics have a certain incidence of failure and the possibility of failure under certain conditions of use. In addition, none of Unicmicro products are designed to be radiation-proof, so to avoid potential damage from product failure or fire resulting in personal accidents, injuries or damages, be sure to take protective measures such as hardware and software safety design (including but not limited to redundant design, fire control and failure prevention, etc.), proper aging treatment or other appropriate measures, etc.

Contents

1 Documentation Conventions 1
1.1 List of Abbreviations for Registers 1
1.2 Glossary 1
2 Product Introduction 3
2.1 Overview 3
2.2 Main Features 4
3 Memory and Bus Architecture 9
3.1 System Architecture 9
3.2 Bus Architecture Diagram 10
3.3 Memory Mapping 10
4 Memory System 15
4.1 FLASH 15
4.1.1 Overview 15
4.1.2 Main Features 15
4.1.3 Functional Description 16
4.1.4 Register Description 19
4.1.5 Operation Procedure 25
4.2 SRAM 27
4.2.1 Overview 27
4.2.2 Main Features 28
4.3 CACHE 28
4.3.1 Register Description 28
4.3.2 Use Process 30
5 Power Management Unit (PMU) 31
5.1 Chip Power Supply 31
5.2 Internal PMU Module Function 32
5.2.1 Introduction to Internal PMU Module 32
5.2.2 Run Mode 34
5.2.3 Sleep Mode 35
5.2.4 Stop Mode 35
5.2.5 Standby Mode 35
5.2.6 DeepStandby Mode 35
5.3 Register Description 35
5.3.1 PMU Mode Register (PMU_MR) 36
5.3.2 Power-down Wakeup Control Register (PMU_PDWKCR) 37

Unicmicro

5.3.3 Power-up Stabilization Time Configuration Register (PMU_PUSTCR) 41
5.3.4 PDR/BOR/LVD Configuration Register (PMU_VDCR) 41
5.3.5 System Auxiliary Status Register (PMU_SASR) 44
5.3.6 XTL Configuration Register (PMU_XTLCR) 46
5.3.7 RCL Configuration Register (PMU_RCLCR) 47
5.3.8 Function Clock Control Register (PMU_FCCR) 47
5.3.9 Function Reset Control Register (PMU_FRCR) 48
5.3.10 Configuration Protection Register (PMU_CPR) 49

6 Reset and Clock Module (RCM) 50
6.1 Clock Logic 50
6.2 Reset Logic 51
6.2.1 Power Reset 52
6.2.2 System Reset 52
6.3 Register Description 54
6.3.1 Clock Control Register 0 (RCM_CR0) 55
6.3.2 PLL0 Configuration Register 0 (RCM_PLL0CFGR0) 58
6.3.3 PLL0 Configuration Register 1 (RCM_PLL0CFGR1) 59
6.3.4 PLL0 Configuration Register 2 (RCM_PLL0CFGR2) 59
6.3.5 PLL1 Configuration Register 0 (RCM_PLL1CFGR0) 60
6.3.6 PLL1 Configuration Register 1 (RCM_PLL1CFGR1) 60
6.3.7 PLL1 Configuration Register 2 (RCM_PLL1CFGR2) 61
6.3.8 PLL Stabilization Time Setting Register (RCM_PLLTSR) 61
6.3.9 Clock Configuration Register 0 (RCM_CFGR0) 62
6.3.10 Clock Configuration Register 1 (RCM_CFGR1) 65
6.3.11 Clock Interrupt Flag Register (RCM_CIFR) 68
6.3.12 Clock Interrupt Enable Register (RCM_CIER) 69
6.3.13 AHB Peripheral Clock Enable Register (RCM_AHBCKENR) 70
6.3.14 AHB0 Peripheral Clock Enable Register (RCM_AHB0CKENR) 71
6.3.15 AHB1 Peripheral Clock Enable Register (RCM_AHB1CKENR) 71
6.3.16 APB0 Peripheral Clock Enable Register (RCM_APB0CKENR) 72
6.3.17 APB1 Peripheral Clock Enable Register (RCM_APB1CKENR) 73
6.3.18 APB2 Peripheral Clock Enable Register (RCM_APB2CKENR) 75
6.3.19 APB3 Peripheral Clock Enable Register (RCM_APB3CKENR) 76
6.3.20 AHB Peripheral Reset Enable Register (RCM_AHBRSTR) 77
6.3.21 AHB0 Peripheral Reset Enable Register (RCM_AHB0RSTR) 77
6.3.22 AHB1 Peripheral Reset Enable Register (RCM_AHB1RSTR) 78
6.3.23 APB0 Peripheral Reset Enable Register (RCM_APB0RSTR) 80
6.3.24 APB1 Peripheral Reset Enable Register (RCM_APB1RSTR) 80

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.25 APB2 Peripheral Reset Enable Register (RCM_APB2RSTR) 82
6.3.26 APB3 Peripheral Reset Enable Register (RCM_APB3RSTR) 84
6.3.27 Software Reset Register (RCM_SOFTRSTR) 85
6.3.28 Reset Flag Register (RCM_RFR) 85
6.3.29 External Reset Filter Control Register (RCM_EXRSTFER) 87
6.3.30 RCM Configuration Protection Register (RCM_RCMPR) 87
7 GPIO 88
7.1 Overview 88
7.2 Main Features 88
7.3 Functional Description 88
7.3.1 GPIO Input and Output 88
7.3.2 GPIO Interrupt Generation 88
7.4 Register Description 88
7.4.1 Function Mode Register (GPIOx_MODE) 90
7.4.2 Output Set Register (GPIO_SET) 90
7.4.3 Output Clear Register (GPIO_CLR) 90
7.4.4 Output Data Register (GPIO_ODATA) 91
7.4.5 Input Data Register (GPIO_IDATA) 91
7.4.6 Interrupt Enable Register (GPIO_IEN) 92
7.4.7 Interrupt Trigger Mode Register (GPIO_IS) 92
7.4.8 Interrupt Edge Trigger Setting Register (GPIO_IBE) 92
7.4.9 Interrupt Level Trigger Setting Register (GPIO_IEV) 93
7.4.10 Interrupt Clear Register (GPIO_IC) 93
7.4.11 Raw Interrupt Status Register (GPIO_RIS) 93
7.4.12 Masked Interrupt Status Register (GPIO_MIS) 94
7.4.13 Filter Enable Register (GPIO_DBEN) 94
7.4.14 Filter Length Register (GPIO_DBL) 95
7.4.15 Configuration Lock Register (GPIO_LOCK) 95
7.4.16 Input Mode Register (GPIO_IM) 95
7.4.17 Pull-up/down Register (GPIO_PULL) 96
7.4.18 Driving Speed Register (GPIO_SR) 96
7.4.19 Driving Capability Register (GPIO_DS) 96
7.4.20 Alternate Function Low Register (GPIO_AFL) 97
7.4.21 Alternate Function High Register (GPIO_AFH) 97
7.5 Operation Procedure 98
7.5.1 Input 98
7.5.2 Output 98
7.5.3 Interrupt Trigger Mode 98

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8 Nested Vectored Interrupt Controller (NVIC) 100

9 System Configuration Controller (SYSCFG) 104
9.1 Register Description 104
9.1.1 EMAC Mode Control Register (SYSCFG_EMACMR) 104
9.1.2 ADC External Trigger Selection Register (SYSCFG_ADCETSR) 105
9.1.3 TIM Break Control Register (SYSCFG_TIMCFGR) 106
9.1.4 External Interrupt Configuration Register 0 (SYSCFG_EXTICR0) 107
9.1.5 External Interrupt Configuration Register 1 (SYSCFG_EXTICR1) 110
9.1.6 External Interrupt Configuration Register 2 (SYSCFG_EXTICR2) 113
9.1.7 External Interrupt Configuration Register 3 (SYSCFG_EXTICR3) 115
9.1.8 External Interrupt Wakeup Configuration Register (SYSCFG_EXTIWR) 118
9.1.9 MISC Control Register (SYSCFG_MISCCR) 122
9.1.10 SysTick Calibration Reference Value Register (SYSCFG_SCRVR) 122

10 Cyclic Redundancy Check Calculation Unit (CRC) 123
10.1 Overview 123
10.2 Main Features 123
10.3 Functional Description 123
10.3.1 CRC Calculation 123
10.3.2 Common CRC Formats 124
10.4 Register Description 124
10.4.1 Data Register (CRC_DATA) 124
10.4.2 Configuration Register (CRC_CFG) 125
10.4.3 Initial Value Register (CRC_INIT) 126
10.5 Operation Procedure 126

11 Direct Memory Access Controller (DMA) 127
11.1 Overview 127
11.2 Main Features 127
11.3 Functional Description 128
11.3.1 Flow Control 128
11.3.2 Handshake Signal 128
11.4 Register Description 129
11.4.1 Source Address Register (DMA_SARx) 131
11.4.2 Destination Address Register (DMA_DARx) 131
11.4.3 Control Register (DMA_CTLx) 132
11.4.4 Control Register (DMA_CTLHx) 133
11.4.5 Configuration Register (DMA_CFGx) 134
11.4.6 Configuration Register (DMA_CFGHx) 135

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11.4.7 Raw Transfer Interrupt Register (DMA_RAWTFR) 136
11.4.8 Raw Block Transfer Interrupt Register (DMA_RAWBLOCK) 137
11.4.9 Raw Source Transfer Interrupt Register (DMA_RAWSRCTRAN) 137
11.4.10 Raw Destination Transfer Interrupt Register (DMA_RAWDSTTRAN) 138
11.4.11 Raw Error Interrupt Register (DMA_RAWERR) 138
11.4.12 Transfer Interrupt Status Register (DMA_STATUSTFR) 139
11.4.13 Block Transfer Interrupt Status Register (DMA_STATUSBLOCK) 139
11.4.14 Source Transfer Interrupt Status Register (DMA_STATUSSRCTRAN) 140
11.4.15 Destination Transfer Interrupt Status Register (DMA_STATUSDSTRAN) 140
11.4.16 Error Interrupt Status Register (DMA_STATUSERR) 141
11.4.17 Transfer Interrupt Mask Register (DMA_MASKTFR) 142
11.4.18 Block Transfer Interrupt Mask Register (DMA_MASKBLOCK) 142
11.4.19 Source Transfer Interrupt Mask Register (DMA_MASKSRCTRAN) 143
11.4.20 Destination Transfer Interrupt Mask Register (DMA_MASKDSTTRAN) ... 143
11.4.21 Error Interrupt Mask Register (DMA_MASKERR) 144
11.4.22 Transfer Interrupt Clear Register (DMA_CLEARTFR) 144
11.4.23 Block Transfer Interrupt Clear Register (DMA_CLEARBLOCK) 145
11.4.24 Source Transfer Interrupt Clear Register (DMA_CLEARSRCTRAN) 145
11.4.25 Destination Transfer Interrupt Clear Register (DMA_CLEARDSTTRAN) .. 146
11.4.26 Error Interrupt Clear Register (DMA_CLEARERR) 146
11.4.27 Interrupt Status Register (DMA_STATUSINT) 146
11.4.28 Destination Transfer Request Signal Register (DMA_REQSRCREG) 147
11.4.29 Destination Transfer Request Signal Register (DMA_REQDSTREG) 147
11.4.30 Source Transfer Single Signal Register (DMA_SGLREQSRCREG) 148
11.4.31 Destination Transfer Single Signal Register (DMA_SGLREQDSTREG) 148
11.4.32 Source Transfer Last Signal Register (DMA_LSTSRCRE) 149
11.4.33 Destination Transfer Last Signal Register (DMA_LSTDSTREG) 149
11.4.34 DMA Module Enable Register (DMA_CFGREG) 150
11.4.35 Channel Enable Register (DMA_CHENREG) 150
11.5 Operation Procedure 151
11.5.1 Basic Hardware Flow Control 151
11.5.2 Software Flow Control 152

12 Digital Camera Interface (DCMI) 153
12.1 Overview 153
12.2 Main Features 153
12.3 Pin Description 153
12.4 Functional Description 154
12.4.1 Data Capture 154

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.4.2 Hardware Synchronization Mode 155
12.4.3 JPEG Mode 156
12.4.4 Embedded Code Synchronization Mode 156
12.4.5 Window Cropping 156
12.4.6 Single-frame Capture 157
12.5 Register 157
12.5.1 Control Register (DCMI_CTRL) 157
12.5.2 Status Register (DCMI_STATUS) 159
12.5.3 Raw Interrupt Register (DCMI_INTRAW) 160
12.5.4 Interrupt Enable Register (DCMI_INTEN) 160
12.5.5 Interrupt Status Register (DCMI_INTMASKED) 161
12.5.6 Interrupt Clear Register (DCMI_INTCLR) 161
12.5.7 Embedded Code Register (DCMI_ESC) 162
12.5.8 Embedded Code Mask Register (DCMI_ESCMASK) 162
12.5.9 Cropping Window Starting Point Register (DCMI_CROPSTART) 163
12.5.10 Cropping Window Size Register (DCMI_CROPSIZE) 163
12.5.11 Data Register (DCMI_DATA) 164
12.5.12 Data Output Control Register (DCMI_MSTCTRL) 164
12.5.13 Data Output Address Register (DCMI_MSTADR) 165
12.6 Operation Procedure 165
12.6.1 Capturing Data Using DCMI and Saving Data to Memory 165
12.6.2 Capturing Data Using DCMI and Transferring It Directly to Destination Address 167
12.6.3 Capturing Data Using DCMI and Transferring It to Destination Address via DMA 168
13 External Memory Controller (EMC) 171
13.1 Overview 171
13.2 Main Features 171
13.3 Pin Description 171
13.4 Functional Description 173
13.4.1 Chip Select 173
13.4.2 EBn Signal Timing 173
13.4.3 Byte Enable Pin 175
13.4.4 TFT_LCD Control 175
13.5 Register Description 176
13.5.1 Chip Select Control Register (EMC_CSCR0) 177
13.5.2 TFT Mode Channel Register (EMC_TTS) 178
13.5.3 TFT Command Channel Register (EMC_CMD) 178

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

13.5.4 TFT Data Channel Register (EMC_DATA) 179
13.5.5 Segment Register (EMC_SEG0) 179
13.6 Operation Procedure 180
13.6.1 Off-chip Memory Read Operation 180
13.6.2 Off-chip Memory Write Operation 180
13.6.3 TFT_LCD Operation 181
14 Hardware Acceleration Co-processor (CORDIC) 182
14.1 Overview 182
14.2 Main Features 182
14.3 Functional Description 182
14.3.1 Data Format and Input/Output 182
14.3.2 Function Mode 0: $\boldsymbol{m \cdot \sin\theta}$ / $\boldsymbol{m \cdot \cos\theta}$ 183
14.3.3 Function Mode 1: Atan2(y, x) / $\sqrt{(x2 + y2)}$ 184
14.3.4 Function Mode 2: $\boldsymbol{y \cdot x}$ 185
14.3.5 Function Mode 3: $\boldsymbol{y/x}$ 185
14.3.6 Function Mode 4: $\boldsymbol{\sinh w}$ / $\boldsymbol{\cosh w}$ 185
14.3.7 Function Mode 5: $\boldsymbol{\tanh - 1(y/x)}$ 186
14.3.8 Function Mode 6: ln($\boldsymbol{x}$) 186
14.3.9 Function Mode 7: $\boldsymbol{\mathrm{Sqr(x)}}$ 187
14.4 Register Description 188
14.4.1 Control Register (CORDIC_CTRL) 189
14.4.2 Data Input Register 1 (CORDIC_DIN1) 191
14.4.3 Data Input Register 2 (CORDIC_DIN2) 191
14.4.4 Data Output Register 1 (CORDIC_DOUT1) 191
14.4.5 Data Output Register 2 (CORDIC_DOUT2) 192
14.5 Operation Procedure 192
14.6 Calculation Example 192
15 Advanced Encryption and Decryption Algorithm Accelerator (AES) 194
15.1 Overview 194
15.2 Main Features 194
15.3 Functional Description 195
15.3.1 Encryption and Decryption 195
15.4 Register Description 195
15.4.1 Message Control Register (AES_MSGCFG) 196
15.4.2 Key Control Register (AES_CTXCFG) 197
15.4.3 Message Length Register (AES_MSGTOTALBYTES) 198
15.4.4 Additional Message Length Register (AES_MSGAADBYTES) 198

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.5 GCM Mode Additional Message Length Register (AES_GCMAADINFO) . 199
15.4.6 Key Selection Register (AES_CTXKEYSEL) 199
15.4.7 Key Register 0 (AES_CTXKEY0) 199
15.4.8 Key Register 1 (AES_CTXKEY1) 200
15.4.9 Key Register 2 (AES_CTXKEY2) 200
15.4.10 Key Register 3 (AES_CTXKEY3) 200
15.4.11 Key Register 4 (AES_CTXKEY4) 200
15.4.12 Key Register 5 (AES_CTXKEY5) 200
15.4.13 Key Register 6 (AES_CTXKEY6) 201
15.4.14 Key Register 7 (AES_CTXKEY7) 201
15.4.15 CBC Mode Key Register 0 (AES_CTXCBCKEY0) 201
15.4.16 CBC Mode Key Register 1 (AES_CTXCBCKEY1) 201
15.4.17 CBC Mode Key Register 2 (AES_CTXCBCKEY2) 202
15.4.18 CBC Mode Key Register 3 (AES_CTXCBCKEY3) 202
15.4.19 CTR Mode Operator Register 0 (AES_CTXCTR0) 202
15.4.20 CTR Mode Operator Register 1 (AES_CTXCTR1) 202
15.4.21 CTR Mode Operator Register 2 (AES_CTXCTR2) 203
15.4.22 CTR Mode Operator Register 3 (AES_CTXCTR3) 203
15.4.23 Initialization Vector Register 0 (AES_CTXIV0) 203
15.4.24 Initialization Vector Register 1 (AES_CTXIV1) 203
15.4.25 Initialization Vector Register 2 (AES_CTXIV2) 204
15.4.26 Initialization Vector Register 3 (AES_CTXIV3) 204
15.4.27 Message Authentication Code Register 0 (AES_CTXMAC0) 204
15.4.28 Message Authentication Code Register 1 (AES_CTXMAC1) 204
15.4.29 Message Authentication Code Register 2 (AES_CTXMAC2) 205
15.4.30 Message Authentication Code Register 3 (AES_CTXMAC3) 205
15.4.31 Input FIFO (AES_INGRESSFIFO) 205
15.4.32 Input FIFO Status Register (AES_INGFSTATUS) 205
15.4.33 Output FIFO (AES_ENGRESSFIFO) 206
15.4.34 Output FIFO Status Register (AES_ENGFSTATUS) 206
15.4.35 DMA Input Length Register (AES_DMAINGLEN) 206
15.4.36 DMA Input Burst Transfer Setting Register (AES_INGDBCFG) 206
15.4.37 DMA Output Length Register (AES_DMAENGLEN) 207
15.4.38 DMA Output Burst Transfer Setting Register (AES_ENGDBCFG) 208
15.4.39 Done Status Register (AES_DONESTATUS) 208
15.4.40 DMA Input Done Status Register (AES_INGDMADONE) 209
15.4.41 DMA Output Done Status Register (AES_ENGDMADONE) 209
15.5 Operation Procedure 210

Unicmicro

15.5.1 AES Encryption Model 210
15.5.2 AES Decryption Model 210

16 Secure Hash Algorithm Accelerator (SHA) 212
16.1 Hash Overview 212
16.2 Main Features 212
16.3 Functional Description 212
16.3.1 Encryption and Decryption 212
16.3.2 OTP Value Comparison 212
16.4 Register Description 213
16.4.1 Control Command Register (SHA_CMD) 214
16.4.2 Bit Mode Register (SHA_MODE) 214
16.4.3 Data Setting Register (SHA_MSGCFG) 214
16.4.4 Data Length Register (SHA_MSGTOTALBYTES) 215
16.4.5 Data Input Register (SHA_DATAIN) 215
16.4.6 ICV Read Register (SHA_ICVn) 216
16.4.7 Verification Result Register (SHA_VERIRESULT) 216
16.4.8 Interrupt Enable Register (SHA_IRQEN) 217
16.4.9 Interrupt Clear Register (SHA_IRQCLR) 217
16.4.10 Interrupt Status Register (SHA_IRQSTATUS) 217
16.4.11 DMA Burst Length Register (SHA_DMABURSTSIZE) 218
16.5 Operation Procedure 218
16.5.1 SHA Algorithm Application (Input Message via DMA) 218
16.5.2 SHA Algorithm Application (Input Message via CPU) 220

17 True Random Number Generator (RNG) 221
17.1 Overview 221
17.2 Main Features 221
17.3 Functional Description 221
17.3.1 RNG 221
17.4 Register Description 221
17.4.1 Data Register (RNG_DATA) 222
17.4.2 Random Seed Register (RNG_SEED) 222
17.4.3 Control Register (RNG_CR) 222
17.5 Operation Procedure 222

18 Advanced Control Timer (TIM1 & TIM8) 223
18.1 Overview 223
18.2 Main Features 223
18.3 System Block Diagram 224

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.4 Pin Description 225
18.5 Timer Interconnection 226
18.6 Functional Description 226
18.6.1 Time-base Unit 226
18.6.2 Counter Operation Mode 229
18.6.3 Repetition Counter 237
18.6.4 Preload Register 238
18.6.5 Counter Clock 239
18.6.6 Internal Trigger Signal (ITRx) 244
18.6.7 Capture/Compare Channels 244
18.6.8 Input Capture Mode 246
18.6.9 Forced Output Mode 249
18.6.10 Output Compare Mode 249
18.6.11 PWM Output 250
18.6.12 Complementary Output and Dead-time Insertion 252
18.6.13 Break Function 253
18.6.14 6-step PWM Generation 255
18.6.15 One-pulse Output Mode 256
18.6.16 Clearing OCxREF Signal on External Event 257
18.6.17 Encoder Interface Mode 258
18.6.18 TIM slave mode: 259
18.6.19 Timer Synchronization 263
18.6.20 DMA Access 264
18.6.21 MA Burst 265
18.6.22 Input XOR Function 266
18.6.23 Debug Mode 266
18.7 Register Description 266
18.7.1 Control Register 1 (TIM_CR1) 267
18.7.2 Control Register 2 (TIM_CR2) 269
18.7.3 Slave Mode Control Register (TIM_SMCR) 272
18.7.4 DMA / Interrupt Enable Register (TIM_DIER) 275
18.7.5 Status Register (TIM_SR) 277
18.7.6 Event Generation Register (TIM_EGR) 279
18.7.7 Capture/Compare Register 1 (TIM_CCMR1) 281
18.7.8 Capture/Compare Register 2 (TIM_CCMR2) 286
18.7.9 Capture/Compare Enable Register (TIM_CCER) 292
18.7.10 Counter Register (TIM_CNT) 297
18.7.11 Prescaler Register (TIM_PSC) 297

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.12 Auto-reload Register (TIM_ARR) 297
18.7.13 Repetition Counter Register (TIM_RCR) 298
18.7.14 Capture/Compare Register 1 (TIM_CCR1) 298
18.7.15 Capture/Compare Register 2 (TIM_CCR2) 298
18.7.16 Capture/Compare Register 3 (TIM_CCR3) 299
18.7.17 Capture/Compare Register 4 (TIM_CCR4) 300
18.7.18 Break and Dead-time Register (TIM_BDTR) 300
18.7.19 DMA Control Register (TIM_DCR) 302
18.7.20 DMA Access Register (TIM_DMAR) 303
18.8 Operation Procedure 304
18.8.1 Counting Mode 304
18.8.2 PWM Mode 304
18.8.3 Input Capture Mode 305
18.8.4 Complementary Output and Dead-time Insertion 306
18.8.5 Break Function 307
18.8.6 Encoder Interface Mode 307
18.8.7 DMA Mode 308
19 General-purpose Timers (TIM2 ~ TIM5 & TIM9 ~ TIM14) 310
19.1 Overview 310
19.2 Main Features 310
19.3 System Block Diagram 311
19.4 Pin Description 311
19.5 Functional Description 314
19.5.1 Time-base Unit 314
19.5.2 Counter Operation Mode 316
19.5.3 Preload Register 323
19.5.4 Counter Clock 324
19.5.5 Internal Trigger Signal (ITRx) 329
19.5.6 Capture/Compare Channels 329
19.5.7 Input Capture Mode 331
19.5.8 Forced Output Mode 334
19.5.9 Output Compare Mode 334
19.5.10 PWM Output 335
19.5.11 One-pulse Output Mode 337
19.5.12 Clearing OCxREF Signal on External Event 339
19.5.13 Encoder Interface Mode 339
19.5.14 TIM slave mode: 341
19.5.15 Timer Synchronization 345

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.16 DMA Access 345
19.5.17 DMA Burst 346
19.5.18 Input XOR Function 347
19.5.19 Debug Mode 347
19.6 Register Description 347
19.6.1 Control Register 1 (TIM_CR1) 348
19.6.2 Control Register 2 (TIM_CR2) 350
19.6.3 Slave Mode Control Register (TIM_SMCR) 351
19.6.4 DMA / Interrupt Enable Register (TIM_DIER) 353
19.6.5 Status Register (TIM_SR) 355
19.6.6 Event Generation Register (TIM_EGR) 358
19.6.7 Capture/Compare Register 1 (TIM_CCMR1) 359
19.6.8 Capture/Compare Register 2 (TIM_CCMR2) 364
19.6.9 Capture/Compare Enable Register (TIM_CCER) 369
19.6.10 Counter Register (TIM_CNT) 371
19.6.11 Prescaler Register (TIM_PSC) 371
19.6.12 Auto-reload Register (TIM_ARR) 372
19.6.13 Capture/Compare Register 1 (TIM_CCR1) 372
19.6.14 Capture/Compare Register 2 (TIM_CCR2) 373
19.6.15 Capture/Compare Register 3 (TIM_CCR3) 373
19.6.16 Capture/Compare Register 4 (TIM_CCR4) 374
19.6.17 DMA Control Register (TIM_DCR) 374
19.6.18 DMA Access Register (TIM_DMAR) 375
19.7 Operation Procedure 375
19.7.1 Counting Mode 375
19.7.2 PWM Mode 376
19.7.3 Input Capture Mode 377
19.7.4 Encoder Interface Mode 378
19.7.5 DMA Mode 379
20 Basic Timer (TIM6 & TIM7) 381
20.1 Overview 381
20.2 Main Features 381
20.3 Functional Description 381
20.3.1 Time-base Unit 381
20.3.2 Counter Operation Mode 383
20.3.3 Preload Register 386
20.3.4 Counter Clock 387
20.3.5 Debug Mode 387

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20.4 Register Description 388
20.4.1 Control Register 1 (TIM_CR1) 388
20.4.2 Control Register 2 (TIM_CR2) 390
20.4.3 DMA / Interrupt Enable Register (TIM_DIER) 390
20.4.4 Status Register (TIM_SR) 391
20.4.5 Event Generation Register (TIM_EGR) 391
20.4.6 Counter Register (TIM_CNT) 392
20.4.7 Prescaler Register (TIM_PSC) 392
20.4.8 Auto-reload Register (TIM_ARR) 392
20.5 Operation Procedure 393
20.5.1 Counter Startup 393
21 Low-power Timer (LPTIM) 394
21.1 Overview 394
21.2 Main Features 394
21.3 System Block Diagram 395
21.4 Pin Description 396
21.5 Functional Description 396
21.5.1 General-purpose Timer 396
21.5.2 Pulse-trigger Counting 396
21.5.3 External Asynchronous Pulse Counting 397
21.5.4 Timeout Mode 397
21.5.5 Counting Mode 397
21.5.6 Externally Triggered Timeout Wakeup 398
21.5.7 16-bit PWM 398
21.6 Register Description 398
21.6.1 Configuration Register (LPTIM_CFG) 399
21.6.2 Counter Register (LPTIM_CNT) 401
21.6.3 Compare Value Register (LPTIM_CMP) 401
21.6.4 Target Value Register (LPTIM_TARGET) 402
21.6.5 Interrupt Enable Register (LPTIM_IE) 402
21.6.6 Interrupt Flag Register (LPTIM_IF) 402
21.6.7 Control Register (LPTIM_CTRL) 403
21.7 Operation Procedure 403
21.7.1 Enabling LPTIM Clock 403
21.7.2 LPTIM Counting 404
21.7.3 LPTIM PWM Output 404
21.7.4 LPTIM External Trigger Count Interrupt 405
21.7.5 LPTIM Timeout Mode 405

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.7.6 LPTIM External Clock Source Count Overflow Interrupt 406

22 Independent Watchdog Timer (IWDT) 407
22.1 Overview 407
22.2 Main Features 407
22.3 Register Description 407
22.3.1 Load Register (IWDT_LOAD) 408
22.3.2 Counter Register (IWDT_CNT) 408
22.3.3 Control Register (IWDT_CTRL) 408
22.3.4 Clear Register (IWDT_CLR) 410
22.3.5 Raw Interrupt Status Register (IWDT_INTRAW) 410
22.3.6 Interrupt Status Register (IWDT_MINTS) 410
22.3.7 Clock Division Register (IWDT_STALL) 411
22.3.8 Counter Lock Register (IWDT_LOCK) 411
22.4 Operation Procedure 412
22.4.1 IWDT Clock Enabling 412
22.4.2 IWDT Load Value Setting 412
22.4.3 IWDT Counter Overflow Interrupt 412
22.4.4 IWDT Counter Overflow Reset 413

23 System Window Watchdog (WWDT) 414
23.1 Overview 414
23.2 Main Features 414
23.3 Functional Description 415
23.4 Register Description 416
23.4.1 Control Register (WWDT_CTRL) 417
23.4.2 Configuration Register (WWDT_CFG) 417
23.4.3 Counter Register (WWDT_CNT) 418
23.4.4 Interrupt Enable Register (WWDT_IE) 418
23.4.5 Interrupt Flag Register (WWDT_IF) 419
23.4.6 PCLK Prescaler Counter Register (WWDT_DIV_CNT) 419
23.5 Operation Procedure 419
23.5.1 Interrupt Mode 419
23.5.2 WWDT Clear Counter 420

24 Real-time Clock (RTC) 421
24.1 Overview 421
24.2 Main Features 421
24.3 Block Diagram of Modules with System Connection 422
24.4 Module Description 423

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.4.1 RTC 423
24.4.2 Tamper Detector 431
24.4.3 Backup Register 432
24.5 Register Description 432
24.5.1 RTC Time Register (RTC_TIME) 433
24.5.2 RTC Date Register (RTC_DATE) 435
24.5.3 RTC Counter Read-write Control Register (RTC_ACCESS) 436
24.5.4 RTC Calibration Control Register (RTC_TRIM) 437
24.5.5 RTC Counter Increment Function Test Register (RTC_TEST) 438
24.5.6 Alarm 1 Time Setting Register (RTC_ALM1TIME) 440
24.5.7 Alarm 1 Date Setting Register (RTC_ALM1DATE) 440
24.5.8 Alarm 1 Setting Enable Register (RTC_ALM1EN) 441
24.5.9 Alarm 2 Setting Register (RTC_ALM2SETTING) 442
24.5.10 Tamper Detector Control and Configuration Register (RTC_TAMPCTRL) 443
24.5.11 Tamper Count Record Register (RTC_TAMPCNT) 443
24.5.12 Third New Tamper Event Timestamp Register (RTC_TAMP3TIME) 444
24.5.13 Third New Tamper Event Datestamp Register (RTC_TAMP3DATE) 444
24.5.14 Second New Tamper Event Timestamp Register (RTC_TAMP2TIME) 445
24.5.15 Second New Tamper Event Datestamp Register (RTC_TAMP2DATE) 445
24.5.16 Latest Tamper Event Timestamp Register (RTC_TAMP1TIME) 446
24.5.17 Latest Tamper Event Datestamp Register (RTC_TAMP1DATE) 447
24.5.18 RTC Interrupt Request Enable Register (RTC_INTEN) 447
24.5.19 RTC Raw Interrupt Status Register (RTC_INTRAW) 448
24.5.20 Interrupt Status Register after RTC Enable (RTC_INTSTA) 448
24.5.21 RTC Interrupt Status Clear Register (RTC_INTCLR) 449
24.5.22 Backup Register (RTC_BKREG_n) 449
24.6 Operation Procedure 449
24.6.1 RTC Clock Enabling 449
24.6.2 RTC Time Reading 450
24.6.3 RTC Alarm 1 Setting 450
24.6.4 RTC Alarm 2 Setting 451
24.6.5 RTC Backup Register 451
24.6.6 RTC Tamper Detection 452
24.6.7 RTC Tamper Counter Value Clearing 453
24.6.8 RTC Calibration 453
24.6.9 RTC Increment Test 453
25 CAN Bus Controller (CAN) 454

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.1 Overview 454
25.2 Main Features 454
25.3 Pin Description 454
25.4 Functional Description 455
25.4.1 Standard Frame Memory Buffer Layout 455
25.4.2 Extended Frame Memory Buffer Layout 456
25.5 List of Registers 456
25.5.1 Mode Register (CAN_MR (CONFIG0[7:0])) 457
25.5.2 Command Register (CAN_CMR (CONFIG0[15: 8])) 458
25.5.3 Status Register (CAN_SR (CONFIG0[23:16])) 459
25.5.4 Interrupt Status Register (CAN_ISR (CONFIG0[31:24])) 460
25.5.5 Interrupt Mask Register (CAN_IMR (CONFIG1[7:0])) 461
25.5.6 RX Data Bit Count Register (CAN_RMC (CONFIG1[15:8])) 462
25.5.7 Bus Timing Register (CAN_BTR0 (CONFIG1[23:16])) 463
25.5.8 Bus Timing Register (CAN_BTR1 (CONFIG1[31:24])) 464
25.5.9 Transmit Buffer Register (CAN_TXBUF) 465
25.5.10 Receive Buffer Register (CAN_RXBUF) 465
25.5.11 RX Filter Match Register (CAN_ACR) 466
25.5.12 RX Filter Mask Register (CAN_AMR) 467
25.5.13 Error Code Capture Register (CAN_ECC (ERRCR [7:0])) 469
25.5.14 Receive Error Count Register (CAN_RXERR (ERRCR [15:8])) 470
25.5.15 Transmit Error Count Register (CAN_TXERR (ERRCR [23:16])) 470
25.5.16 Arbitration Lost Capture Register (CAN_ALC (ERRCR [31:24])) 471
25.6 Operation Procedure 472
25.6.1 Transmitting CAN Data Frame 472
25.6.2 Receiving CAN Data Frame 473
26 Inter-integrated Circuit (I2C) Interface 474
26.1 Overview 474
26.2 Main Features 474
26.3 Pin Description 475
26.4 Functional Description 475
26.4.1 SMBus / PMBus 475
26.4.2 Bus Clearing 485
26.4.3 Configuring SCLHCNT and SCLLCNT for I2C 486
26.4.4 SDA Hold Time 487
26.4.5 Communication Using DMA 489
26.5 Register Description 489
26.5.1 I2C Control Register (I2C_CR) 491

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.2 I2C Slave Address Access Register (I2C_TAR) 495
26.5.3 I2C Slave Address Register (I2C_SAR) 496
26.5.4 I2C Data Command Register (I2C_DATACMD) 497
26.5.5 I2C Standard Mode SCL High-level Configuration Register (I2C_SSSCLHCNT) 498
26.5.6 I2C Standard Mode SCL Low-level Configuration Register (I2C_SSSCLLCNT) 499
26.5.7 I2C Fast Mode SCL High-level Configuration Register (I2C_FSSCLHCNT) 499
26.5.8 I2C Fast Mode SCL Low level Configuration Register (I2C_FSSCLLCNT) 499
26.5.9 I2C Interrupt Status Register (I2C_ISR) 500
26.5.10 I2C Interrupt Mask Register (I2C_INTMASK) 501
26.5.11 I2C Raw Interrupt Register (I2C_RAWISR) 503
26.5.12 I2C RX FIFO Threshold Register (I2C_RXTL) 506
26.5.13 I2C TX FIFO Threshold Register (I2C_TXTL) 506
26.5.14 I2C Combined and Independent Interrupt Clear Register (I2C_CLR) 507
26.5.15 I2C RX_UNDER Interrupt Clear Register (I2C_CLRRXUNDER) 507
26.5.16 I2C RX_OVER Interrupt Clear Register (I2C_CLRRXOVER) 507
26.5.17 I2C TX_OVER Interrupt Clear Register (I2C_CLRTXOVER) 508
26.5.18 I2C RD_REQ Interrupt Clear Register (I2C_CLRRDREQ) 508
26.5.19 I2C TX_ABRT Interrupt Clear Register (I2C_CLRTXABRT) 508
26.5.20 I2C RX_DONE Interrupt Clear Register (I2C_CLRRXDONE) 509
26.5.21 I2C ACTIVITY Interrupt Clear Register (I2C_CLRACTIVITY) 509
26.5.22 I2C STOP_DET Interrupt Clear Register (I2C_CLRSTOPDET) 509
26.5.23 I2C START_DET Interrupt Clear Register (I2C_CLRSTARTDET) 510
26.5.24 I2C GEN_CALL Interrupt Clear Register (I2C_CLRGENCALL) 510
26.5.25 I2C Enable Register (I2C_EN) 510
26.5.26 I2C Status Register (I2C_SR) 512
26.5.27 I2C Transmit Buffer Depth Register (I2C_TXFLR) 513
26.5.28 I2C Receive Buffer Depth Register (I2C_RXFLR) 513
26.5.29 I2C SDA Hold Time Register (I2C_SDAHOLD) 514
26.5.30 I2C Slave Data NACK Generation Register (I2C_SLVDATANACKONLY) . 514
26.5.31 I2C DMA Control Register (I2C_DMACR) 514
26.5.32 I2C DMA Transmit Data Level Register (I2C_DMATDLR) 515
26.5.33 I2C DMA Receive Data Level Register (I2C_DMARDLR) 515
26.5.34 I2C SDA Setup Time Register (I2C_SDASETUP) 515
26.5.35 I2C Broadcast Call ACK Register (I2C_ACKGENERALCALL) 516
26.5.36 I2C Enable Status Register (I2C_ENSR) 516

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.37 I2C Spike Suppression Limit Register (2C_FSSPKLEN) 517
26.5.38 I2C SMBUS Slave Clock Extension Timeout Register (I2C_SMBUSCLOCKLOWSEXT) 517
26.5.39 I2C SMBUS Master Clock Extension Timeout Register (I2C_SMBUSCLOCKLOWMEXT) 518
26.5.40 I2C SMBUS Maximum Bus Idle Count Register (I2C_SMBUSTHIGHMAXIDLECOUNT) 518
26.5.41 I2C SMBus Interrupt Status Register (I2C_SMBUSISR) 518
26.5.42 I2C SMBus Interrupt Mask Register (I2C_SMBUSINTMASK) 520
26.5.43 I2C SMBus RAW Interrupt Status Register (I2C_SMBUSRAWISR) 522
26.5.44 I2C SMBus Interrupt Status Clear Register (I2C_CLRSMBUSISR) 523
26.5.45 I2C SMBus ARP UDID LSB register (I2C_ SMBUSUDIDLSB) 524
26.6 Operation Procedure 524
26.6.1 Master and Slave Initialization Configuration 525
26.6.2 Master Transmitter 526
26.6.3 Master Receiver 527
26.6.4 Slave Transmitter 527
26.6.5 Slave Receiver 528
27 Inter-IC Sound (I2S) Interface 529
27.1 Overview 529
27.2 Pin Description 530
27.3 Functional Description 531
27.3.1 TX FIFO 531
27.3.2 RX FIFO 531
27.3.3 16-bit Channel Control 531
27.3.4 32-bit Channel Control 532
27.3.5 Port Synchronization 532
27.3.6 Interface Timing 533
27.4 Register Description 535
27.4.1 I2S Write Data Register (I2S_WR) 536
27.4.2 I2S Read Data Register (I2S_RD) 537
27.4.3 I2S Current Status Register (I2S_CSR) 538
27.4.4 I2S Global Control Register (I2S_GCR) 539
27.4.5 I2S Data Format Register (I2S_DFR) 541
27.4.6 I2S Interrupt Status Register (I2S_ISR) 543
27.4.7 I2S Interrupt Enable Register (I2S_IER) 545
27.4.8 I2S Interrupt Clear Register (I2S_ICR) 546
27.5 Operation Procedure 547

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28 Universal Asynchronous Receiver Transmitter (UART) 548
28.1 Overview 548
28.2 Main Features 548
28.3 Pin Description 549
28.4 Functional Description 550
28.4.1 Configurable Baud Rate 550
28.4.2 UART Transmission Mode 550
28.4.3 UART Reception Mode 550
28.4.4 IrDA Mode 551
28.5 Register Description 551
28.5.1 Receive Buffer Register (UART_RBR) 552
28.5.2 Transmit Buffer Register (UART_THR) 552
28.5.3 Baud Rate Parameter Low-order Register (UART_DLL) 553
28.5.4 Baud Rate Parameter High-order Register (UART_DLH) 553
28.5.5 Interrupt Enable Register (UART_IER) 553
28.5.6 Interrupt Status Register (UART_IIR) 554
28.5.7 FIFO Control Register (UART_FCR) 555
28.5.8 LINE Control Register (UART_LCR) 556
28.5.9 Flow Control Register (UART_MCR) 558
28.5.10 LINE Status Register (UART_LSR) 559
28.5.11 Flow Status Register (UART_MSR) 561
28.5.12 Low-power Baud Rate Division Low-order Register (UART_LPDLL) 561
28.5.13 Low-power Baud Rate Division High-order Register (UART_LPDLH) 562
28.5.14 Status Register (UART_USR) 562
28.5.15 TX FIFO Data Bit Count Register (UART_TFL) 563
28.5.16 RX FIFO Data Bit Count Register (UART_RFL) 563
28.5.17 Fractional Frequency Division Register (UART_DLF) 563
28.5.18 Receive Address Matching Register (UART_RAR) 564
28.5.19 Transmit Address Matching Register (UART_TAR) 564
28.5.20 LINE Control Extension Register (UART_LCRE) 564
28.6 Use Process 565
28.6.1 UART Initialization 565
28.6.2 UART Transmission Process 566
28.6.3 UART Reception Process 566

29 Low-power Universal Asynchronous Receiver Transmitter (LPUART) 567
29.1 Overview 567
29.2 Main Features 567
29.3 System Block Diagram 568

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

29.3.1 Block Diagram 568
29.3.2 Interface Timing 569
29.3.3 Reception Timing 569
29.3.4 Transmission Timing 569
29.4 Pin Description 570
29.5 Functional Description 570
29.5.1 Transmission Mode 570
29.5.2 Reception Mode 570
29.6 Register Description 571
29.6.1 Receive Data Register (LPUART_RXD) 571
29.6.2 Transmit Data Register (LPUART_TXD) 571
29.6.3 Status Register (LPUART_STA) 572
29.6.4 Control Register (LPUART_CON) 572
29.6.5 Interrupt Flag Register (LPUART_IF) 574
29.6.6 Baud Rate Register (LPUART_BAUD) 574
29.6.7 Transmit / Receive Enable Register (LPUART_EN) 575
29.6.8 Data Matching Register (LPUART_CMPARE) 575
29.6.9 Modulation Control Register (LPUART_MODU) 576
29.7 Operation Procedure 576
29.7.1 Initialization 576
29.7.2 Reception Process 576
29.7.3 Transmission Process 577
29.7.4 Suggested Configuration for Modulation Control Register 577

30 Universal Synchronous / Asynchronous Receiver Transmitter (USART) 578
30.1 Overview 578
30.2 Main Features 578
30.3 Pin Description 580
30.4 Functional Description 580
30.4.1 Baud Rate Generator 580
30.4.2 Receiver and Transmitter Control 582
30.4.3 Synchronous and Asynchronous Modes 583
30.4.4 IrDA Mode 604
30.4.5 SPI Mode 607
30.4.6 LIN Mode 613
30.4.7 LIN Error 625
30.4.8 LIN Frame Processing 627
30.4.9 Using DMA/PDC to Process LIN Frames 631
30.4.10 Test Mode 635

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5 Register Description 637
30.5.1 USART Control Register (USART_CR) 637
30.5.2 USART Control Register (USART_CR, SPI Mode) 640
30.5.3 USART Mode Register (USART_MR) 641
30.5.4 USART Mode Register (USART_MR, SPI Mode) 645
30.5.5 USART Interrupt Enable Register (UART_IER) 646
30.5.6 USART Interrupt Enable Register (USART_IER, SPI Mode) 647
30.5.7 USART Interrupt Enable Register (USART_IER, LIN Mode) 648
30.5.8 USART Interrupt Disable Register (USART_IDR) 649
30.5.9 USART Interrupt Disable Register (USART_IDR, SPI Mode) 649
30.5.10 USART Interrupt Disable Register (USART_IDR, LIN Mode) 650
30.5.11 USART Interrupt Mask Register (USART_IMR) 651
30.5.12 USART Interrupt Mask Register (USART_IMR, SPI Mode) 652
30.5.13 USART Interrupt Mask Register (USART_IMR, LIN Mode) 652
30.5.14 USART Channel Status Register (USART_CSR) 653
30.5.15 USART Channel Status Register (USART_CSR, SPI Mode) 656
30.5.16 USART Channel Status Register (USART_CSR, LIN Mode) 657
30.5.17 USART Receive Holding Register (USART_RHR) 661
30.5.18 USART Transmit Holding Register (USART_THR) 661
30.5.19 USART Baud Rate Generator Register (USART_BRGR) 662
30.5.20 USART Receiver Timeout Register (USART_RTOR) 662
30.5.21 USART Transmit Time Protection Register (USART_TTGR) 663
30.5.22 USART FI/DI Ratio Register (USART_FIDI) 663
30.5.23 USART IrDA Filter Register (USART_IF) 663
30.5.24 USART Manchester Configuration Register (USART_MAN) 664
30.5.25 USART LIN Mode Register (USART_LINMR) 665
30.5.26 USART LIN Identifier Register (USART_LINIR) 667
30.5.27 USART LIN Baud Rate Register (USART_LINBRR) 667
30.5.28 USART Write Protection Mode Register (USART_WPMR) 668
30.5.29 USART Write Protection Status Register (USART_WPSR) 668
30.6 Operation Procedure 669
30.6.1 UART Mode 669
30.6.2 SPI Mode 670
30.6.3 LIN Mode 672
31 Serial Peripheral Interface (SPI) 676
31.1 Overview 676
31.2 Main Features 676
31.3 System Block Diagram 677

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.4 Pin Description 677
31.5 Functional Description 678
31.5.1 SPI clock frequency control register 678
31.5.2 TX FIFO 678
31.5.3 RX FIFO 679
31.5.4 SPI Control Logic 680
31.5.5 Data Concatenation 682
31.5.6 Interface Timing 683
31.6 Register Description 685
31.6.1 Transmit Data Register (SPI_TXREG) 686
31.6.2 Receive Data Register (SPI_RXREG) 686
31.6.3 Current Status Register (SPI_CSTAT) 686
31.6.4 Interrupt Status Register (SPI_INTSTAT) 687
31.6.5 Masked Interrupt Status Register (SPI_MINTSTAT) 690
31.6.6 Interrupt Enable Register (SPI_INTEN) 692
31.6.7 Interrupt Clear Register (SPI_INTCLR) 693
31.6.8 Global Control Register (SPI_GCTL) 695
31.6.9 General Control Register (SPI_CCTL) 697
31.6.10 SPI Clock Frequency Control Register (SPI_SPBRG) 699
31.6.11 Receive Data Number Register (SPI_RXDNR) 699
31.6.12 Transmit Data Number Register (SPI_TXDNR) 700
31.6.13 Slave Chip Select Register (SPI_SCSR) 701
31.7 Operation Procedure 701
31.7.1 Master Mode 701
31.7.2 Slave Mode 707
32 Quad-SPI Controller (QSPI) 713
32.1 Overview 713
32.2 Main Features 713
32.3 System Block Diagram 714
32.4 Pin Description 714
32.5 Functional Description 715
32.5.1 AHB Control Interface 715
32.5.2 Direct Access Controller (DAC) 717
32.5.3 Indirect Access Controller (INDAC) 719
32.5.4 DMA Peripheral Controller 732
32.5.5 Software Triggered Instruction Generator (STIG) 737
32.5.6 Arbitration Between Direct/Indirect Access Controllers and STIG 738
32.5.7 SPI Command Conversion 739

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.5.8 Flash Instruction Type Selection 740
32.5.9 APB Interface and Register Module 743
32.5.10 SRAM Module 743
32.6 Register Description 743
32.6.1 QSPI Configuration Register (QSPI_CR) 745
32.6.2 QSPI Device Read Instruction Register (QSPI_DRIR) 749
32.6.3 QSPI Device Write Instruction Register (QSPI_DWIR) 751
32.6.4 QSPI Device Delay Register (QSPI_DDLR) 752
32.6.5 QSPI Read Data Capture Register (QSPI_RDCR) 753
32.6.6 QSPI Device Size Configuration Register (QSPI_DSCR) 754
32.6.7 QSPI SRAM Partitioning Configuration Register (QSPI_SPR) 755
32.6.8 QSPI Indirect AHB Address Trigger Register (QSPI_IATR) 755
32.6.9 QSPI DMA Peripheral Configuration Register (QSPI_DMACR) 756
32.6.10 QSPI Address Remapping Register (QSPI_RAR) 757
32.6.11 QSPI Mode Bit Register (QSPI_MBR) 757
32.6.12 QSPI SRAM Fill Level Register (QSPI_SFLR) 758
32.6.13 QSPI Transmit Threshold Register (QSPI_TXHR) 758
32.6.14 QSPI Receive Threshold Register (QSPI_RXHR) 758
32.6.15 QSPI Write Completion Control Register (QSPI_WCR) 759
32.6.16 QSPI Polling End Register (QSPI_PER) 760
32.6.17 QSPI Interrupt Flag Register (QSPI_IFR) 760
32.6.18 QSPI Interrupt Mask Register (QSPI_IMR) 762
32.6.19 QSPI Write Protection Low Register (QSPI_WPLR) 763
32.6.20 QSPI Write Protection High Register (QSPI_WPHR) 763
32.6.21 QSPI Write Protection Configuration Register (QSPI_WPCR) 763
32.6.22 QSPI Indirect Read Transfer Control Register (QSPI_IRTR) 764
32.6.23 QSPI Indirect Read Transfer Fill Level Watermark Register (QSPI_IRTWR) 766
32.6.24 QSPI Indirect Read Transfer Start Address Register (QSPI_IRTSAR) 766
32.6.25 QSPI Indirect Read Transfer Byte Number Register (QSPI_IRTNR) 766
32.6.26 QSPI Indirect Write Transfer Control Register (QSPI_IWTR) 767
32.6.27 QSPI Indirect Write Transfer Fill Level Watermark Register (QSPI_IWTWR) 768
32.6.28 QSPI Indirect Write Transfer Start Address Register (QSPI_IWTSAR) 768
32.6.29 QSPI Indirect Write Transfer Byte Number Register (QSPI_IWTNR) 769
32.6.30 QSPI Indirect Trigger Address Range Register (QSPI_ITARR) 769
32.6.31 QSPI Flash Command Control Register (QSPI_FCR) 770
32.6.32 QSPI Flash Command Address Register (QSPI_FCAR) 772

Unicmicro

32.6.33 QSPI Flash Command Read Data Low Register (QSPI_FCRLR) 772
32.6.34 QSPI Flash Command Read Data High Register (QSPI_FCRHR) 772
32.6.35 QSPI Flash Command Write Data Low Register (QSPI_FCWLR) 773
32.6.36 QSPI Flash Command Write Data High Register (QSPI_FCWHR) 773
32.6.37 QSPI Polling Flash Status Register (QSPI_PFSR) 774
32.7 Operation Procedure 774
32.7.1 Configuring QSPI Controller after Reset 775
32.7.2 QSPI Controller Configuration Optimization 775
32.7.3 Usage of Flash Command Control Register (STIG Operation) 777
32.7.4 Traditional SPI Mode 777
32.7.5 Entering and Exiting XIP Mode 778
32.7.6 Indirect Data Transfer Mode 781
32.7.7 AHB Address Remapping 782
32.7.8 Interrupt Response 782
32.7.9 AHB Protection Register 783
32.7.10 Example of DAC Mode Configuration Process 783
33 Secure Digital Input and Output (SDIO) 784
33.1 Overview 784
33.2 Main Features 784
33.3 Pin Description 785
33.4 Functional Description 785
33.4.1 Internal DMA (IDMA) 785
33.4.2 Interrupt 790
33.4.3 Data Transfer 791
33.5 Register Description 794
33.5.1 Control Register (SDIO_CTRL) 795
33.5.2 Power Enable Register (SDIO_POWEN) 797
33.5.3 Clock Division Register (SDIO_CLKDIV) 797
33.5.4 Clock Enable Register (SDIO_CLKENA) 798
33.5.5 Timeout Register (SDIO_TIMEOUT) 798
33.5.6 Bit Width Register (SDIO_WIDTH) 799
33.5.7 Block Size Register (SDIO_BLKSIZ) 799
33.5.8 Byte Count Register (SDIO_BYTCNT) 800
33.5.9 Interrupt Mask Register (SDIO_INTMASK) 800
33.5.10 Command Parameter Register (SDIO_CMDARG) 801
33.5.11 Command Register (SDIO_CMD) 801
33.5.12 Response 0 Register (SDIO_RESP0) 805
33.5.13 Response 1 Register (SDIO_RESP1) 805

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.14 Response 2 Register (SDIO_RESP2) 805
33.5.15 Response 3 Register (SDIO_RESP3) 805
33.5.16 Maskable Interrupt Status Register (SDIO_MINTSTS) 806
33.5.17 Raw Interrupt Status Register (SDIO_RINTSTS) 807
33.5.18 Status Register (SDIO_STATUS) 808
33.5.19 FIFO Compare Register (SDIO_FIFOTH) 809
33.5.20 Card Detection Register (SDIO_CDETECT) 811
33.5.21 Write Protection Register (SDIO_WRTPRT) 811
33.5.22 TCBCNT Register (SDIO_TCBCNT) 811
33.5.23 TBBCNT Register (SDIO_TBBCNT) 811
33.5.24 DEBOUNCE Register (SDIO_DEBOUNCE) 812
33.5.25 Hardware Reset Register (SDIO_RST) 812
33.5.26 Bus Mode Register (SDIO_BMOD) 812
33.5.27 Poll Demand Register (SDIO_PLDMND) 813
33.5.28 Descriptor List Base Address Register (SDIO_DBADDR) 814
33.5.29 Internal DMA Status Register (SDIO_IDSTS) 814
33.5.30 Internal DMA Interrupt Enable Register (SDIO_IDINTEN) 816
33.5.31 Current Host Descriptor Address Register (SDIO_DSCADDR) 816
33.5.32 Current Buffer Descriptor Address Register (SDIO_BUFADDR) 817
33.5.33 Phase Shift Enable Register (SDIO_ENABLESHIFT) 817
33.5.34 Data Register (SDIO_DATA) 818
33.6 Operation Procedure 818
33.6.1 Initialization 818
33.6.2 Card Enumeration 819
33.6.3 IDMA 821
33.6.4 SDIO Application Notes 824
34 Ethernet Media Access Controller (EMAC) 827
34.1 Overview 827
34.2 Main Features 827
34.3 Pin Description 828
34.4 List of EMAC Registers 829
34.4.1 EMAC Configuration Register (EMAC_CONFIG) 830
34.4.2 Frame Filter Register (EMAC_FRAMEFILTER) 833
34.4.3 Hash Table High Register (EMAC_HASHTABLEHIGH) 835
34.4.4 Hash Table Low Register (EMAC_HASHTABLELOW) 835
34.4.5 GMII Address Register (EMAC_GMIIADDRESS) 836
34.4.6 GMII Data Register (EMAC_GMIIDATA) 837
34.4.7 Flow Control Register (EMAC_FLOWCONTROL) 837

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.4.8 VLAN Tag Register (EMAC_VLANTAG) 838
34.4.9 Debug Register (EMAC_DEBUGER) 839
34.4.10 LPI Control Status Register (EMAC_LPICS) 841
34.4.11 LPI Timer Register (EMAC_LPIT) 842
34.4.12 Interrupt Status Register (EMAC_INTSTATUS) 842
34.4.13 Interrupt Mask Register (EMAC_INTMASK) 843
34.4.14 EMAC Address Register High 0−15 (EMAC_ADDR0−15H) 844
34.4.15 EMAC Address Register Low 0−15 (EMAC_ADDR0−15L) 844
34.4.16 RGMII Control Status Register (EMAC_RGMIICS) 844
34.4.17 Timeout Watchdog Register (EMAC_WDG) 845
34.5 List of PTP Registers 845
34.5.1 Timestamp Control Register (PTP_TSCTL) 845
34.5.2 Timestamp Sub-second Increment Register (PTP_SUBSECINC) 848
34.5.3 Timestamp Seconds Register (PTP_SEC) 849
34.5.4 Timestamp Nanoseconds Register (PTP_NANOSEC) 849
34.5.5 Timestamp Seconds Update Register (PTP_SECUD) 849
34.5.6 Timestamp Nanoseconds Update Register (PTP_NANOSECUD) 850
34.5.7 Timestamp Addend Register (PTP_TSADDEND) 850
34.5.8 Timestamp Target Seconds Register (PTP_TGTSEC) 850
34.5.9 Timestamp Target Nanoseconds Register (PTP_TGTNANOSEC) 851
34.6 List of DMA Registers 851
34.6.1 Bus Mode Register (DMA_BUSMODE) 852
34.6.2 Transmit Register (DMA_TPD) 853
34.6.3 Receive Register (DMA_RPD) 854
34.6.4 Receive Descriptor Address Register (DMA_RLA) 854
34.6.5 Transmit Descriptor Address Register (DMA_TLA) 854
34.6.6 DMA Status Register (DMA_STATUS) 855
34.6.7 DMA Operation Mode Register (DMA_OPMODE) 857
34.6.8 DMA Interrupt Enable Register (DMA_INTEN) 860
34.6.9 DMA Lost Frame and Buffer Overflow Counter Register (DMA_FMBOCNT) 861
34.6.10 Receive Interrupt Watchdog Timer Register (DMA_RIWDT) 862
34.6.11 DMA Current Transmit Descriptor Register (DMA_CTD) 862
34.6.12 DMA Current Receive Descriptor Register (DMA_CRD) 862
34.6.13 DMA Current Transmit Buffer Address Register (DMA_CTB) 863
34.6.14 DMA Current Receive Buffer Address Register (DMA_CRB) 863
34.7 List of MMC Registers 863
34.7.1 MMC Control Register (MMC_CONTROL) 868

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.7.2 MMC Receive Interrupt Register (MMC_RI) 868
34.7.3 MMC Transmit Interrupt Register (MMC_TI) 872
34.7.4 MMC Receive Interrupt Mask Register (MMC_RIM) 876
34.7.5 MMC Transmit Interrupt Mask Register (MMC_TIM) 880
34.7.6 MMC Receive Checksum Interrupt Mask Register (MMC_IPCINTRMASK) 884
34.7.7 MMC Receive Checksum Interrupt Register (MMC_IPCINTR) 888
34.8 DMA Descriptors 893
34.8.1 Transmit Descriptor 0 (TDES0) 893
34.8.2 Transmit Descriptor 1 (TDES1) 894
34.8.3 Transmit Descriptor 2 (TDES2) 895
34.8.4 Transmit Descriptor 3 (TDES3) 895
34.8.5 Receive Descriptor 0 (RDES0) 896
34.8.6 Receive Descriptor 1 (RDES1) 897
34.8.7 Receive Descriptor 2 (RDES2) 898
34.8.8 Receive Descriptor 3 (RDES3) 898
34.9 Operation Procedure 898
34.9.1 DMA Initialization 898
34.9.2 EMAC Initialization 899
34.9.3 Performing Transmission 899
34.9.4 Stopping and Restarting Transmission 899
34.9.5 Guidelines for Using EEE Features 900
35 Universal Serial Bus Full-speed Device Interface (USB) 901
35.1 Overview 901
35.2 Main Features 901
35.3 Functional Description 902
35.3.1 Interrupt Status and Control Registers 902
35.3.2 Address Setting 902
35.3.3 Remote Wakeup 902
35.3.4 Selective NAK Response for CRC Errors in Token and Data Packets 903
35.3.5 Packet Length Exceeding 64 Bytes 903
35.3.6 Packet Loss EOP 903
35.3.7 IN Operation ACK Timeout, Responding NAK for Next IN Operation 904
35.3.8 FIFO Access and Memory Access 904
35.4 Register Description 905
35.4.1 Working Mode Register (USB_WORKINGMODE) 906
35.4.2 EP0 Transfer Control Register (USB_EP0CSR) 909
35.4.3 EP1 Transfer Control Register (USB_EP1CSR) 911

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.4.4 EP2 Transfer Control Register (USB_EP2CSR) 913
35.4.5 EP3 Transfer Control Register (USB_EP3CSR) 915
35.4.6 EP4 Transfer Control Register (USB_EP4CSR) 917
35.4.7 USB Address Register (USB_ADDR) 919
35.4.8 SETUP Data Packet Register (USB_SETUP03DATA) 920
35.4.9 SETUP Data Packet Register (USB_SETUP47DATA) 920
35.4.10 Endpoint Address Configuration Register (USB_EPADDR) 920
35.4.11 Bus Packet PID Register (USB_CURRENTPID) 920
35.4.12 Frame Number Register (USB_CURRENTFRAMENUMBER) 921
35.4.13 CRC Error Counter Register (USB_CRCERRORCNT) 921
35.4.14 Detection Time Register (USB_STATUSDETECTCNT) 921
35.4.15 EP0 Transmit Data Byte Number Register (USB_EP0SENDBN) 922
35.4.16 EP1 Transmit Data Byte Number Register (USB_EP1SENDBN) 922
35.4.17 EP2 Transmit Data Byte Number Register (USB_EP2SENDBN) 923
35.4.18 EP3 Transmit Data Byte Number Register (USB_EP3SENDBN) 923
35.4.19 EP4 Transmit Data Byte Number Register (USB_EP4SENDBN) 923
35.4.20 EP0 FIFO Access Entry (USB_EP0FIFO) 924
35.4.21 EP1 FIFO Access Entry (USB_EP1FIFO) 924
35.4.22 EP2 FIFO Access Entry (USB_EP2FIFO) 924
35.4.23 EP3 FIFO Access Entry (USB_EP3FIFO) 924
35.4.24 EP4 FIFO Access Entry (USB_EP4FIFO) 925
35.4.25 Status Register (USB_INTSTATRAW) 925
35.4.26 Interrupt Enable Register (USB_INTEN) 927
35.4.27 Interrupt Clear Register (USB_INTCLR) 930
35.5 Operation Procedure 934
35.5.1 USB Connection 934
35.5.2 SETUP Data and EP0 Control Transfer Data 934
35.5.3 Endpoint In Transfer 936
35.5.4 Endpoint Out Transfer 937
36 Analog-to-digital Converter (ADC) 938
36.1 Overview 938
36.2 Main Features 938
36.3 System Block Diagram 940
36.4 Pin Description 940
36.5 Functional Description 941
36.5.1 Analog Watchdog 941
36.5.2 ADC Clock Generator 942
36.5.3 Data Receiver 943

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.5.4 RX FIFO 944
36.6 Conversion Mode Description 944
36.6.1 Overview of Conversion Sequence 944
36.6.2 Basic Conversion Mode of Regular Sequence 947
36.6.3 Basic Conversion Mode of Injection Sequence 948
36.6.4 Dual-ADC Cooperative Mode for Regular Sequence 948
36.6.5 Dual-ADC Cooperative Mode for Injection Sequence 950
36.7 Register Description 951
36.7.1 Analog Circuit Control Register (ADC_ANCTRL) 953
36.7.2 Digital Circuit Control Register (ADC_DGCTRL) 956
36.7.3 ADCx Clock Control Register (ADC_CLKCTRL) 960
36.7.4 Multiple Average Setting Register 0 (ADC_CHAVGCFG0) 962
36.7.5 Multiple Average Setting Register 1 (ADC_CHAVGCFG1) 964
36.7.6 Regular Sequence Channel Setting Register 0 (ADC_RGLCHCFG0) 965
36.7.7 Regular Sequence Channel Setting Register 1 (ADC_RGLCHCFG1) 966
36.7.8 Regular Sequence Channel Setting Register 2 (ADC_RGLCHCFG2) 967
36.7.9 Regular Sequence Channel Setting Register 3 (ADC_RGLCHCFG3) 968
36.7.10 Injection Sequence Channel Setting Register (ADC_INJCHCFG) 969
36.7.11 Channel Data Registers 0−16 (ADC_CHDAT0−19) 971
36.7.12 Dual-data Shadow Register (ADC_DUALDAT) 971
36.7.13 RX FIFO Data Register (ADC_FIFO_OUT) 972
36.7.14 Watchdog Enable Register (ADC_WDGEN) 972
36.7.15 Watchdog Comparison Condition Register (ADC_WDGCOND) 974
36.7.16 Current Status Register (ADC_STAT) 975
36.7.17 Interrupt Status / Clear Register (ADC_INTSTAT) 977
36.7.18 Interrupt Enable Register (ADC_INTEN) 979
36.7.19 Enabled Interrupt Status Shadow Register (ADC_MINTSTAT) 979
36.8 Operation Procedure 980
36.8.1 Multi-channel A/D Conversion in Single Scan Mode of Regular Sequence 980
36.8.2 Multi-channel A/D Conversion in Continuous Scan Mode of Regular Sequence 981
36.8.3 Multi-channel A/D Conversion in Intermittent Scan Mode of Regular Short Sequence 982
36.8.4 Multi-channel A/D Conversion in Single Scan Mode of Injection Sequence 983
36.8.5 Analog Watchdog 984
36.8.6 Differential Channel Input 985

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.8.7 Dual-ADC Cooperative Mode for Regular Sequence 986
36.8.8 Dual-ADC Cooperative Mode for Injection Sequence 987
36.8.9 ADC Sampling via OPA Buffering 988
37 Digital-to-analog Converter (DAC) 990
37.1 Overview 990
37.2 Main Features 990
37.3 System Block Diagram 991
37.4 Functional Description 991
37.4.1 DAC Conversion Control 991
37.4.2 Trigger Mode 992
37.4.3 Noise Wave 992
37.4.4 Triangle Wave 992
37.4.5 DMA 992
37.4.6 Dual DAC Coordination 993
37.5 Register Description 993
37.5.1 Control Register (DAC_CTRL) 994
37.5.2 Software Trigger Register (DAC_SWTRG) 998
37.5.3 DAC1 12-bit Right-aligned Data Register (DAC_DHR12R0) 998
37.5.4 DAC1 12-bit Left-aligned Data Register (DAC_DHR12L0) 998
37.5.5 DAC1 8-bit Right-aligned Data Register (DAC_DHR8R0) 998
37.5.6 DAC2 12-bit Right-aligned Data Register (DAC_DHR12R1) 999
37.5.7 DAC2 12-bit Left-aligned Data Register (DAC_DHR12L1) 999
37.5.8 DAC2 8-bit Right-aligned Data Register (DAC_DHR8R1) 999
37.5.9 Dual-DAC 12-bit Right-aligned Data Register (DAC_DHR12RD) 1000
37.5.10 Dual-DAC 12-bit Left-aligned Data Register (DAC_DHR12LD) 1000
37.5.11 Dual-DAC 8-bit Right-aligned Data Register (DAC_DHR8RD) 1000
37.5.12 DAC1 Output Data Register (DAC_DOR0) 1001
37.5.13 DAC2 Output Data Register (DAC_DOR1) 1001
37.5.14 Interrupt Status Register (DAC_IS) 1001
37.5.15 DAC Clock Setting Register (DAC_CLK) 1002
37.6 Operation Procedure 1002
37.6.1 Direct Output of New Values 1002
37.6.2 Output Value Update Using Trigger Control 1003
37.6.3 Output Value Update Using DMA Control 1003
37.6.4 Noise Wave Generation 1004
37.6.5 Triangle Wave Generation 1005
38 Analog Comparator (ACMP) 1006

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

38.1 Overview 1006
38.2 Main Features 1006
38.3 System Block Diagram 1007
38.4 Functional Description 1007
38.4.1 Voltage Comparator 1007
38.4.2 Comparator Hysteresis Function 1008
38.5 Register Description 1008
38.5.1 Write Unlock Register (ACMP_UNLOCK) 1009
38.5.2 Analog Comparator 1 Register (ACMP_CFG0) 1009
38.5.3 Analog Comparator 2 Register (ACMP_CFG1) 1010
38.5.4 Analog Comparator 3 Register (ACMP_CFG2) 1012
38.6 Operation Procedure 1013
39 Operational Amplifier (OPA) 1014
39.1 Overview 1014
39.2 Main Features 1014
39.3 System Block Diagram 1015
39.4 Register Description 1016
39.4.1 Write Unlock Register (OPA_UNLOCK) 1016
39.4.2 OPA1 Setting Register (OPA_CFG0) 1016
39.4.3 OPA2 Setting Register (OPA_CFG1) 1018
39.5 Operation Procedure 1020
39.5.1 UNITBUFF Mode 1020
39.5.2 External Feedback Circuit Setup in OPA Mode 1020
39.5.3 PGA Mode 1021
39.5.4 CMP Mode 1021
40 Temperature Sensor (TS) 1023
40.1 Overview 1023
40.2 Main Features 1023
40.3 Functional Description 1024
40.3.1 Temperature Measurement 1024
40.4 Register Description 1025
40.4.1 Write Unlock Register (TS_UNLOCK) 1025
40.4.2 TSVREF Setting Register (TS_VREFCFG) 1025
40.4.3 Temperature Sensor Setting Register (TS_CFG) 1026
40.4.4 Temperature Sensor Data Register (TS_DATA) 1027
40.5 Operation Procedure 1027
41 Internal Reference Voltage Source (VREF) 1028

Unicmicro

41.1 Overview 1028
41.2 Register Description 1028
41.2.1 Write Unlock Register (VREF_UNLOCK) 1028
41.2.2 TSVREF Setting Register (TS_VREFCFG) 1029
41.3 Operation Procedure 1029
42 SysTick Timer 1030
42.1 Overview 1030
42.2 Register Description 1030
42.2.1 SysTick Control and Status Register (SYSTICK_CTRL) 1031
42.2.2 SysTick Reload Value Register (SYSTICK_LOAD) 1031
42.2.3 SysTick Current Value Register (SYSTICK_VAL) 1032
42.3 Operation Procedure 1032
43 Debug Support (DBG) 1033
44 Revision History 1034

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

List of Tables

Table 3-1: Memory Remapping Address....12
Table 3-2: Memory Mapping Address....12
Table 4-1: EFC Special Feature Address Description....17
Table 4-2: OTP Area of EFC....17
Table 4-3: List of EFC Registers....19
Table 4-4: Relationship between Read Wait Cycle and Frequency at V_{DD} = 1.1 V....21
Table 4-5: List of Cache Registers....28
Table 5-1: Low-power Modes Summary....32
Table 5-2: List of PMU Registers....36
Table 5-3: LVD (VDT) Settings....42
Table 5-4: BOR Settings....43
Table 5-5: PDR Settings....44
Table 6-1: System Reset Source....53
Table 6-2: Reset Mode....53
Table 6-3: List of RCM Registers....54
Table 7-1: List of GPIO Registers....89
Table 8-1 : Cortex-M4 Interrupt Sources....100
Table 9-1: List of SYSCFG Registers....104
Table 10-1: Common CRC Formats....124
Table 10-2: List of CRC Registers....124
Table 11-1: DMA0 Handshake Signal....128
Table 11-2: DMA1 Handshake Signal....129
Table 11-3: List of DMA Registers....129
Table 12-1: DCMI Pin Description....153
Table 12-2: 8-bit Data Storage Format....154
Table 12-3: 10-bit Data Storage Format....155
Table 12-4: 12-bit Data Storage Format....155
Table 12-5: 14-bit Data Storage Format....155
Table 12-6: List of DCMI Registers....157
Table 13-1: EMC Pin Description....171
Table 13-2: Corresponding Time of Read/Write Timing of Off-chip Memory....174
Table 13-3: Correspondence between EBn and Peripheral Data Line....175
Table 13-4: List of EMC Registers....176
Table 13-5: Chip Select Segment Start Address....176
Table 14-1: Input and Output Table of Cordic Controller Function Mode 0....184
Table 14-2: Input and Output Table of Cordic Controller Function Mode 1....184
Table 14-3: Input and Output Table of Cordic Controller Function Mode 2....185

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 14-4: Input and Output Table of Cordic Controller Function Mode 3 185
Table 14-5: Input and Output Table of Cordic Controller Function Mode 4 185
Table 14-6: Input and Output Table of Cordic Controller Function Mode 5 186
Table 14-7: Input and Output Table 1 of Cordic Controller Function Mode 6 186
Table 14-8: Input and Output Table 2 of Cordic Controller Function Mode 6 186
Table 14-9: Input and Output Table 3 of Cordic Controller Function Mode 6 187
Table 14-10: Input and Output Table 4 of Cordic Controller Function Mode 6 187
Table 14-11: Input and Output Table 1 of Cordic Controller Function Mode 7 187
Table 14-12: Input and Output Table 2 of Cordic Controller Function Mode 7 187
Table 14-13: Input and Output Table 3 of Cordic Controller Function Mode 7 188
Table 14-14: Input and Output Table 4 of Cordic Controller Function Mode 7 188
Table 14-15: List of Cordic Controller Registers 188
Table 15-1: List of AES Registers 195
Table 16-1: List of SHA Registers 213
Table 17-1: List of RNG Registers 221
Table 18-1: TIM0& TIM7 Pin Description 225
Table 18-2: Timer Interconnection 226
Table 18-3: Counting Direction versus Encoder Signals 258
Table 18-4: Seven DMA Requests Supported by TIM 264
Table 18-5: List of Registers Regarding Advanced-control Timers TIM0 & TIM7 266
Table 18-6: State Correspondence between Control Register and Complementary Output Channel 295
Table 19-1: Pin Descriptions on TIM1–TIM4 & TIM8–TIM13 311
Table 19-2: Counting Direction versus Encoder Signals 339
Table 19-3: Seven DMA Requests Supported by TIM 345
Table 19-4: General-purpose Timers (TIM1–TIM4 & TIM8–TIM13) 348
Table 20-1: List of TIM5 & TIM6 Registers 388
Table 21-1: LPTIM Pin Description 396
Table 21-2: List of LPTIM Registers 398
Table 22-1: List of IWDT Registers 407
Table 23-1: WWDT Counter Overflow Time Calculation 415
Table 23-2: List of WWDT Registers 417
Table 24-1: HOUR Register Coding 425
Table 24-2: Example of Alarm 1 Setting 428
Table 24-3 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x0 430
Table 24-4 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x1 430
Table 24-5 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x2 430
Table 24-6 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x3 431

Unicmicro

Table 24-7: List of RTC Registers 432
Table 25-1: CAN Pin Description 454
Table 25-2: Standard Frame Transmit/Receive Buffer Layout Diagram 455
Table 25-3: Extended Frame Memory Buffer Layout Diagram 456
Table 25-4: List of CAN Registers 456
Table 25-5: ALC Register Description 472
Table 26-1: I2C Pin Description 475
Table 26-2: SMBus Protocol Usage Table 476
Table 26-3: Table of SMBus ARP Commands Derived via TxFIFO Commands in I2C 478
Table 26-4: List of I2C Registers 490
Table 26-5: I2C Slave and Master Configuration Table 495
Table 27-1: I2S Pin Description 530
Table 27-2: PCLK Frequency Requirements for I2S in Different Modes 532
Table 27-3: List of I2S Registers 535
Table 28-1: UART Pin Description 549
Table 28-2: List of UART Registers 551
Table 29-1: LPUART Pin Description 570
Table 29-2: List of LPUART Registers 571
Table 29-3: MCTL Configuration Parameter Table for LPUART_MODU 577
Table 30-1: Pin Description 580
Table 30-2: Example of USART Parity Bits 596
Table 30-3: Maximum Time Protection Period 598
Table 30-4: Maximum Timeout Periods at Certain Standard Baud Rates 599
Table 30-5: IrDA Pulse Duration 605
Table 30-6: IrDA Baud Rate Error 606
Table 30-7: SPI Bus Protocol Modes 610
Table 30-8: Response Data Length with DLM = 1 622
Table 30-9: List of USART Registers 637
Table 30-10 : Reference Table of USART Clock Frequency Division 662
Table 31-1: SPI Pin Description 677
Table 31-2: Motorola Timing Mode 683
Table 31-3: TI Timing Mode 684
Table 31-4: List of SPI Registers 685
Table 32-1: QSPI Pin Description 714
Table 32-2: SRAM Access Priority 732
Table 32-3: List of QSPI Flash Controller Registers 743
Table 33-1: SDIO Pin Description 785
Table 33-2: DES0 Definition 788

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 33-3: DES1 Definition..... 789
Table 33-4: DES2 Definition..... 789
Table 33-5: DES3 Definition..... 789
Table 33-6: SDIO Interrupt Source Descriptions..... 790
Table 33-7: List of SDIO Registers..... 794
Table 34-1: EMAC Pin Description..... 828
Table 34-2: List of EMAC Registers..... 829
Table 34-3 : Timestamp Snapshot Relationship Table..... 848
Table 34-4: List of DMA Registers..... 851
Table 34-5: List of MMC Registers..... 863
Table 35-1: List of USB Registers..... 905
Table 36-1: ADC Pin Description..... 940
Table 36-2: Data Format in RX FIFO..... 944
Table 36-3: List of ADC Memory-mapped Registers..... 951
Table 37-1: List of DAC Registers..... 993
Table 38-1: List of ACMP Registers..... 1008
Table 39-1: List of OPA Registers..... 1016
Table 40-1: List of TS Registers..... 1025
Table 41-1: List of VREF Registers..... 1028
Table 42-1: List of SysTick Registers..... 1030

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

List of Figures

Figure 3-1: Bus Architecture Diagram 10
Figure 3-2: Memory Address Mapping 11
Figure 4-1: EFC Memory Address Mapping 16
Figure 6-1: Clock Logic 50
Figure 13-1: Write Operation Timing Diagram for Off-chip Memory Controller 173
Figure 13-2: Read Operation Timing Diagram for Off-chip Memory Controller 174
Figure 13-3: I8080 TFT_LCD Timing Interface 176
Figure 18-1: TIM0 & TIM7 System Block Diagram 224
Figure 18-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 2 228
Figure 18-3: Counter Timing Diagram with Prescaler Division Changing from 1 to 4 228
Figure 18-4: Up-counting Waveform Diagram, Internal Clock not Divided 230
Figure 18-5: Up-counting Waveform Diagram, Internal Clock Divided by 2 230
Figure 18-6: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded) 231
Figure 18-7: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded) 231
Figure 18-8: Down-counting Waveform Diagram, Internal Clock not Divided 232
Figure 18-9: Down-counting Waveform Diagram, Internal Clock Divided by 2 233
Figure 18-10: Down-counting Waveform Diagram, Internal Clock Divided by 2 233
Figure 18-11: Down-counting Waveform Diagram, Update Event when Repetition Counter is not Used 234
Figure 18-12: Center-aligned Counter Timing Diagram, TIM_PCS = 0, TIM_ARR = 0x6 235
Figure 18-13: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Underflow) . 236
Figure 18-14: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Overflow) 236
Figure 18-15: Examples of Updating Rate in Different Modes and TIM_RCR Register Setting 237
Figure 18-16: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1 239
Figure 18-17: External Clock Connection Diagram 240
Figure 18-18: Timing Diagram in External Clock Source Mode 1 240
Figure 18-19: Timing Diagram in External Clock Source Mode 1 242
Figure 18-20: External Trigger Input Block Diagram 242
Figure 18-21: Timing 1 Diagram in External Clock Source Mode 2 243
Figure 18-22: Timing 2 Diagram in External Clock Source Mode 2 244
Figure 18-23: Capture/Compare Channel (Channel 1 Input Stage) 245
Figure 18-24: Capture/Compare Channel 1 Main Circuit 245
Figure 18-25: Output Stage of Capture/Compare Channel (Channels 1–3) 246
Figure 18-26: Output Stage of Capture/Compare Channel (Channel 4) 246
Figure 18-27: PWM Input Capture Mode Timing Diagram 248

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Figure 18-28: Output Compare Mode, Toggle on OC1 250
Figure 18-29: Edge-aligned PWM Waveform (ARR = 7) 251
Figure 18-30: Center-aligned PWM Waveform (ARR = 7) 252
Figure 18-31: Waveform of Complementary Output with Dead-time Insertion 253
Figure 18-32: Dead-time Waveform with Delay Greater than Negative Pulse 253
Figure 18-33: Dead-time Waveform with Delay Greater than Positive Pulse 253
Figure 18-34: Various Output Behavior in Response to A Break Event 254
Figure 18-35: 6-step PWM Generation, COM Example (OSSR = 1) 255
Figure 18-36: Timing Diagram ofOne-pulse Mode 256
Figure 18-37: Timing Diagram of ETR Signal Clearing OCxREF of TIM 258
Figure 18-38: Example of Counter Operation in Encoder Interface Mode 259
Figure 18-39: Timing Diagram in Reset Mode 260
Figure 18-40: Timing Diagram in Gated Mode 261
Figure 18-41: Timing Diagram in Trigger Mode 262
Figure 18-42: Timing Diagram in External Clock Mode 2 + Trigger Mode 263
Figure 19-1: System Block Diagram of TIM1–TIM4 & TIM8–TIM13 311
Figure 19-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 2 315
Figure 19-3: Counter Timing Diagram with Prescaler Division Changing from 1 to 4 316
Figure 19-4: Up-counting Waveform Diagram, Internal Clock not Divided 317
Figure 19-5: Up-counting Waveform Diagram, Internal Clock Divided by 2 317
Figure 19-6: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded) 318
Figure 19-7: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded) 318
Figure 19-8: Down-counting Waveform Diagram, Internal Clock not Divided 319
Figure 19-9: Down-counting Waveform Diagram, Internal Clock Divided by 2 320
Figure 19-10: Down-counting Waveform Diagram, Internal Clock Divided by 2 320
Figure 19-11: Down-counting Waveform Diagram, Update Event when Repetition Counter is not Used 321
Figure 19-12: Center-aligned Counter Timing Diagram, TIM_PCS = 0, TIM_ARR = 0x6 322
Figure 19-13: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Underflow) . 322
Figure 19-14: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Overflow) 323
Figure 19-15: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1 325
Figure 19-16: Example of External Clock Connection 325
Figure 19-17: Timing Diagram in External Clock Source Mode 1 326
Figure 19-18: Timing Diagram in External Clock Source Mode 1 327
Figure 19-19: External Trigger Input Block Diagram 327
Figure 19-20: Timing 1 Diagram in External Clock Source Mode 2 328
Figure 19-21: Timing 2 Diagram in External Clock Source Mode 2 329

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Figure 19-22: Capture/Compare Channel (Channel 1 Input Stage)..... 330
Figure 19-23: Capture/Compare Channel 1 Main Circuit 330
Figure 19-24: Output Stage of Capture/Compare Channel (Channels 1−3) 331
Figure 19-25: Output Stage of Capture/Compare Channel (Channel 4) 331
Figure 19-26: PWM Input Capture Mode Timing Diagram..... 333
Figure 19-27: Output Compare Mode, Toggle on OC1 335
Figure 19-28: Edge-aligned PWM Waveform (ARR = 7)..... 336
Figure 19-29: Center-aligned PWM Waveform (ARR = 7)..... 336
Figure 19-30: Timing Diagram of One-pulse Mode..... 337
Figure 19-31: ETR Signal Clearing OCxREF of TIM 339
Figure 19-32: Example of Counter Operation in Encoder Interface Mode..... 340
Figure 19-33: Timing Diagram in Reset Mode 342
Figure 19-34: Timing Diagram in Gated Mode 343
Figure 19-35: Timing Diagram in Trigger Mode..... 343
Figure 19-36: Timing Diagram in External Clock Mode 2 + Trigger Mode 344
Figure 20-1: Counter Timing Diagram with Prescaler Division Changing from 1 to 2 382
Figure 20-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 4 383
Figure 20-3: Up-counting Waveform Diagram, Internal Clock not Divided 384
Figure 20-4: Up-counting Waveform Diagram, Internal Clock Divided by 2..... 384
Figure 20-5: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded) 385
Figure 20-6: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded) 385
Figure 20-7: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1 387
Figure 21-1: System Block Diagram 395
Figure 23-1: WWDT Counter Refresh Timing Diagram 416
Figure 24-1 : BBU Module with Analog Module..... 422
Figure 24-2: Oscillator Calibration 429
Figure 25-1: Bit Period Structure of CAN..... 464
Figure 25-2: RXBUF Register Data Format..... 466
Figure 25-3: Bit Format Corresponding to Single Filter in Standard Frame..... 467
Figure 25-4: Bit Format Corresponding to Single Filter in Extended Frame..... 468
Figure 25-5: Bit Format Corresponding to Double Filters in Standard Frame 468
Figure 25-6: Bit Format Corresponding to Double Filters in Extended Frame..... 469
Figure 25-7: Arbitration Lost Bit Interrupt Diagram..... 471
Figure 26-1 : Recovery of SDA Using 9 SCL Clock Pulses 485
Figure 26-2 : Recovery of SDA Using 6 SCL Clock Pulses 486
Figure 26-3: Timing Diagram for I2C Master Mode TX_HOLD = 3 489
Figure 27-1: I2S Signal Waveform for 16-bit Channel..... 534

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Figure 27-2: I2S Signal Waveform for 32-bit Channel 535
Figure 29-1: LPUART Block Diagram 568
Figure 29-2: LPUART Interface Timing Diagram 569
Figure 29-3: LPUART Reception Timing Diagram 569
Figure 29-4: LPUART Transmission Timing Diagram 570
Figure 30-1: USART Character Transmitting 583
Figure 30-2: USART Transmitter Status 584
Figure 30-3: NRZ Code to Manchester Code 585
Figure 30-4: Preamble Signal Mode 586
Figure 30-5: Frame Start Delimiter 587
Figure 30-6: Bit Synchronization Diagram 588
Figure 30-7: Asynchronous Start Bit Detection 589
Figure 30-8: Character Reception 589
Figure 30-9: Asynchronous Start Bit Detection 590
Figure 30-10 : Preamble Mismatch 591
Figure 30-11: Manchester Error Flag 591
Figure 30-12: Manchester Encoded Character RF Transmission 592
Figure 30-13: ASK Modulator Output 593
Figure 30-14: FSK Modulator Output 594
Figure 30-15: Character Reception Timing in Synchronous Mode 594
Figure 30-16: Receiver Status 595
Figure 30-17: Parity Error Timing Diagram 596
Figure 30-18: Timing Diagram of Time Protection Operation 598
Figure 30-19: Timing Diagram of Frame Error Status 600
Figure 30-20: Transmission of Break 602
Figure 30-21: Hardware Handshake with Remote Devices 603
Figure 30-22: Receiver Operation Behavior When Hardware Handshake is Enabled 603
Figure 30-23: Transmitter Operation Behavior When Hardware Handshake is Enabled 604
Figure 30-24: Connection with IrDA Transceiver 604
Figure 30-25: IrDA Modulation 605
Figure 30-26: IrDA Demodulator Operation 607
Figure 30-27: SPI Transmission Format (CPHA = 1, 8 bits per transmission) 610
Figure 30-28: SPI Transmission Format (CPHA = 0, 8 bits per transmission) 611
Figure 30-29: Message Transmission 616
Figure 30-30: Message Reception 617
Figure 30-31: Sync Field 617
Figure 30-32: Timing Diagram of Slave Node Synchronization 619
Figure 30-33: Response Data Length Diagram 623

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Figure 30-34: USART Frame Slot Mode 625
Figure 30-35: Master Node Configuration, NACT = PUBLISH 628
Figure 30-36: Master Node Configuration, NACT = SUBSCRIBE 628
Figure 30-37: Master Node Configuration, NACT = IGNORE 629
Figure 30-38: Slave Node Configuration, NACT = PUBLISH 630
Figure 30-39: Slave Node Configuration, NACT = SUBSCRIBE 631
Figure 30-40: Slave Node Configuration, NACT = IGNORE 631
Figure 30-41: Master Node with DMA/PDC (PDCM = 1) 632
Figure 30-42: Master Node with DMA/PDC (PDCM = 0) 633
Figure 30-43: USART Slave Node Configuration 633
Figure 30-44: USART Normal Mode 635
Figure 30-45: USART Auto-response Mode 636
Figure 30-46: USART Local Loop-back Mode 636
Figure 30-47: USART Remote Loop-back Mode 636
Figure 31-1: Data Concatenation Example 1 (SPI_CCTL[12:8] = 5) 682
Figure 31-2: Data Concatenation Example 2 (SPI_CCTL[12:8] = 13) 683
Figure 31-3: Data Concatenation Example 3 (SPI_CCTL[12:8] = 28) 683
Figure 31-4: Transmission Timing Diagram of SPI in Different Modes in Motorola Mode 684
Figure 31-5: Transmission Timing Diagram of SPI in MODE2 in TI Mode 685
Figure 32-1: QSPI System Block Diagram 714
Figure 33-1: Double Buffer Structure Diagram 786
Figure 33-2: Chain Structure Diagram 787
Figure 33-3: DES0–DES3 Structure Diagram 787
Figure 35-1: USB Transfer Diagram 935
Figure 35-2: USB Endpoint In Transfer Flowchart 936
Figure 35-3: USB Endpoint Out Transfer Flowchart 937
Figure 36-1: ADC System Block Diagram 940
Figure 37-1: DAC System Block Diagram 991
Figure 38-1: ACMP System Block Diagram 1007
Figure 38-2: Basic Function Diagram of Comparator 1007
Figure 38-3: Voltage Comparison Function with Hysteresis 1008
Figure 39-1: OPA System Block Diagram 1015

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

1 Documentation Conventions

1.1 List of Abbreviations for Registers

The following abbreviations are used in register descriptions:

Read/write (R/W)	Software can read and write to this bit.
Read-only (R)	Software can only read this bit.
Write-only (W)	Software can only write to this bit, and reading this bit returns the reset value.
Read/clear write 1 (R/W1C)	Software can read as well as clear this bit by writing 1. Writing 0 has no effect on the bit value.
Read/clear write 0 (R/W0C)	Software can read as well as clear this bit by writing 0. Writing 1 has no effect on the bit value.

1.2 Glossary

This section outlines the definitions of acronyms and abbreviations used in this document:

- In this document, the Cortex-M4 core with FPU is referred to as Cortex-M4F.
- The CPU core integrates two debug ports:
  - JTAG debug port (JTAG-DP) provides a 5-pin standard interface based on the joint test action group (JTAG) protocol.
  - SWD debug port (SWD-DP) provides a 2-pin (clock and data) interface based on the serial wire debug (SWD) protocol.
- Word: Data/instruction of 32-bit length
- Half-word: Data/instruction of 16-bit length

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Byte: Data of 8-bit length
- Double-word: Data of 64-bit length
- IAP (in-application programming): IAP is the ability to re-program the Flash memory of a microcontroller while the user program is running.
- ICP (in-circuit programming): ICP is the ability to program the Flash memory of a microcontroller using the JTAG protocol, the SWD protocol or the bootloader while the device is mounted on the user application board.
- I-Code: this bus connects the instruction bus of the CPU core to the Flash memory instruction interface. Prefetching can be performed via this bus.
- D-Code: this bus connects the D-Code bus (literal load and debug access) of CPU to the Flash memory data interface.
- Option byte: production configuration bits stored in the Flash memory
- OBL: option byte loader
- AHB: Advanced high-performance bus
- CPU: Cortex-M4F

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

2 Product Introduction

2.1 Overview

The UM3241F / UM32417F series are general microprocessor chips based on the ARM® Cortex® -M4 core with high performance and low power. The M4 core implements a full set of DSP (digital signal processing) instructions and features a floating-point unit (FPU) and a memory protection unit (MPU). The maximum system clock frequency is 288MHz. It features an integrated FLASH memory with a maximum capacity of 1MB (512KB + 512KB), SRAM with a maximum capacity of 160KB, and DMA SRAM with a capacity of 32KB. The chips operate at a wide supply voltage of 1.8 V to 3.6 V and are available in the industrial temperature range from -40℃ to 105℃.

The UM3241F / UM32417F series chips feature a rich set of peripherals, including: 1 × USB full-speed device interface, 1 × SDIO/SD/eMMC interface, 1 × 10/100M/1000M Ethernet interface, 1 × external high-speed memory interface (EMC), 2 × 12-bit high-speed ADCs, 2 × 12-bit DACs, 1 × integrated temperature sensor, 3 × comparators, 2 × operational amplifiers, 2 × USARTs, 6 × UARTs, 4 × SPI interfaces, 3 × I2C interfaces, 2 × I2S interfaces, 2 × watchdog timers, 2 × CAN bus interfaces, 1 × QSPI interface, 14 × counters/timers (including advanced controllers and general-purpose timers), 1 × low-power UART (LPUART), 2 × low-power timers (LPTIMx), 1 × 32-bit RTC clock and counter, up to 82 general-purpose input/output (GPIO) channels, digital camera interface (DCMI) and I8080 LCD display interface, integrated hardware CORDIC module (supporting sin, cos, arctan, square root, multiplication, and division), built-in encryption/decryption engine (including AES, SHA, etc.), 1 × random number generator (RNG) that can produce random keys.

Applications:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Motor, IoT, etc.

2.2 Main Features

- **ARM Cortex™-M4F core with high performance**
  - Main frequency up to 288 MHz
  - Memory protection unit (MPU)
  - 16 programmable interrupt priority levels
  - IEEE 754 compliant FPU
  - DSP instruction extension with hardware divider and 32-bit single-cycle hardware multiplier
  - 512 KB Flash and 160 KB SRAM + 4 KB backup SRAM + 32 KB SRAM3
  - Two DMA controllers with 16 channels in total
  - Online debugging using SWD/JTAG protocol
- **Bootloader**
  - Supporting ISP and IAP for Flash memory
- **Flexible clock source**
  - External crystal input: 1 MHz–48 MHz
  - Internal 48 MHz RCH oscillator
  - Internal 32 kHz RCL oscillator
  - Internal PLL (supporting integer-N, fractional-N and spread spectrum modes)
  - External 32.768 kHz XTL and 32 kHz RTC

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- **Analog peripherals**
  - 2 x 12-bit successive ADCs with 16 external channels, supporting single-ended and differential inputs
  - 2 x 12-bit 1 Msps DACs with buffer
  - 3 x analog comparators (ACMP)
  - 2 x operational amplifiers (OPA) available in PGA or comparator mode
  - Internal voltage reference (VREF)
  - Internal temperature sensor (TS)
- **Hardware acceleration co-processor**
  - CORDIC co-processor supports functions as follows: $m \cdot \sin\theta$, $m \cdot \cos\theta$, atan2(y, x), $\sqrt{x^2+y^2}$, $y \cdot x$, y/x, $\sinh w$, $\cosh w$, $tanh^{-1}(y/x)$, ln(x), $\sqrt{x}$,etc.
- **Rich communication interface**
  - 6 x UARTs (up to 10.5 Mbps) supporting IrDA
  - 2 x USARTs supporting UART / SPI / IrDA / LIN
  - 1 x LPUART
  - 3 x I2Cs (up to 1 Mbps), SMBus compliant
  - 2 x I2S, supporting I2S / PCM standard
  - 4 × general-purpose SPIs
  - 1 × QSPI

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- 2 x CAN industrial buses, CAN2.0B compliant
- 1 x 48 MHz SDIO/SD/eMMC interface
- 1 x 10/100/1000 M Ethernet RGMII/RMII/MII interface
- 1 x full-speed USB20 device interface (with PHY)

- **External memory interface**
  - 1 x EMC high-speed memory controller, interface with SRAM/NOR Flash

- **Encryption engine**
  - Supporting AES 128/256
  - Supporting SHA 256
  - A random number generator (RNG) can generate random key.

- **Timer**
  - 2 x 32-bit advanced timers (eQCT™: TIM1 and TIM8), including a 32-bit auto-reload counter and a programmable prescaler, supporting input capture, output compare, PWM output (complementary PWM with dead-time insertion) Each advanced timer supports 4 PWM outputs; 8 PWM outputs in total.
  - 10 x 32-bit general timers (TIM2 ~ 5, TIM9 ~ 14), each with 4 PWM outputs, 40 PWM outputs in total
  - 2 x 32-bit basic timers (TIM6 and TIM7)
  - 2 x 16-bit low-power timers (LPTIM1 ~ 2), with 2 PWM outputs in total
  - 1 x 24-bit SysTick timer

- **Real-time clock (RTC)**

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Calendar (with seconds, minutes, hours, weekday, date, month and year in BCD format), and calibration supported

- **GPIO**
  - Up to 82 GPIOs
  - 58 I/Os with 5-V tolerant capability
- **Video interface**
  - Interface with TFT LCD modules, supporting Intel 8080 mode
  - Digital camera interface (DCMI)
- **Power Management**
  - Low-power modes: Sleep, Stop, Standby and DeepStandby
  - Born-out reset (BOR) and low-voltage detection (LVD), with under-voltage interrupt and forced reset generated from two sets of detection points respectively
  - Power-on reset (POR) and power-down reset (PDR)
  - Integrated power management unit (PMU)
- **Electrical characteristics**
  - Operating voltage: 1.8 ~ 3.6 V
  - Operating temperature: -40 ~ +105℃
- **Security**
  - Anti-copy board to prevent programs in eFlash from being pirated
  - CRC16-CCITT / CRC32 hardware accelerator

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- 128-bit UUID

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3 Memory and Bus Architecture

3.1 System Architecture

The main system consists of 32-bit multilayer AHB bus matrix that interconnects:

- **8 masters:**
  - Cortex™-M4F core I-bus, D-bus and S-bus
  - DMA1
  - DMA2
  - SDIO DMA
  - Ethernet DMA
  - DCMI controller DMA
- **12 slaves:**
  - FLASH NVR
  - Internal Flash memory on I-code bus
  - Internal Flash memory on D-code bus
  - Main internal SRAM1 (128 KB)
  - Auxiliary internal SRAM2 (32 KB)
  - DMA SRAM3 (32 KB)
  - AHB0 peripheral
  - AHB1 peripheral
  - AHB2APB1 peripheral
  - AHB2APB2 peripheral
  - EMC peripheral
  - QSPI

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3.2 Bus Architecture Diagram

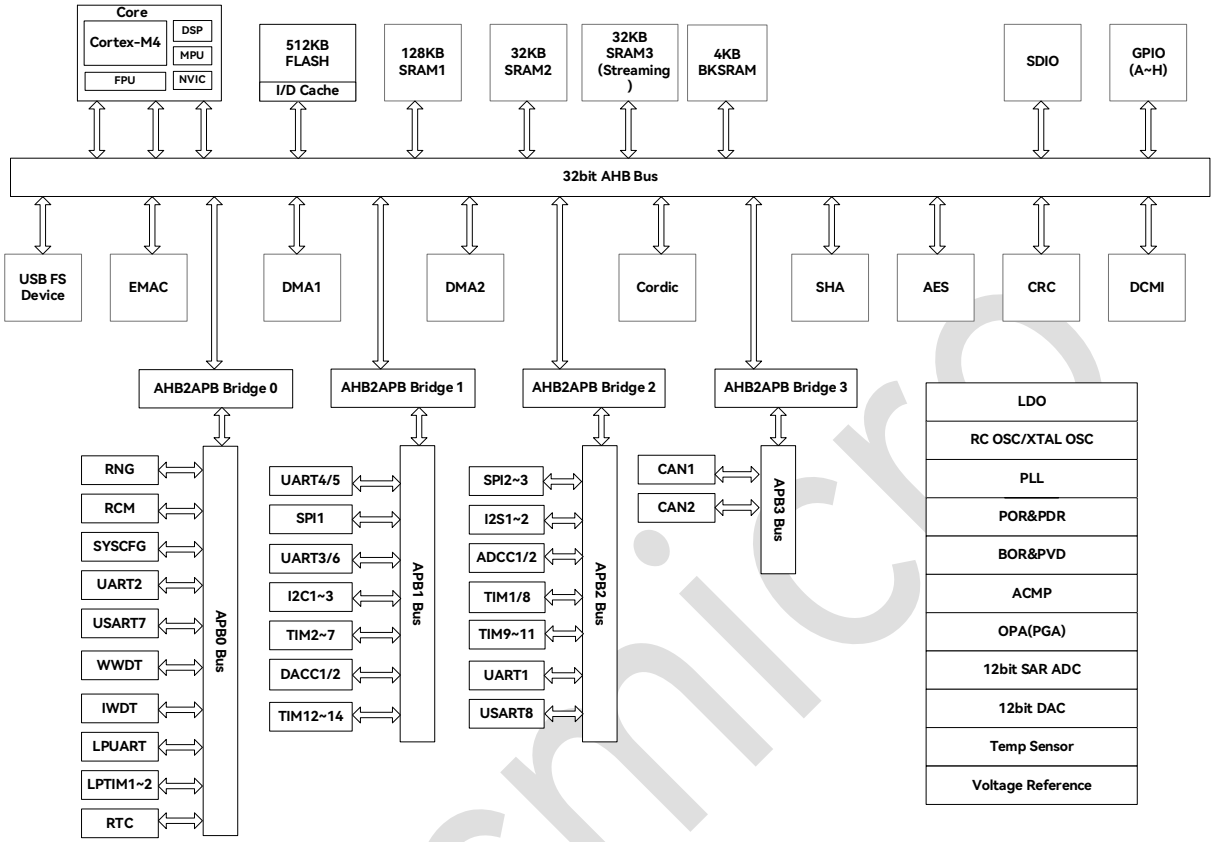

Figure 3-1: Bus Architecture Diagram

3.3 Memory Mapping

Program memory, data memory, registers and I/O ports are organized within the same linear 4-GB address space.

The bytes are coded in memory in little-endian format. The lowest numbered byte in a word is considered the word's least significant byte and the highest numbered byte the most significant.

For the detailed mapping of peripheral registers, please refer to the related chapters.

All the memory map areas that are not allocated to on-chip memories and peripherals are considered "reserved".

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

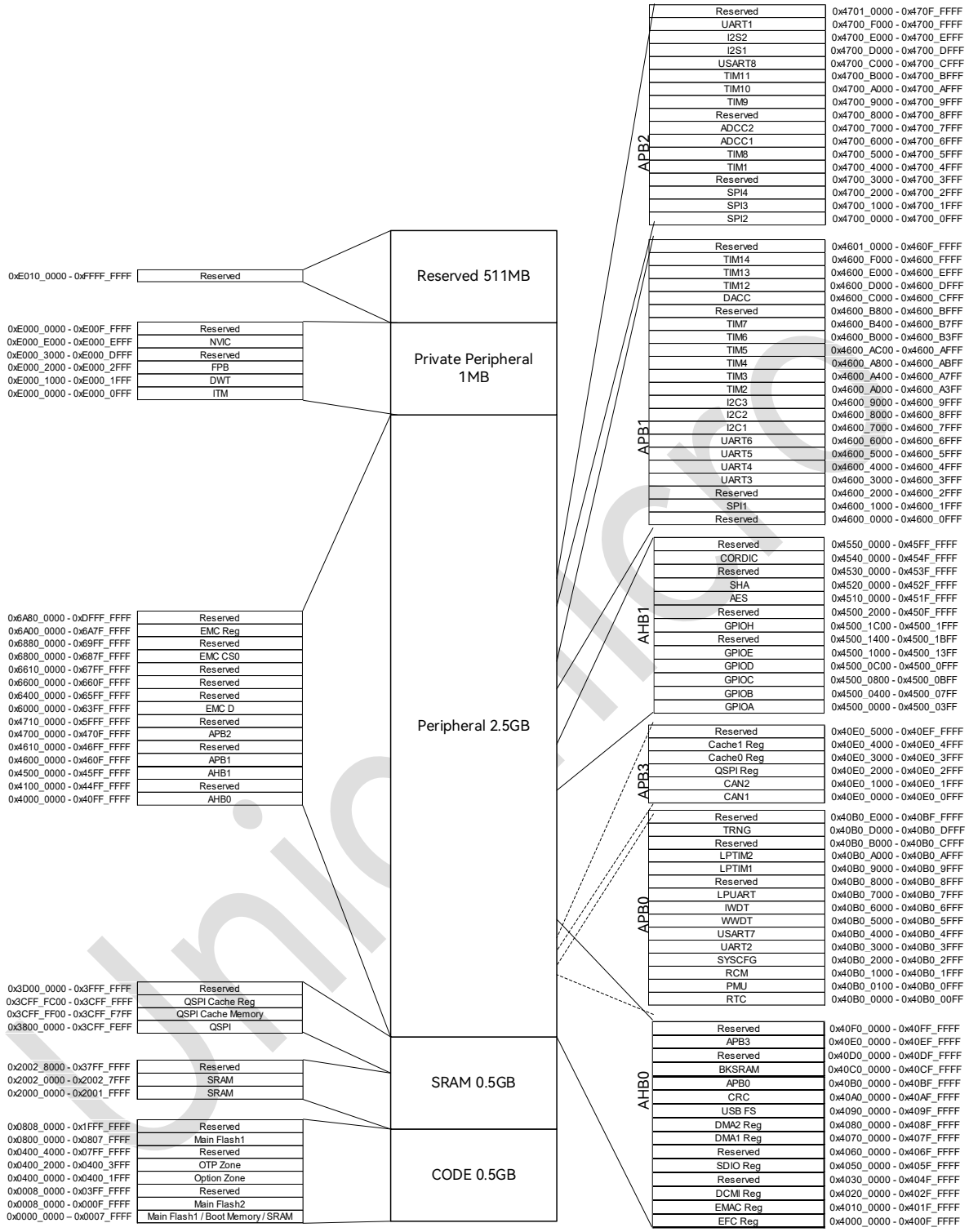

Figure 3-2: Memory Address Mapping

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 3-1: Memory Remapping Address

Block / Module	Boundary Address	Size
Main FLASH 1	0x0000_0000–0x0007_FFFF 0x0800_0000–0x0807_FFFF	512 KB
SRAM1	0x2000_0000–0x2001_FFFF	128 KB
SRAM2	0x2002_0000–0x2002_7FFF	32 KB
SRAM3 (DMA)	0x2002_8000–0x2002_FFFF	32 KB
EMC_D	0x6000_0000–0x63FF_FFFF	16 MB
EMC_CS0	0x6800_0000–0x687F_FFFF	8 MB
EMC_Config	0x6A00_0000–0x6A7F_FFFF	8 MB
QSPI	0x0008_0000–0x03FF_FFFF 0x3800_0000–0x3CFF_FFFF	16 MB
QSPICACHE_DATA_MEM	0x3CFF_F000–0x3CFF_F7FF	-
QSPICACHE_TAG_MEM	0x3CFF_F800–0x3CFF_FBFF	-
QSPICACHE_REG	0x3CFF_FC00–0x3CFF_FFFF	-

Table 3-2: Memory Mapping Address

Block / Module		Boundary Address
AHB0 peripheral	EFC	0x4000_0000–0x400F_FFFF
	EMAC	0x4010_0000–0x401F_FFFF
	DCMI	0x4020_0000–0x402F_FFFF
	SDIO	0x4050_0000–0x405F_FFFF
	DMA1	0x4070_0000–0x407F_FFFF
	DMA2	0x4080_0000–0x408F_FFFF
	USB FS device	0x4090_0000–0x409F_FFFF
	CRC	0x40A0_0000–0x40AF_FFFF
	BKSRAM	0x40C0_0000–0x40CF_FFFF
AHB1 peripheral	GPIOA	0x4500_0000–0x4500_03FF
	GPIOB	0x4500_0400–0x4500_07FF
	GPIOC	0x4500_0800–0x4500_0BFF
	GPIOD	0x4500_0C00–0x4500_0FFF
	GPIOE	0x4500_1000–0x4500_13FF
	GPIOH	0x4500_1C00–0x4500_1FFF
	AES	0x4510_0000–0x451F_FFFF
	SHA	0x4520_0000–0x452F_FFFF
	CORDIC	0x4540_0000–0x454F_FFFF
AHB2APB0 bridge	RTC	0x40B0_0000–0x40B0_00FF

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Block / Module		Boundary Address
	PMU	0x40B0_0100–0x40B0_0FFF
	RCM	0x40B0_1000–0x40B0_1FFF
	SYSCFG	0x40B0_2000–0x40B0_2FFF
	UART2	0x40B0_3000–0x40B0_3FFF
	USART7	0x40B0_4000–0x40B0_4FFF
	WWDT	0x40B0_5000–0x40B0_5FFF
	IWDT	0x40B0_6000–0x40B0_6FFF
	LPUART	0x40B0_7000–0x40B0_7FFF
	LPTIM1	0x40B0_9000–0x40B0_9FFF
	LPTIM2	0x40B0_A000–0x40B0_AFFF
	RNG	0x40B0_D000–0x40B0_DFFF
AHB2APB1 bridge	SPI1	0x4600_1000–0x4600_1FFF
	UART3	0x4600_3000–0x4600_3FFF
	UART4	0x4600_4000–0x4600_4FFF
	UART5	0x4600_5000–0x4600_5FFF
	UART6	0x4600_6000–0x4600_6FFF
	I2C1	0x4600_7000–0x4600_7FFF
	I2C2	0x4600_8000–0x4600_8FFF
	I2C3	0x4600_9000–0x4600_9FFF
	TIM2 ~ 5	0x4600_A000–0x4600_AFFF
	TIM6~ 7	0x4600_B000–0x4600_BFFF
	DAC	0x4600_C000–0x4600_CFFF
	TIM12	0x4600_D000–0x4600_DFFF
	TIM13	0x4600_E000–0x4600_EFFF
	TIM14	0x4600_F000–0x4600_FFFF
AHB2APB2 bridge	SPI2	0x4700_0000–0x4700_0FFF
	SPI3	0x4700_1000–0x4700_1FFF
	SPI4	0x4700_2000–0x4700_2FFF
	TIM1	0x4700_4000–0x4700_4FFF
	TIM8	0x4700_5000–0x4700_5FFF
	ADC1	0x4700_6000–0x4700_6FFF
	ADC2	0x4700_7000–0x4700_7FFF
	TIM9	0x4700_9000–0x4700_9FFF
	TIM10	0x4700_A000–0x4700_AFFF
	TIM11	0x4700_B000–0x4700_BFFF
	USART8	0x4700_C000–0x4700_CFFF

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Block / Module		Boundary Address
	I2S1	0x4700_D000–0x4700_DFFF
	I2S2	0x4700_E000–0x4700_EFFF
	UART1	0x4700_F000–0x4700_FFFF
AHB2APB3 bridge	CAN1	0x40E0_0000–0x40E0_0FFF
	CAN2	0x40E0_1000–0x40E0_1FFF
	QSPI	0x40E0_2000–0x40E0_2FFF
	EFC ICache	0x40E0_3000–0x40E0_3FFF
	EFC DCache	0x40E0_4000–0x40E0_4FFF

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4 Memory System

4.1 FLASH

4.1.1 Overview

The Flash memory module features the capacity of up to 1MB (512 KB of on-chip Flash + 512 KB of FLASH), and the Flash controller (EFC) performs read, write, erase and other operations on eFlash with the cooperation of CPU.

4.1.2 Main Features

- 8/16/32-bit Flash memory read operation, 32-bit Flash memory program operation
- On-chip Flash with a capacity of 512 KB has 64 pages in total, each of 8 KB
- Page and chip erase
- Flash controller supports continuous programming, saving time when programming data into a 512-byte aligned memory.
- Configurable read waiting time
- Erase/write protection
- Automatic bus locking
- Automatic read-back verification after programming
- OTP (one-time programmable) area
- Instruction and data cache

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.1.3 Functional Description

Flash controller can perform read, write, erase and other operations.

It requires a frequency greater than 1 MHz for erasing. The user shall configure the EFC_TIME register according to the actual operating frequency.

Flash can be programmed with 32-bit wide data, which shall be unlocked before programming.

Flash controller supports continuous programming, saving time when programming data into a 512-byte aligned memory. Continuous programming shall be performed on memory other than this Flash.

The controller features erase verification function that can automatically check whether all data bits in the Flash are changed to 1 in page erase and mass erase modes.

4.1.3.1 Flash Bank 1 Address Mapping

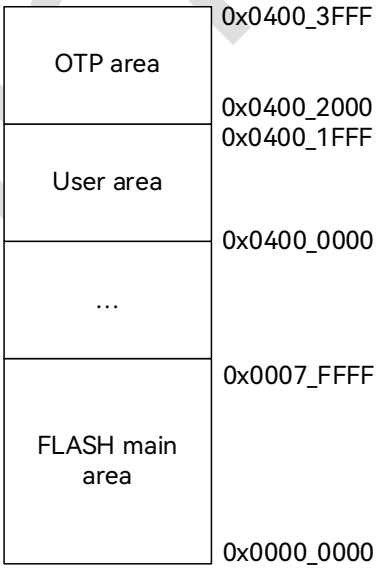

Figure 4-1: EFC Memory Address Mapping

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.1.3.2 User Option Bytes

The address range 0x0400_0000 to 0x0400_1FFF is designated as the user option page, where some addresses have special functions, as described in the following table:

Table 4-1: EFC Special Feature Address Description

Name	Address	Bit Width	Content
WRITE_PROTECT & IWDT_HW_START	0x04001FA0	32 bits	Pages in the specified numbering interval (0–63) in the Flash main area are prohibited from being erased. Bit [31:24]: A5: Write protection activated Others: Write protection not activated Bit [23:14]: Reserved Bit [13:8]: Ending number of specified protection page Bit [7:6]: Reserved Bit [5:0]: Starting number of specified protection page
DEBUG_DISABLE	0x04001FB0	8 bits	Disable JTAG or SWG debug port operation: 0xAB: debug port disabled, JTAG will not be recognized Others: debug port operation not disabled

4.1.3.3 One-time Programmable (OTP) Area

Table 4-2: OTP Area of EFC

No.	OTP Storage Area Address	Size	OTP Lock Control Address	Remarks
0	0x04002000–0x040020FF	256 bytes	0x04003F80	-
1	0x04002100–0x040021FF	256 bytes	0x04003F84	-
2	0x04002200–0x040022FF	256 bytes	0x04003F88	-
3	0x04002300–0x040023FF	256 bytes	0x04003F8C	-
4	0x04002400–0x040024FF	256 bytes	0x04003F90	-
5	0x04002500–0x040025FF	256 bytes	0x04003F94	-
6	0x04002600–0x040026FF	256 bytes	0x04003F98	-

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

No.	OTP Storage Area Address	Size	OTP Lock Control Address	Remarks
7	0x04002700–0x040027FF	256 bytes	0x04003F9C	-
8	0x04002800–0x040028FF	256 bytes	0x04003FA0	-
9	0x04002900–0x040029FF	256 bytes	0x04003FA4	-
10	0x04002A00–0x04002AFF	256 bytes	0x04003FA8	-
11	0x04002B00–0x04002BFF	256 bytes	0x04003FAC	-
12	0x04002C00–0x04002CFF	256 bytes	0x04003FB0	-
13	0x04002D00–0x04002DFF	256 bytes	0x04003FB4	-
14	0x04002E00–0x04002EFF	256 bytes	0x04003FB8	-
15	0x04002F00–0x04002FFF	256 bytes	0x04003FBC	-
16	0x04003000–0x040030FF	256 bytes	0x04003FC0	-
17	0x04003100–0x040031FF	256 bytes	0x04003FC4	-
18	0x04003200–0x040032FF	256 bytes	0x04003FC8	-
19	0x04003300–0x040033FF	256 bytes	0x04003FCC	-
20	0x04003400–0x040034FF	256 bytes	0x04003FD0	Reserved area
21	0x04003500–0x040035FF	256 bytes	0x04003FD4	Reserved area
22	0x04003600–0x040036FF	256 bytes	0x04003FD8	Reserved area
23	0x04003700–0x040037FF	256 bytes	0x04003FDC	Reserved area
24	0x04003800–0x040038FF	256 bytes	0x04003FE0	-
25	0x04003900–0x040039FF	256 bytes	0x04003FE4	-
26	0x04003A00–0x04003AFF	256 bytes	0x04003FE8	-
27	0x04003B00–0x04003BFF	256 bytes	0x04003FEC	Reserved area
28	0x04003C00–0x04003CFF	256 bytes	0x04003FF0	Reserved area
29	0x04003D00–0x04003DFF	256 bytes	0x04003FF4	Reserved area
30	0x04003E00–0x04003EFF	256 bytes	0x04003FF8	-
31	0x04003F00–0x04003F7F	128 bytes	-	Reserved area
32	0x04003F80–0x04003FFF	128 bytes	-	OTP lock control address

Notes:

- Programming the first byte of the lock control address to 8'h00 locks the corresponding storage area so that it cannot be programmed.
- The OTP area is not erasable.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.1.4 Register Description

Register base address: 0x4000_0000

The registers are listed below:

Table 4-3: List of EFC Registers

Offset Address	Name	Description
0x00	EFC_CTRL	Control register
0x04	EFC_TIME	Timing register
0x08	EFC_SEC	Security operation register
0x0C	EFC_STATUS	Status register
0x10	EFC_INTSTATUS	Interrupt status register
0x14	EFC_INTEN	Interrupt enable register
0x18	EFC_LPCR	Low-power control register
0x1C	EFC_ADDRREC	Last operation address register

Registers are detailed in the following sections.

4.1.4.1 Control Register (EFC_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	ERASEVE	R/W	0x0	Erase verification enable: 1: Enabled 0: Disabled
4	PROGVE	R/W	0x0	Programming verification enable (valid only in single programming): 1: Enabled 0: Disabled
3	MERASES	R/W	0x0	Mass erase mode (erase all pages in main memory) selection bit: 1: Selected 0: Not selected

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	PERASES	R/W	0x0	Page erase mode selection bit: 1: Selected 0: Not selected
1	CONPROGS	R/W	0x0	Continuous programming mode selection bit: 1: Selected 0: Not selected
0	PROGS	R/W	0x0	Single programming mode selection bit: 1: Selected 0: Not selected

Note: Only one of bits 0–3 can be selected.

4.1.4.2 Timing Register (EFC_TIME)

Offset address: 0x04

Reset value: 0x0000 6018

Bit	Name	Attribute	Reset Value	Description
31:24	REGWE	W	0x0	This register is writable when this field is written with A5. Please write 0xA50XXXXX to this register once.
23:18	RSV	-	-	Reserved
17:16	RECYC	R/W	0x0	Waiting time between two reads
15:12	RWAITCYC	R/W	6	Setting bit of read wait cycle, which can be larger, refer to the table below.
11: 9	RSV	-	-	Reserved
8:0	FREQ	R/W	24	Erase time scale, please fill in (EFC operating frequency - 1), in MHz. The minimum value is 0, which corresponds to the minimum frequency of 1 MHz required for erasing. (Withstand voltage: -10%–+30%)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 4-4: Relationship between Read Wait Cycle and Frequency at V_{DD} = 1.1 V

Frequency/MHz	Read Wait Cycle
< 50	0
50–100	1
100–150	2
150–200	3
200–250	4
250–300	5
…	…

4.1.4.3 Security Operation Register (EFC_SEC)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	UNLOCK	R/W	0x0	This register shall be unlocked when programming/erasing Flash. Writing 0x55AAAA55 to this register unlocks it, and after unlocking, reading this register will yield a value of 1. Writing any other value will relock it. The register will also be relocked when programming/erasing begins.

4.1.4.4 Status Register (EFC_STATUS)

Offset address: 0x0C

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	VDDS	R	0x0	VDDH status (LVD) indication bit: 0: Normal status 1: Low voltage warning

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7: 6	RSV	-	-	Reserved
5:4	LPS	R	0x0	Low-power mode status indication bit: 11: Being in power-down mode 10: Being in sleep mode 01: Operating at low voltage 00: Operating at high voltage
3	RSV	-	-	Reserved
2	CONPROGRDY	R/W	0x0	Flash continuous programming status indication bit (writing 0 to this bit can exit the continuous programming mode): 1: Waiting for the next continuous programming 0: Not in the state of waiting for the next continuous programming
1	RSV	-	-	Reserved
0	RDYS	R	0x1	Flash status indication bit: 1: Flash being idle 0: Flash being busy

4.1.4.5 Interrupt Status Register (EFC_INTSTATUS)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3	EVFS	R/W1C	0x0	Erase verification failure status bit: Writing 1 clears this bit. If interrupt is enabled, an interrupt will be generated. 0: Erase verification failed, not all data in Flash is 1 0: No erase verification failure occurred

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	PVFS	R/W1C	0x0	Write verification failure status bit: Writing 1 clears this bit. If interrupt is allowed, an interrupt will be generated. 0: Write verification failed, the data in Flash does not match the written data. 0: No write verification failure occurred
1	VDDLS	R/W1C	0x0	Low VDDH (LVD) interrupt status bit: Writing 1 clears this bit. If interrupt is enabled, an interrupt will be generated. 1: Low VDDH (LVD) warning occurred 0: Normal status
0	OPDS	R/W1C	0x0	Program/erase done interrupt status bit: Writing 1 clears this bit. If interrupt is enabled, an interrupt will be generated. 1: Program/erase done 0: Program/erase undone

4.1.4.6 Interrupt Enable Register (EFC_INTEN)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3	EVFE	R/W	0x0	Erase verification failure interrupt enable: 1: Enabled 0: Disabled
2	PVFE	R/W	0x0	Programming verification failure interrupt enable: 1: Enabled 0: Disabled
1	VDDLE	R/W	0x0	Low VDDH (LVD) interrupt enable: 1: Enabled 0: Disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	OPDE	R/W	0x0	Programming/erasing done interrupt enable: 1: Enabled 0: Disabled

4.1.4.7 Low-power Control Register (EFC_LPCR)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	REGWE	W	0x0	This register is writable when this field is written with A5. Please write 0xA500000X to this register once.
23:1	RSV	-	-	Reserved
0	MODES	R/W	0x0	Selection bit for Flash behavior when the system enters low-power mode: 1: Entering power-down mode 0: Entering sleep mode Note: In stop mode, both modes can be selected. In standby mode, it is recommended to be set as 1.

4.1.4.8 Last Operation Address Register (EFC_ADDRREC)

Offset address: 0x1C

Reset value: 0x0C07 FFFF

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	-	-	Reserved
20:0	ADDR	R	0xC07FFFF	Record the address of the most recent erase or program operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.1.5 Operation Procedure

4.1.5.1 Timing Setting

When using Flash, no matter what operation is performed, it is required to set the EFC_TIME register first to make the control value in the correct range.

1. Confirm according to the system frequency that the value (system frequency - 1) to be entered into the FREQ field is correct before using the program, erase, and low-power mode functions (system frequency is not required to be greater than 1 MHz, just fill in 0 if it is lower than 1 MHz).
2. Confirm according to the system frequency that the value to be entered into the RWAITCYC (Read_Wait_Cycle) is not less than the required value. To increase the system frequency, the EFC_TIME register shall be modified before increasing the system frequency; while to decrease the system frequency, the system frequency shall be decreased before modifying the EFC_TIME register.
3. Writing the EFC_TIME register once requires a combination of writing eigenvalues to the REGWE (Reg_Wr_Enable) field.

4.1.5.2 Single Programming

1. Operate __set_PRIMASK() to disable the general interrupt.
2. Configure the EFC_CTRL register to select single programming mode with the option to enable programming verification.
3. Unlock the EFC_SEC register and write 0x55AAAA55 at a time.
4. Write programming data to the address to be programmed.
5. Wait for EFC_STATUS[0] to be idle.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Reset the EFC_CTRL register to its default value.
7. Operate __set_PRIMASK() to enable the general interrupt.
8. Operate the EFC_SEC register to lock the Flash and write 0x55AA0000 at a time.

4.1.5.3 Continuous Programming

Programming can be carried out multiple times within the range of 512 bytes, and the control program can be really accelerated only when it is running in memory other than Flash.

1. Configure the EFC_CTRL register to select continuous programming mode.
2. Unlock the EFC_SEC register and write 0x55AAAA55 at a time.
3. Write programming data to the address to be programmed.
4. Repeat steps 2 and 3 as required.
5. Upon completion of the last programming, write 0 to the EFC_STATUS register to end continuous programming and prevent waiting in continuous programming mode.
6. Reset the EFC_CTRL register to its default value.
7. Operate the EFC_SEC register to lock the Flash and write 0x55AA0000 at a time.

4.1.5.4 Page Erase

1. Operate _set_PRIMASK() to disable the general interrupt.
2. Configure the EFC_CTRL register to select page erase mode.
3. Unlock the EFC_SEC register and write 0x55AAAA55 at a time.
4. Write arbitrary data to any address in the page to be erased.
5. Wait for EFC_STATUS[0] to be idle.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Reset the EFC_CTRL register to its default value.
7. Operate the EFC_SEC register to lock the Flash and write 0x55AA0000 at a time.
8. Operate _set_PRIMASK() to enable the general interrupt.

4.1.5.5 Mass Erase

Mass erase will erase all pages in the Flash main memory.

1. Operate _set_PRIMASK() to disable the general interrupt.
2. Configure the EFC_CTRL register to select mass erase mode.
3. Unlock the EFC_SEC register and write 0x55AAAA55 at a time.
4. Write arbitrary data to any address in the main memory.
5. Wait for EFC_STATUS[0] to be idle.
6. Reset the EFC_CTRL register to its default value.
7. Operate the EFC_SEC register to lock the Flash and write 0x55AA0000 at a time.
8. Operate _set_PRIMASK() to enable the general interrupt.

4.2 SRAM

4.2.1 Overview

SRAM is mainly used for code running, storing variables and data or stacks during program execution, with a maximum capacity of 192 KB (including SRAM1 with a capacity of 128KB, supporting 0-wait access, SRAM1 (32 KB) and SRAM2 (32 KB) require one cycle for access) and a maximum address range of 0x2000_0000−0x2002_7FFF.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.2.2 Main Features

- Read and write access supported by bytes, half-words or full-words
- DMA available

4.3 CACHE

The cache controller can cache the data fetched from Flash and accelerate the next read for higher system performance. This cache is divided into instruction cache (I-Cache) with a size of 4 kB and data cache (D-Cache) with a size of 256 B. The cache controller can cache the data fetched from Flash and accelerate the next read. It is optionally to enable cache line prefetch, and the cache area will be automatically initialized when the cache is activated.

4.3.1 Register Description

The instruction and data cache control registers are as follows:

I-Cache (instruction cache) register base address: 0x40E0_3000

D-Cache (data cache) register base address: 0x40E0_4000

The I-Cache and D-Cache registers are listed below:

Table 4-5: List of Cache Registers

Offset Address	Name	Description
0x00	CACHE_CTRL	Control register
0x04	CACHE_STATUS	Status register
0x14	CACHE_HITCNT	Hit counter register
0x18	CACHE_MISSCNT	Miss counter register

Registers are detailed in the following sections.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4.3.1.1 Control Register (CACHE_CTRL)

Offset address: 0x00

Reset value: 0x0000 0040

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved
6	STATIS_EN	R/W	0x1	Statistical counter enable: 0: Disabled 1: Enabled
5	PREFETCH	R/W	0x0	Prefetch enable: 0: Disabled 1: Enabled
4:1	RSV	-	-	Reserved
0	CACHE_EN	R/W	0x0	Cache enable: 0: Disabled 1: Enabled

4.3.1.2 Status Register (CACHE_STATUS)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2	INV_STATUS	R	0x0	Cache being initialized
1:0	CACHE_STATUS	R	00	Cache enable status: 00: Disabled 01: Being enabled 10: Enabled 11: Being disabled

4.3.1.3 Hit Counter Register (CACHE_HITCNT)

Offset address: 0x14

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	HIT_CNT	R/W	0x0	Counting cache hits

4.3.1.4 Miss Counter Register (CACHE_MISSCNT)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MISS_CNT	R/W	0x0	Counting cache misses

4.3.2 Use Process

4.3.2.1 Enabling Cache

1. (Optional) Configure the prefetch (CACHE_CTRL[5]) when the cache is not enabled, note that enabling prefetch may affect the efficiency.
2. Enable the cache (CACHE_CTRL[0]).
3. Query the cache status (CACHE_STATUS[1:0]) until it is displayed as enabled.

4.3.2.2 Updating Flash Content

If the content stored in Flash is updated after the cache is enabled, inconsistencies will occur and the cache shall be restarted to empty the cache space.

1. Disable the cache (CACHE_CTRL[0]).
2. Query the cache status (CACHE_STATUS[1:0]) until it is displayed as disabled.
3. Enable the cache (CACHE_CTRL[0]).
4. Query the cache status (CACHE_STATUS[1:0]) until it is displayed as enabled.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5 Power Management Unit (PMU)

5.1 Chip Power Supply

The chip is supplied with single power supply and VBAT backup power supply. It requires an external operating supply voltage between 1.8 V and 3.6 V, and a digital circuit operating voltage generated by the built-in LDO.

The system power supply scheme is shown below.

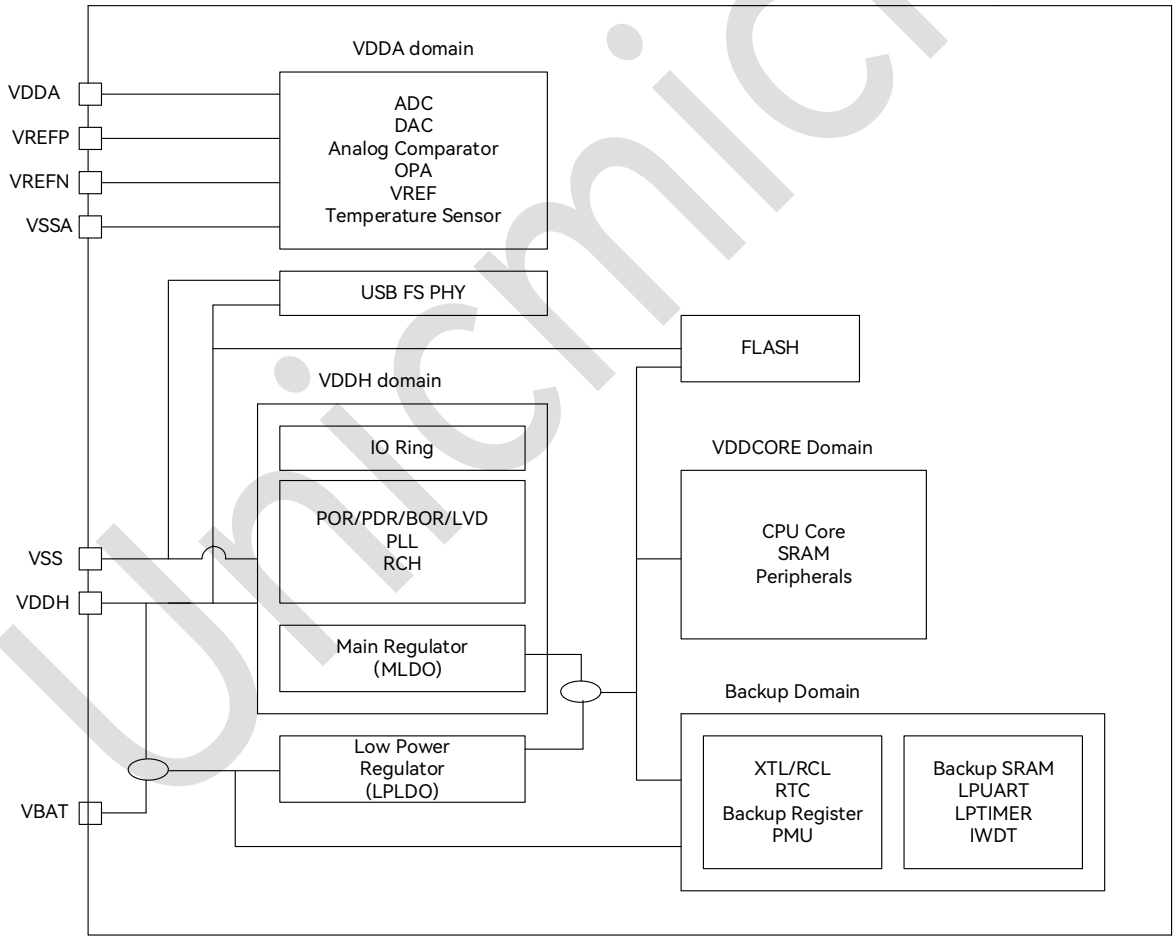

Figure 5-1: System Power Supply Scheme

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5.2 Internal PMU Module Function

5.2.1 Introduction to Internal PMU Module

The chip provides different power consumption modes based on application scenarios to reduce the overall power consumption.

- Run mode
- Sleep mode
- Stop mode
- Standby mode
- DeepStandby mode

The low-power modes and wake-up information are shown in the following table:

Table 5-1: Low-power Modes Summary

Power Mode	Description	Entry Condition	Wakeup Source
Run mode	All supplies are powered on, and the high-speed clock remains active.	-	-
Sleep mode	All supplies are powered on, with the M4 core high-speed clock off and other peripherals on.	1. PMU_STANDBY_EN = 0, PMU_MODE[1:0] = 00, PMU_BKSRAMOFF = 0 2. EFC_Sys_Mode = 0 3. WFI/WFE	1. Wakeup by any GPIO interrupt 2. Wakeup by any peripheral interrupt 3. Wakeup by reset (RESETN, IWDT reset)
Stop mode	All supplies are powered on, with the high-speed clock off and the	1. PMU_STANDBY_EN = 0, PMU_MODE[1:0] = 01, PMU_BKSRAMOFF = 0 2. EFC_Sys_Mode = 0	1. Wakeup by any GPIO interrupt 2. Wakeup by RTC alarm interrupt, RTC

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Power Mode	Description	Entry Condition	Wakeup Source
	32K low-speed clock on.	3. Set SLEEPDEEP = 1 for M4 core. 4. WFI/WFE	timestamp and tamper interrrupts 3. Wakeup by IWDT reset or interrupt 4. Wakeup by external LPUART 5. Wakeup by LPTIM1 ~ 2 timing
Standby0 mode	The CORE domain is powered off, the BBU domain is kept on, and the 32K low-speed clock remains active. BKSRAM / IWDT / LPTIM / LPUART are powered on.	1. PMU_STANDBY_EN = 1, PMU_MODE[1:0] = 10, PMU_BKSRAMOFF = 0 2. EFC_Sys_Mode = 1 3. Set SLEEPDEEP = 1 for M4 core. 4. WFI/WFE	1. Wakeup by external pins PA0, PA2, PC0, PC2 and PC3 2. Wakeup by RTC alarm interrupt, RTC timestamp and tamper interrrupts 3. Wakeup by LPTIM1 ~ 2 timing 4. Wakeup by LPUART (only PC2) 5. Wakeup by IWDT reset or interrupt 6. Wakeup by reset (RESETN)
Standby1 mode	The CORE domain is powered off, the BBU domain is kept on, and the 32K low-speed clock remains active. BKSRAM / IWDT / LPTIM / LPUART are powered off.	1. PMU_STANDBY_EN = 1, PMU_MODE[1:0] = 10, PMU_BKSRAMOFF = 1 2. EFC_Sys_Mode = 1 3. Set SLEEPDEEP = 1 for M4 core. 4. WFI/WFE	1. Wakeup by external pins of PA0, PA2, PC0, PC2, PC3 and PC13 2. Wakeup by RTC alarm interrupt, RTC timestamp and tamper interrrupts 3. Wakeup by reset (RESETN)
DeepStandby0 mode	The CORE domain is powered off, the BBU domain is kept on, and the	1. PMU_STANDBY_EN = 1, PMU_MODE[1:0] = 11, PMU_BKSRAMOFF = 0 2. EFC_Sys_Mode = 1	1. Wakeup on external pins of PA0, PA2, PC0, PC2, PC3 and PC13

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Power Mode	Description	Entry Condition	Wakeup Source
	32K low-speed clock is inactive. BKSRAM / IWDT / LPTIM / LPUART are powered on.	3. Set SLEEPDEEP = 1 for M4 core. 4. WFI/WFE	2. Wakeup by reset (RESETN)
DeepStandby1 mode	The CORE domain is powered off, the BBU domain is kept on, and the 32 K low-speed clock is inactive. BKSRAM / IWDT / LPTIM / LPUART are powered off.	1. PMU_STANDBY_EN = 1, PMU_MODE[1:0] = 11, PMU_BKSRAMOFF = 1 2. EFC_Sys_Mode = 1 3. Set SLEEPDEEP = 1 for M4 core. 4. WFI/WFE	1. Wakeup by external pins of PA0, PA2, PC0, PC2, PC3 and PC13 2. Wakeup by reset (RESETN)
Power-off mode	All supplies are powered off.	Powering off external VDDH & VBAT	Power on

Notes:

- BBU domain includes RTC, backup register and PMU logic.
- Sleep mode and stop mode supports IO retention.
- Standby mode and DeepStandby mode do not support IO retention, except for IO woken up by external input, other IOs are in high-impedance state.

5.2.2 Run Mode

In run mode, according to different application scenarios, it is also possible to reduce power consumption by reducing the operating frequency or turning off some unused peripherals.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5.2.3 Sleep Mode

In Sleep mode, CPU stops working and the interrupt handling function is retained. The clock and reset of other peripheral modules can be set by software. This low-power mode is entered by the CPU by executing the specific instruction WFI, and the wake-up is triggered by interrupt.

5.2.4 Stop Mode

In Stop mode, the system RCH/XTH/PLL stop working (RCL/XTL remain active) and the interrupt handling function is retained. The clock and reset of other peripheral modules can be set by software. This low-power mode is entered by the CPU by executing the specific instruction WFI, and the wake-up is triggered by interrupt.

5.2.5 Standby Mode

In Standby mode, RTC keeps running, LPUART/LPTIM1 ~ 2 can be configured as active or inactive, and BKSRAM can be configured to be powered off or not.

5.2.6 DeepStandby Mode

In DeepStandby mode, the internal low-speed clock stops working, the backup register is retained, and BKSRAM can be powered off or retained.

5.3 Register Description

Note: If XTL or RCL is selected as the system clock, the register of this PMU is not accessible.

Register base address: 0x40B0_0100

The PMU registers are listed below:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 5-2: List of PMU Registers

Offset Address	Name	Description
0x00	PMU_MR	PMU mode register
0x04	PMU_PDWKCR	Power-down wakeup control register
0x08	PMU_PUSTCR	Power-up stabilization time configuration register
0x0C	PMU_VDCR	PDR/BOR/LVD configuration register
0x14	PMU_SASR	System auxiliary status register
0x1C	PMU_XTLCR	XTL configuration register
0x20	PMU_RCLCR	RCL configuration register
0x24	PMU_FCCR	Function clock control register
0x28	PMU_FRCR	Function reset control register
0x50	PMU_CPR	Configuration protection register

5.3.1 PMU Mode Register (PMU_MR)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	R	0	Reserved
8	VOLTAGE_SEL	R/W	0	MLDO output voltage selection: 0: 1.1-V output (adopting trimming value at 1.1 V) 1: 0.9-V output (adopting trimming value at 0.9 V)
7:5	RSV	R	0	Reserved
4	STDBY_EN	R/W	0	Standby mode enable: 0: Standby mode disabled 1: Standby mode enabled
3	STOP_CLK_SEL	R/W	0	Stop mode clock selection: 0: When entering stop mode, switch to RCH first, and still use RCH on wakeup. 1: Keep the working clock (e.g. PLL clock) unchanged before entering stop mode.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	BKSRAMOFF	R/W	0	BKSRAM and other logics power-down control: 0: BKSRAM, IWDT, LPUART and LPTIM1 ~ 2 not powered down 0: BKSRAM, IWDT, LPUART and LPTIM1 ~ 2 powered down
1:0	PMU_MODE	R/W	0	PMU mode register: 2'b00: Run mode 2'b01: Stop mode 2'b10: Standby mode/Power-down mode 2'b11: DeepStandby mode / Deep power-down mode

5.3.2 Power-down Wakeup Control Register (PMU_PDWKCR)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	IO_WK_CLR	W	-	Clear the wakeup status bits of PA0/PA2/PC0/PC2/PC3/PC13. Note: Before clearing the wakeup status bit, it is required to make the wakeup enable bit invalid first, and then write 1 to clear all wakeup status flag bits.
30	RESETN_WK_FLAG	R	0	External RESETN wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by the RESETN pin 1: Woken up by the RESETN pin (in the case of single-pin wakeup), or possibly woken up by the RESETN pin (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
29	PC13_WK_FLAG	R	0	PC13 wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by PC13 1: Woken up by PC13 (in the case of single-pin wakeup), or possibly woken up by PC13 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].
28	PC3_WK_FLAG	R	0	PC3 wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by PC3 1: Woken up by PC3 (in the case of single-pin wakeup), or possibly woken up by PC3 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].
27	PC2_WK_FLAG	R	0	PC2 wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by PC2 1: Woken up by PC2 (in the case of single-pin wakeup), or possibly woken up by PC2 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].
26	PC0_WK_FLAG	R	0	PC0 wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by PC0 1: Woken up by PC0 (in the case of single-pin wakeup), or possibly woken up by PC0 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
25	PA2_WK_FLAG	R	0	PA2 wakeup status bit (valid only in Standby & DeepStandby modes): 0: Not woken up by PA2 1: Woken up by PA2 (in the case of single-pin wakeup), or possibly woken up by PA2 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].
24	PA0_WK_FLAG	R	0	PA0 wakeup status bit (only valid in Standby & DeepStandby modes): 0: Not PA0 pin wakeup 1: Woken up by PA0 (in the case of single-pin wakeup), or possibly woken up by PA0 (in the case of multiple-pin wakeup) Note: This bit is cleared by writing 1 to bit [31].
23:22	RSV	R	-	Reserved
21	PC13_WKEG	R/W	0	PC13 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger
20	PC3_WKEG	R/W	0	PC3 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger
19	PC2_WKEG	R/W	0	PC2 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger
18	PC0_WKEG	R/W	0	PC0 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger
17	PA2_WKEG	R/W	0	PA2 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger
16	PA0_WKEG	R/W	0	PA0 wakeup edge selection: 0: Rising-edge trigger 1: Falling-edge trigger

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:13	RSV	R	0	Reserved
12	PC13_WKE	R/W	0	PC13 wakeup interrupt enable: 0: Wakeup event not verified 1: Wakeup event verified
11	PC3_WKE	R/W	0	PC3 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
10	PC2_WKE	R/W	0	PC2 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
9	PC0_WKE	R/W	0	PC0 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
8	PA2_WKE	R/W	0	PA2 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
7	LPTIM2_WKE	R/W	0	LPTIM2 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
6	LPTIM1_WKE	R/W	0	LPTIM1 interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
5	LPUART_WKE	RW	0	LPUART wakeup interrupt enable: 0: Wakeup event not verified 1: Wakeup event verified
4	IWDT_WKE	R/W	0	IWDT interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
3	RTC_TAMP_WKE	R/W	0	RTC TAMPER interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified Note: In DeepStandby mode, it is woken up upon PC13 interrupt event.
2	RTC_ALARM_WKE	R/W	0	RTC ALARM interrupt wakeup enable: 0: Wakeup event not verified 1: Wakeup event verified
1	PA0_WKE	R/W	0	PA0 reset wakeup enable:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Wakeup event not verified 1: Wakeup event verified
0	RSTN_WKE	R/W	0	External pin reset wakeup enable (falling-edge trigger): 0: Wakeup event not verified 1: Wakeup event verified

5.3.3 Power-up Stabilization Time Configuration Register (PMU_PUSTCR)

Offset address: 0x08

Reset value: 0x0000 007F

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	R	0	Reserved
6:0	CNT_VALUE	R/W	7'h7F	MLDO stabilization time setting: The default value is 128 clock cycles of 32 K (around 4 ms), and it can be set to 3 or less in the actual application.

5.3.4 PDR/BOR/LVD Configuration Register (PMU_VDCR)

Offset address: 0x0C

Reset value: 0x0000 A701

Bit	Name	Attribute	Reset Value	Description
31:18	RSV	R	0	Reserved
17	LVD_INT_EN	R/W	0	LVD interrupt enable register: 0: Disabled 1: Enabled
16	LVD_RST_EN	R/W	0	LVD reset enable register: 0: Disabled 1: Enabled
15:12	LVDS	R/W	4'hA	LVD settings

Unicmicro

Bit	Name	Attribute	Reset Value	Description
				Refer to Table 5-3
11:8	BORS	R/W	4'h7	BOR settings Refer to Table 5-4
7:6	PDRS	R/W	2'b00	PDR settings Refer to Table 5-5
5	RSV	R	0	Reserved
4	LVD_FILTER_EN	R/W	0	LVD filter enable bit: 0: Disabled 1: Enabled (two 32K clock cycles)
3	RSV	R/W	0	Reserved
2	LVD_EN	R/W	0	Low-voltage detection enable: 0: Disabled 1: Enabled
1	BOR_EN	R/W	0	Brown-out reset control: 0: BOR detection disabled, no reset generated 1: BOR detection enabled, reset generated Note: This bit must be set to 1 during use to enable the BOR detection reset.
0	PDR_EN	R/W	1	Power-down reset control: 0: PDR detection disabled, no reset generated 1: PDR detection enabled, reset generated Note: This bit shall be held at 1 in run mode, and setting it to 0 is only permissible in test mode.

Table 5-3: LVD (VDT) Settings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operating current	-	-	-	40	-	nA
LVD trigger point	V_{LVD}	LVDS[3:0] = 0000	-	1.59	-	V
		LVDS[3:0] = 0001	-	1.68	-	V
		LVDS[3:0] = 0010	-	1.78	-	V
		LVDS[3:0] = 0011	-	1.88	-	V

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
		LVDS[3:0] = 0100	-	1.98	-	V
		LVDS[3:0] = 0101	-	2.08	-	V
		LVDS[3:0] = 0110	-	2.18	-	V
		LVDS[3:0] = 0111	-	2.28	-	V
		LVDS[3:0] = 1000	-	2.38	-	V
		LVDS[3:0] = 1001	-	2.48	-	V
		LVDS[3:0] = 1010	-	2.58	-	V
		LVDS[3:0] = 1011	-	2.67	-	V
		LVDS[3:0] = 1100	-	2.77	-	V
		LVDS[3:0] = 1101	-	2.87	-	V
		LVDS[3:0] = 1110	-	2.97	-	V
		LVDS[3:0] = 1111	-	3.07	-	V

Table 5-4: BOR Settings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operating current	-	-	-	40	-	nA
BOR trigger point	V_{BOR}	BORS[3:0] = 0000	-	1.56	-	V
		BORS[3:0] = 0001	-	1.66	-	V
		BORS[3:0] = 0010	-	1.75	-	V
		BORS[3:0] = 0011	-	1.85	-	V
		BORS[3:0] = 0100	-	1.95	-	V
		BORS[3:0] = 0101	-	2.05	-	V
		BORS[3:0] = 0110	-	2.14	-	V
		BORS[3:0] = 0111	-	2.24	-	V
		BORS[3:0] = 1000	-	2.34	-	V
		BORS[3:0] = 1001	-	2.43	-	V
		BORS[3:0] = 1010	-	2.53	-	V
		BORS[3:0] = 1011	-	2.63	-	V
		BORS[3:0] = 1100	-	2.73	-	V
		BORS[3:0] = 1101	-	2.83	-	V
		BORS[3:0] = 1110	-	2.92	-	V
		BORS[3:0] = 1111	-	3.02	-	V

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 5-5: PDR Settings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operating current	-	-	-	40	-	nA
PDR rising trigger point	V_{PDR_R}	PDRS[1:0] = 00	-	1.57	-	V
		PDRS[1:0] = 01	-	1.77	-	V
		PDRS[1:0] = 10	-	1.87	-	V
		PDRS[1:0] = 11	-	1.97	-	V
PDR falling trigger point	V_{PDR_F}	PDRS[1:0] = 00	-	1.49	-	V
		PDRS[1:0] = 01	-	1.67	-	V
		PDRS[1:0] = 10	-	1.77	-	V
		PDRS[1:0] = 11	-	1.87	-	V
PDR trigger point hysteresis voltage	V_{PDR_HY}	PDRS[1:0] = 00	-	0.08	-	V
		PDRS[1:0] = 01	-	0.1	-	V
		PDRS[1:0] = 10	-	0.1	-	V
		PDRS[1:0] = 11	-	0.1	-	V

5.3.5 System Auxiliary Status Register (PMU_SASR)

Offset address: 0x14

Reset value: 0x0000 0234

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	R	0	Reserved
13	LVD_FLAG	R/W	0	LVD reset status flag bit: 0: No reset occurred 1: Reset occurred Note: This bit is cleared by writing 1 to the RST_FLAG_CLR bit, and is valid only when LVD reset occurs on PMU.
12	BOR_FLAG	R/W	0	BOR status flag: 0: No reset occurred 1: Reset occurred Note: This bit is cleared by writing 1 to the RST_FLAG_CLR bit, and is valid only when BOR reset occurs on PMU.
11	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	XTL_RSTN_FLAG	R	0	XTL exception status flag bit: 0: No exception occurred 1: Exception occurred Notes: 1) This bit is cleared by writing 1 to the RST_FLAG_CLR bit. 2) The XTL detection circuit function is recommended to be enabled only when XTL is used as the BBU module clock. After reset, the clock automatically switches to RCL. In addition, this status will be sent to the CPU core as an interrupt signal, sharing the same interrupt number with the RTC TAMPER interrupt.
9	CORE_PDN_FLAG	R/W	1	Core power-down reset status flag bit: 0: No reset occurred 1: Reset occurred Note: This bit is cleared by writing 1 to the RST_FLAG_CLR bit.
8	PDR_FLAG	R/W	0	PDR reset status flag bit: 0: No reset occurred 1: Reset occurred Note: This bit is cleared by writing 1 to the RST_FLAG_CLR bit.
7	RST_FLAG_CLR	W	0	Reset clear flag bit: 0: Not cleared 1: Cleared
6	XTL_DT_EN	R/W	0	XTL detection circuit enable: 0: Disabled 1: Enabled
5	LVD_PMU_EN	R/W	1	Whether the LVD reset signal can reset PMU state machine: 0: PMU state machine cannot be reset. 1: PMU state machine can be reset.
4	BOR_PMU_EN	R/W	1	Whether the BOR reset signal can reset PMU state machine:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: PMU state machine cannot be reset. 1: PMU state machine can be reset.
3	EFC_LOW_VDD_EN	R/W	0	Low voltage warning enable: 0: Low voltage warning disabled 1: Low voltage warning enabled
2:1	RSV	R/W	2'b10	Reserved
0	BAT_SEL	R/W	0	External battery backup status register: 0: No external battery backup power (only one main power supply for all system chips) 1: External battery backup power exists (two external power supplies for system chip) If the external V_{DDH} is abnormally powered off, the PDR_EN bit is recommended to be configured as 1. If the chip actively enters the low-power mode and then V_{DDH} is powered off, the PDR_EN bit is recommended to be configured as 0.

5.3.6 XTL Configuration Register (PMU_XTLCR)

Offset address: 0x1C

Reset value: 0x0000 9240

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	R	0	Reserved
15:13	ISET	R/W	3'b100	Oscillator current
12:10	RSET	R/W	3'b100	Bias resistance
9:7	GSET	R/W	3'b100	Gain of oscillator
6:4	FBRSET	R/W	3'b100	Feedback resistance
3:2	RSV	R	0	Reserved
1	XTL_EN	R/W	0	XTL enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	LSCLK_SEL	R/W	0	Low-speed clock selection: 0: RCL selected 1: XTL selected

5.3.7 RCL Configuration Register (PMU_RCLCR)

Offset address: 0x20

Reset value: 0xXXXX XXXX

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	0xXXXX XXXX	Reserved
0	RCL_PD	R/W	0	RCL enable: 0: RCL enabled 1: RCL disabled (not recommended) Note: RCL is also controlled by other logics. RCL is automatically disabled in DeepStandby mode while automatically enabled when there is an external wakeup.

5.3.8 Function Clock Control Register (PMU_FCCR)

Offset address: 0x24

Reset value: 0x0000 004F

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	R	0	Reserved
6	RTCEN	R/W	1	RTC module clock enable: 0: Disabled 1: Enabled
5:4	RSV	R	0	Reserved
3	LPTIM2EN	R/W	1	LPTIM2 module clock enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	LPTIM1EN	R/W	1	LPTIM1 controller clock enable: 0: Disabled 1: Enabled
1	IWDTEN	R/W	1	IWDT controller clock enable: 0: Disabled 1: Enabled
0	LPUARTEN	R/W	1	LPUART controller clock enable: 0: Disabled 1: Enabled

5.3.9 Function Reset Control Register (PMU_FRCR)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	R	0	Reserved
6	RTCRST	R/W	0	RTC control register reset enable: 0: Disabled 1: Enabled
5:4	RSV	R	0	Reserved
3	LPTIM2RST	R/W	0	LPTIM2 control register reset enable: 0: Disabled 1: Enabled
2	LPTIM1RST	R/W	0	LPTIM1 controller reset enable: 0: Disabled 1: Enabled
1	IWDTRST	R/W	0	IWDT control register reset enable: 0: Disabled 1: Enabled
0	LPUARTRST	R/W	0	LPUART control register reset enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5.3.10 Configuration Protection Register (PMU_CPR)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	R	0	Reserved
15:0	CONFIGEN	R/W	0	Configuration protection register: Write 16'hABCD to enable write operation to PMU register, only after which can write operation be performed on PMU-related configuration registers. Write 16'h459E to disable the write operation to PMU register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6 Reset and Clock Module (RCM)

6.1 Clock Logic

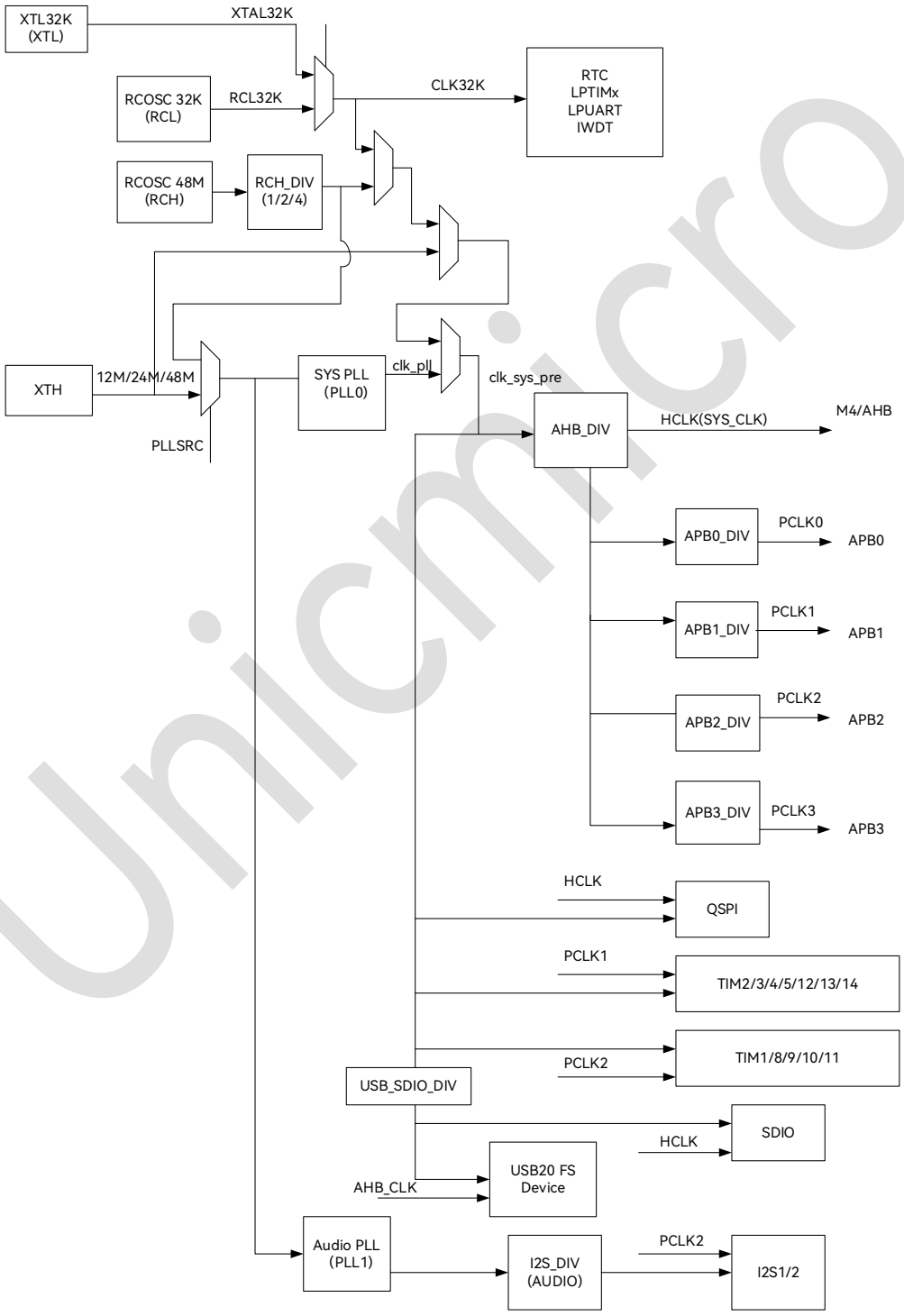

Figure 6-1: Clock Logic

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Four different sources can be used to drive the system clock:

- 48 MHz high-precision internal RCH
- 32 kHz internal RCL
- 32.768 kHz external crystal XTL
- External crystal XTH

There are two internal PLLs for system, audio and USB clocks.

The clock controller distributes the clocks coming from different oscillators to the core and the peripherals. It features a 48 MHz internal RC oscillator (RCH), an external high-speed crystal oscillator (XTH), a 32 kHz internal RC oscillator (RCL), a low-speed external crystal oscillator (XTL), two PLLs, an XTH monitor, clock prescaler, clock multiplexer and clock gating circuitry.

AHB, APB and Cortex™-M4 are derived from the system clock (SYS_CLK), and the clock source of SYS_CLK can be XTH, RCH, PLL, or RCL/XTL with a low frequency of 32 kHz. The independent watchdog is clocked from a low-frequency RCL or XTL, and the RTC is clocked from RCL or XTL.

Notes:

- If audio support is required, PLL fractional frequency division can be configured.
- For applications with strict EMC requirements (motor, etc.), PLL spread spectrum function can be enabled.

6.2 Reset Logic

There are two types of reset, defined as:

- Power reset
- System reset

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.2.1 Power Reset

Power reset, also known as cold reset, resets all systems except for the backup domain. System reset will reset all parts except for the backup domain, including the processor core and peripheral IPs. The backup domain reset will reset only the backup domain area. A reset can be triggered by external signals, internal events and reset generators.

A power reset will be generated when any of the following events occurs:

- Power-on/power-down reset (POR/PDR)
- Brown-out reset (BOR)
- Generated by the internal reset generator upon returning from standby mode

The power reset affects all registers except those in the backup domain. The power reset is active low and becomes inactive when the internal LDO power reference is ready to supply the core voltage (e.g. 1.1 V). The reset entry vector is fixed at the memory-mapped address 0xXXXX_0004.

6.2.2 System Reset

A system reset will be generated when any of the following events occurs:

- Power-on reset (POR)
- External pin reset (RESETN)
- M4 window watchdog counter termination (WWDT_RSTN)
- Independent watchdog counter termination (IWDT_RSTN)
- The SYSRESETREQ bit in the Cortex™-M4 interrupt application and reset control register is set to 1.

System reset will reset all parts except for the backup domain, including the processor core and peripheral IPs.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

There are seven reset sources, all of which are active low. Each reset signal allows to re-run the CPU. Most registers will be reset again and the program counter (PC) will be remapped at address 0x0000 0000.

Table 6-1: System Reset Source

Reset Signal Name	Active Level	Description
RESETN	Low	Global hardware reset pin: reset the whole chip (except BBU and other registers)
POR_RSTN	Low	1.1 V power-on reset: reset the whole chip
BOR_RSTN	Low	Brown-out reset: reset the whole chip
SOFT_RSTN	Low	Global soft reset control register: reset the system
IWDT_RSTN	Low	Watchdog timer reset: reset the system
WWDT_RSTN	Low	Window watchdog timer reset: reset the system
Block reset signals	Low	Reset a single module

Table 6-2: Reset Mode

Reset Mode	Generation Condition
POR & PDR	Power on with V_{DDH} (1.8–3.6 V) and internal core voltage
RESETN pin reset	External RESETN pin inputs low level
BOR	V_{DDH} drops below the V_{BOR} threshold
LVD reset	V_{DDH} drops below the V_{LVD} threshold
Window watchdog reset (WWDT)	-
Independent watchdog reset (IWDT)	-
Reset upon wakeup from power-down mode	The reset is generated by setting the power-down mode, the core wakes up from the reset state upon occurrence of the power-down wakeup event.
Software reset	-
Reset upon abnormal oscillation stop of external high-speed oscillator	The abnormal oscillation stop of external high-speed oscillator is detected.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3 Register Description

Register base address: 0x40B0_1000

Table 6-3: List of RCM Registers

Offset Address	Name	Description
0x000	RCM_CR0	Clock control register 0
0x008	RCM_PLL0CFGR0	PLL0 configuration register 0
0x00C	RCM_PLL0CFGR1	PLL0 configuration register 1
0x010	RCM_PLL0CFGR2	PLL0 configuration register 2
0x014	RCM_PLL1CFGR0	PLL1 configuration register 0
0x018	RCM_PLL1CFGR1	PLL1 configuration register 1
0x01C	RCM_PLL1CFGR2	PLL1 configuration register 2
0x020	RCM_PLLTSR	PLL stabilization time setting register
0x030	RCM_CFGR0	Clock configuration register 0
0x034	RCM_CFGR1	Clock configuration register 1
0x040	RCM_CIFR	Clock interrupt flag register
0x044	RCM_CIER	Clock interrupt enable register
0x060	RCM_AHBCKENR	AHB peripheral clock enable register
0x064	RCM_AHB0CKENR	AHB0 peripheral clock enable register
0x068	RCM_AHB1CKENR	AHB1 peripheral clock enable register
0x06C	RCM_APB0CKENR	APB0 peripheral clock enable register
0x070	RCM_APB1CKENR	APB1 peripheral clock enable register
0x074	RCM_APB2CKENR	APB2 peripheral clock enable register
0x078	RCM_APB3CKENR	APB3 peripheral clock enable register
0x07C	RCM_AHBRSTR	AHB peripheral reset register
0x080	RCM_AHB0RSTR	AHB0 peripheral reset register
0x084	RCM_AHB1RSTR	AHB1 peripheral reset register
0x088	RCM_APB0RSTR	APB0 peripheral reset register
0x08C	RCM_APB1RSTR	APB1 peripheral reset register
0x090	RCM_APB2RSTR	APB2 peripheral reset register
0x094	RCM_APB3RSTR	APB3 peripheral reset register
0x098	RCM_SOFTRSTR	Software reset register
0x0E0	RCM_RFR	Reset flag register
0x0E4	RCM_EXRSTFER	External reset pin filter enable register
0x0F0	RCM_RCMPR	RCM write protection register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.1 Clock Control Register 0 (RCM_CR0)

Address: 0x000

Reset value: 0x0020 0421

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	R	0	Reserved
28	PLLSRC	R/W	0	PLL0/PLL1 input clock selection: 0: RCH selected as PLL0/PLL1 input clock 1: XTH selected as PLL0/PLL1 input clock Note: It can only be configured when PLL0/PLL1 is not working.
27	PLL1STB	R	0	PLL1 (Audio) stability flag: 0: Unstable 1: Stable
26	PLL1EN	R/W	0	PLL1 (Audio) enable: 0: Disabled 1: Enabled
25	PLL0STB	R	0	PLL0 stabilization flag: 0: Unstable 1: Stable
24	PLL0EN	R/W	0	PLL0 enable: 0: Disabled 1: Enabled
23:22	XTH_SF	R/W	0	XTH oscillation frequency selection (SF0, SF1): 2'b00: 1–4 MHz 2'b01: 4.1–12 MHz 2'b10: 12.1–24 MHz 2'b11: 24.1–48 MHz
21:20	XTHSS	R/W	2’b10	XTH stabilization time selection: 2’b00: 4096 cycles 2’b01: 16384 cycles 2’b10: 32768 cycles 2’b11: 65535 cycles

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19	RSV	R	0	Reserved
18	XTH_BYP	R/W	0	XTH bypass selection: 1: External crystal not required, clock input directly from pin 0: Out from XTH
17	XTH_STB	R	0	XTH stabilization flag: 1: Stable 0: Unstable
16	XTH_EN	R/W	0	XTH enable: 0: Disabled 1: Enabled
15:14	XTH_SFRB	R/W	0	XTH feedback resistance selection: The resistance configuration signal is of the following correspondence: 00: 500 kΩ 01: 100 kΩ 1X: HiZ
13:11	RSV	R	-	Reserved
10	RCH_STB	R	1	Internal RCH stabilization flag: 1: RCH is stable, can be used for internal circuit. 0: RCH is not stable, can not be used for internal circuit.
9:8	RCHSS	R/W	2'b00	Internal RCH stabilization time selection: 00: 4 cycles 01: 16 cycles 10: 64 cycles 11: 256 cycles Note: This value can be reduced in Stop mode to speed up the wakeup time.
7:6	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5:4	XTLSS	R/W	2'h2	External XTL stabilization time selection: 2'b00: 1024 cycles 2'b01: 4096 cycles 2'b10: 16384 cycles 2'b11: 32768 cycles
3	XTH_RST_INT_EN	R/W	0	Enable reset or interrupt upon XTH exception: 0: Disabled 1: Enabled Note: This register will not be reset by the XTH exception. The register value shall be cleared by writing 0 to clear the oscillation stop interrupt.
2	XTH_STOP_SEL	R/W	0	XTH exception status selection: 0: Interrupt generated upon clock exception 1: Reset generated upon clock exception Note: This register will not be reset by the XTH exception. The register value shall be cleared by writing 0.
1	XTH_MEN	R/W	0	XTH monitoring enable: 0: Disabled 1: Enabled
0	RCH_EN	R/W	1	Internal RCH enable: 0: Disabled 1: Enabled Note: When the system enters Standby or DeepStandby mode, RCH will automatically turn off.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.2 PLL0 Configuration Register 0 (RCM_PLL0CFGR0)

Address: 0x008

Reset value: 0x0001 0388

Bit	Name	Attribute	Reset Value	Description
31:21	SSRATE	R/W	0	Spread spectrum slope setting
20:15	PLL0_DM	R/W	6'h2	PLL0_DM signal
14:5	PLL0_DN	R/W	10'h1C	PLL0_DM signal: the value written shall be greater than or equal to 16.
4:2	PLL0_DP	R/W	3'h2	PLL0_DP signal
1	PLL0_BYP	R/W	0	PLL0 bypass control signal: 0: Clock signal comes from PLL VCO 1: PLL VCO bypassed and clock signal comes from REF CLK
0	PLL0_CFGEN	R/W	0	PLL0 configuration enable (with parameter change): 0: Without parameter change 1: With configuration change Note: After the PLL parameter configuration is changed, enable this bit and wait for the PLL to stabilize before switching the PLL clock.

Notes:

1. For integer frequency division, the formula of PLL0 output clock is as follows:

   $f_{CLKO} = f_{refclk} * PLL0_DN / (PLL0_DM * PLL0_DP)$

2. For fractional frequency division, the formula of PLL0 output clock is as follows:

   $f_{CLKO} = f_{refclk} * (PLL0_DN + PLLI0_FRAC/2^{24}) / (PLL0_DM * PLL0_DP)$

3. For the above two formulas, the frequency of ($f_{refclk} * PLL0_DN/PLL0_DM$) shall be greater than 300 MHz and less than 600 MHz.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.3 PLL0 Configuration Register 1 (RCM_PLL0CFGR1)

Address: 0x00C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	R	0	Reserved
25:24	SSC_MODE	R/W	0	PLL0 spread spectrum mode selection: 2'b00: Down spread 2'b01: Center spread 2'b10: Up spread 2'b11: Reserved
23:0	SLOPE	R/W	0	PLL0 spread spectrum slope coefficient

6.3.4 PLL0 Configuration Register 2 (RCM_PLL0CFGR2)

Address: 0x010

Reset value: 0x0800 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	R	0	Reserved
29:26	COUNT	R/W	4'h2	PLL control signal: COUNT[0]: Dithering mode enable (1: Enabled) COUNT[1]: PLL fractional part logic reset control (1: Fractional part logic reset; 0: Fractional part logic reset release); in fractional mode and spread spectrum mode, this bit is written as 0. COUNT[3:2]: Reserved
25:24	MODE	R/W	0	Operation mode selection: 2'b00: Integer mode 2'b01: Fractional mode 2'b10: Spread spectrum mode 2'b11: Reserved
23:0	FRAC	R/W	0	PLL0 fractional division factor

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.5 PLL1 Configuration Register 0 (RCM_PLL1CFGR0)

Address: 0x014

Reset value: 0x0001 0390

Bit	Name	Attribute	Reset Value	Description
31:21	SSRATE	R/W	0	Spread spectrum slope setting
20:15	PLL1_DM	R/W	6'h2	PLL1_DM signal
14:5	PLL1_DN	R/W	10'h1C	PLL1_DM signal: The value written shall be greater than or equal to 16.
4:2	PLL1_DP	R/W	3'h4	PLL1_DP signal
1	PLL1_BYP	R/W	0	PLL1 bypass control signal: 0: Clock signal comes from PLL VCO. 1: PLL VCO bypassed and clock signal comes from REF CLK.
0	PLL1_CFGEN	R/W	0	PLL1 configuration enable (with parameter change): 0: Without parameter change 1: With configuration change Note: After the PLL parameter configuration is changed, enable this bit and wait for the PLL to stabilize before switching the PLL clock.

Note: For integer frequency division, the formula of PLL1 output clock is as follows:

$f_{CLKO} = f_{refclk} * PLL1_DN / (PLL1_DM * PLL1_DP)$

6.3.6 PLL1 Configuration Register 1 (RCM_PLL1CFGR1)

Address: 0x018

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
25:24	SSC_MODE	R/W	0	PLL1 spread spectrum mode selection: 2'b00: Down spread 2'b01: Center spread 2'b10: Upper spread spectrum 2'b11: Reserved
23:0	SLOPE	R/W	0	PLL1 spread spectrum slope coefficient

6.3.7 PLL1 Configuration Register 2 (RCM_PLL1CFGR2)

Address: 0x01C

Reset value: 0x0800 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	R	0	Reserved
29:26	COUNT	R/W	4'h2	PLL control signal: COUNT[0]: Dithering mode enable (1: Enabled) COUNT[1]: PLL fractional part logic reset control (1: Fractional part logic reset; 0: Fractional part logic reset release); in fractional mode and spread spectrum mode, this bit is written as 0. COUNT[3:2]: Reserved
25:24	MODE	R/W	0	Operation mode selection: 2'b00: Integer mode 2'b01: Fractional mode 2'b10: Spread spectrum mode 2'b11: Reserved
23:0	FRAC	R/W	0	PLL1 fractional division factor

6.3.8 PLL Stabilization Time Setting Register (RCM_PLLTSR)

Address: 0x020

Reset value: 0x0000 2EE0

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	R	0	Reserved
16	PLL_LT_EN	R/W	0	PLL output clock: 0: The software waits for PLL0STB or PLL1STB before switching the system clock to PLL (software waiting). 1: PLL stabilizes before outputting clock to be applied to the system. Note: This bit shall be set to 1 if PLL is selected as the system clock in Stop mode and it is expected to be kept after wakeup.
15:0	PLL_LT	R/W	16'h2EE0	PLL lock time setting register: 16'h2EE0 (16'd12000) corresponds to PLL lock time of 0.5 ms (counting with 24 MHz clock)

6.3.9 Clock Configuration Register 0 (RCM_CFGR0)

Address: 0x030

Reset value: 0x0900 8040

Bit	Name	Attribute	Reset Value	Description
31:28	I2S_DIV	R/W	0	I2S_MCLK clock division factor (based on audio PLL): 4'b0xxx: Not divided 4'b1000: Divided by 2 4'b1001: Divided by 4 4'b1010: Divided by 8 4'b1011: Divided by 16 4'b1100: Divided by 32 4'b1101: Divided by 64 4'b1110: Divided by 128 4'b1111: Divided by 256
27:24	APB3_DIV	R/W	4'b1001	APB3 clock division factor (based on AHB clock): 4'b0xxx: Not divided

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4’b1000: Divided by 2 4’b1001: Divided by 4 4’b1010: Divided by 8 4’b1011: Divided by 16 4’b1100: Divided by 32 4’b1101: Divided by 64 4’b1110: Divided by 128 4’b1111: Divided by 256
23:20	APB2_DIV	R/W	0	APB2 clock division factor (based on AHB clock): 4’b0xxx: Not divided 4’b1000: Divided by 2 4’b1001: Divided by 4 4’b1010: Divided by 8 4’b1011: Divided by 16 4’b1100: Divided by 32 4’b1101: Divided by 64 4’b1110: Divided by 128 4’b1111: Divided by 256
19:16	APB1_DIV	R/W	0	APB1 clock division factor (based on AHB clock): 4’b0xxx: Not divided 4’b1000: Divided by 2 4’b1001: Divided by 4 4’b1010: Divided by 8 4’b1011: Divided by 16 4’b1100: Divided by 32 4’b1101: Divided by 64 4’b1110: Divided by 128 4’b1111: Divided by 256
15:12	APB0_DIV	R/W	4’b1000	APB0 clock division factor (based on AHB clock): 4’b0xxx: Not divided 4’b1000: Divided by 2 4’b1001: Divided by 4 4’b1010: Divided by 8 4’b1011: Divided by 16

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4’b1100: Divided by 32 4’b1101: Divided by 64 4’b1110: Divided by 128 4’b1111: Divided by 256
11:8	AHB_DIV	R/W	0	AHB clock division factor (based on clk_sys_pre clock): 4’b0xxx: Not divided 4’b1000: Divided by 2 4’b1001: Divided by 4 4’b1010: Divided by 8 4’b1011: Divided by 16 4’b1100: Divided by 32 4’b1101: Divided by 64 4’b1110: Divided by 128 4’b1111: Divided by 256
7:6	RCH_DIV	R/W	2’b01	Internal RCH division factor: At power-on prefetch, RCH (48 MHz) is divided by a fixed factor of 4 to generate 12 MHz clocks. 2’b00: Divided by 1 2’b01: Divided by 2 2’b10: Divided by 3 2’b11: Divided by 4 Note: Normal operation is performed with 24 MHz clock.
5	I2S_SRC	R/W	0	I2S clock source selection: 0: PLL1 (audio PLL) clock output as I2S clock 1: External pin I2S_MCLK input clock as I2S clock
4	RSV	R	0	Reserved
3:2	SYS_SWS	R	2’b00	System clock switching status: 2’b00: RCH selected as system clock 2’b01: XTH selected as system clock 2’b10: PLL0 output clock selected as system clock

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				2’b11: 32 kHz RCL or XTL selected as system clock Note: There exists the case that XTH is not available and it is switched to RCH.
1:0	SYS_SW	R/W	2’b00	System clock switching: 2’b00: RCH selected as system clock 2’b01: XTH selected as system clock 2’b10: PLL0 output clock selected as system clock 2’b11: 32 kHz RCL or XTL selected as system clock

6.3.10 Clock Configuration Register 1 (RCM_CFGR1)

Address: 0x034

Reset value: 0xFF06 2002

Bit	Name	Attribute	Reset Value	Description
31:28	USART8_DIV	R/W	4’hF	USART8 clock division factor (based on PCLK2 clock): 4’b0000: Reserved 4’b0001: Divided by 2 …… 4’b1111: Divided by 16
27:24	USART7_DIV	R/W	4’hF	USART7 clock division factor (based on PCLK0 clock): 4’b0000: Reserved 4’b0001: Divided by 2 …… 4’b1111: Divided by 16
23	USART8_CLK_SEL	R/W	0	USART8 low-speed clock source selection: 0: RCL/XTL selected 1: PCLK2 selected
22	USART7_CLK_SEL	R/W	0	USART7 low-speed clock source selection: 0: RCL/XTL selected

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: PCLK0 selected
21	TIM_CLK_SEL2	R/W	0	TIM1/8/9/10/11 clock source selection: 0: APB2_CLK selected 1: SYSPLL selected Note: If it is set to 1, APB2_CLK = HCLK = SYSPLL / 2. For example, if the SYSPLL clock is 336 MHz, then APB2_CLK = HCLK = 168 MHz, and the input clock of TIM1/8/9/10/11 is 336 MHz.
20	TIM_CLK_SEL1	R/W	0	TIM2/3/4/5/12/13/14 clock source selection (TIM5/6 clock source is always APB1_CLK): 0: APB1_CLK selected 1: SYSPLL selected Note: If it is set to 1, APB1_CLK = HCLK = SYSPLL / 2. For example, if the SYSPLL clock is 336 MHz, then APB1_CLK = HCLK = 168 MHz, and the input clock of TIM2/3/4/5/12/13/14 (except TIM6/7) is 336 MHz.
19	RSV	R	0	Reserved
18:16	USB_SDIO_DIV (48M_DIV)	R/W	3’b110	Division factor setting for USB and SDIO module operating clocks (if USB is used, appropriate division factor must be configured to generate 48 MHz clock): 3’b000: Not divided 3’b001: Divided by 2 3’b010: Divided by 3 3’b011: Divided by 4 3’b100: Divided by 5 3’b101: Divided by 6 3’b110: Divided by 7

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				3’b111: Divided by 8 Note: Used for USB and SDIO modules.
15	RSV	R	0	Reserved
14:12	MCO2_DIV	R/W	3’b010	MCO2 division factor: 3’b000: Not divided 3’b001: Divided by 2 3’b010: Divided by 4 3’b011: Divided by 8 3’b100: Divided by 16 3’b101: Divided by 32 3’b110: Divided by 64 3’b111: Divided by 128
11	RSV	R	0	Reserved
10:8	MCO2	R/W	3’b000	MCO2 clock output: 3’b000: RCH_DIV (e.g. 24 MHz) selected to be output onto MCO2 pin (PC9) 3’b001: XTH selected to be output onto MCO2 pin (PC9) 3’b001: low-speed clock (RCL or XTL) selected to be output onto MCO2 pin (PC9) 3’b011: XTL selected to be output onto MCO2 pin (PC9) 3’b001: PLL0 clock selected to be output onto MCO2 pin (PC9) 3’b101: PLL1 clock selected to be output onto MCO2 pin (PC9) 3’b110: Reserved 3’b111: AHB clock selected to be output onto MCO2 pin (PC9)
7	RSV	R	0	Reserved
6:4	MCO1_DIV	R/W	3’b000	MCO1 division factor: 3’b000: Not divided 3’b001: Divided by 2 3’b010: Divided by 4

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				3’b011: Divided by 8 3’b100: Divided by 16 3’b101: Divided by 32 3’b110: Divided by 64 3’b111: Divided by 128
3	RSV	R	0	Reserved
2:0	MCO1	R/W	3’b010	MCO1 clock output: 3’b000: RCH_DIV (e.g. 24 MHz) selected to be output onto MCO1 pin (PA8) 3’b001: XTH selected to be output onto MCO1 pin (PA8) 3’b001: Low-speed clock (RCL or XTL) selected to be output onto MCO1 pin (PA8) 3’b011: XTL clock selected to be output onto MCO1 pin (PA8) 3’b001: PLL0 clock selected to be output onto MCO1 pin (PA8) 3’b101: PLL1 clock selected to be output onto MCO1 pin (PA8) 3’b110: Reserved 3’b111: AHB clock selected to be output onto MCO1 pin (PA8)

6.3.11 Clock Interrupt Flag Register (RCM_CIFR)

Address: 0x040

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	R	0	Reserved
7	CSSF	R/W	0	Clock security system interrupt flag. Writing 1 clears this interrupt. 0: XTH is normal 1: XTH output failure
6	RSV	R	0	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	PLL1RDYF	R/W	0	PLL1 clock lock interrupt flag. Writing 1 clears this interrupt. 0: PLL1 output clock not locked 1: PLL1 output clock locked
4	PLL0RDYF	R/W	0	PLL0 clock lock interrupt flag. Writing 1 clears this interrupt. 0: PLL0 output clock not locked 1: PLL0 output clock locked
3	HSERDYF	R/W	0	XTH ready interrupt flag. Writing 1 clears this interrupt. 0: XTH clock not ready 1: XTH clock ready
2	RSV	R	0	Reserved
1	LSERDYF	R/W	0	XTL ready interrupt flag. Writing 1 clears this interrupt. 0: XTL clock not ready 1: XTL clock ready
0	RSV	R	0	Reserved

6.3.12 Clock Interrupt Enable Register (RCM_CIER)

Address: 0x044

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	R	0	Reserved
7	CSSIE	R/W	0	Clock security system interrupt enable: 0: Disabled 1: Enabled
6	RSV	R	0	Reserved
5	PLL1RDYIE	R/W	0	PLL1 ready interrupt enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	PLL0RDYIE	R/W	0	PLL0 ready interrupt enable: 0: Disabled 1: Enabled
3	HSERDYIE	R/W	0	XTH ready interrupt enable: 0: Disabled 1: Enabled
2	RSV	R	0	Reserved
1	LSERDYIE	R/W	0	XTL ready interrupt enable: 0: Disabled 1: Enabled
0	RSV	R	0	Reserved

6.3.13 AHB Peripheral Clock Enable Register (RCM_AHBCKENR)

Address: 0x060

Reset value: 0x0001 0700

Bit	Name	Attribute	Reset Value	Description
31:18	RSV	R	0	Reserved
17	EMCEN	R/W	0	EMC controller clock enable: 0: Disabled 1: Enabled
16	QSPIEN	R/W	1	QSPI controller clock enable: 0: Disabled 1: Enabled
15:11	RSV	R	0	Reserved
10	SRAM3EN	R/W	1	SRAM3 controller clock enable: 0: Disabled 1: Enabled
9	SRAM2EN	R/W	1	SRAM2 controller clock enable: 0: Disabled 1: Enabled
8	SRAM1EN	R/W	1	SRAM1 controller clock enable: 0: Disabled 1: Enabled
7:0	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.14 AHB0 Peripheral Clock Enable Register (RCM_AHB0CKENR)

Address: 0x064

Reset value: 0x0000 0011

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	R	0	Reserved
12	DCMIEN	R/W	0	DCMI controller clock enable: 0: Disabled 1: Enabled
11	RSV	R	0	Reserved
10	EMACEN	R/W	0	EMAC controller clock enable: 0: Disabled 1: Enabled
9	RSV	R	0	Reserved
8	SDIOEN	R/W	0	SDIO controller clock enable: 0: Disabled 1: Enabled
7	RSV	R	0	Reserved
6	DMA2EN	R/W	0	DMA2 controller clock enable: 0: Disabled 1: Enabled
5	DMA1EN	R/W	0	DMA1 controller clock enable: 0: Disabled 1: Enabled
4	CRCEN	R/W	1	CRC controller clock enable: 0: Disabled 1: Enabled
3:1	RSV	R	0	Reserved
0	USBEN	R/W	1	USB (device) module clock enable: 0: Disabled 1: Enabled

6.3.15 AHB1 Peripheral Clock Enable Register (RCM_AHB1CKENR)

Address: 0x068

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 01FF

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	R	0	Reserved
15	CORDICEN	R/W	0	CORDIC controller clock enable: 0: Disabled 1: Enabled
14	RSV	R	0	Reserved
13	SHAEN	R/W	0	SHA controller clock enable: 0: Disabled 1: Enabled
12	AESEN	R/W	0	AES controller clock enable: 0: Disabled 1: Enabled
11:8	RSV	R	0	Reserved
7	GPIOHEN	R/W	1	GPIOH controller clock enable: 0: Disabled 1: Enabled
6:5	RSV	R	0	Reserved
4	GPIOEEN	R/W	1	GPIOE controller clock enable: 0: Disabled 1: Enabled
3	GPIODEN	R/W	1	GPIOD module clock enable: 0: Disabled 1: Enabled
2	GPIOCEN	R/W	1	GPIOC controller clock enable: 0: Disabled 1: Enabled
1	GPIOBEN	R/W	1	GPIOB controller clock enable: 0: Disabled 1: Enabled
0	GPIOAEN	R/W	1	GPIOA controller clock enable: 0: Disabled 1: Enabled

6.3.16 APB0 Peripheral Clock Enable Register (RCM_APB0CKENR)

Address: 0x06C

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0400

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	R	0	Reserved
12	USART7EN	R/W	0	USART7 controller clock enable: 0: Disabled 1: Enabled
11	UART2EN	R/W	0	UART2 controller clock enable: 0: Disabled 1: Enabled
10	WWDTEN	R/W	1	WWDT controller clock enable: 0: Disabled 1: Enabled
9:0	RSV	R	0	Reserved

6.3.17 APB1 Peripheral Clock Enable Register (RCM_APB1CKENR)

Address: 0x070

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	R	0	Reserved
20	SPI1EN	R/W	0	SPI1 controller clock enable: 0: Disabled 1: Enabled
19	RSV	R	0	Reserved
18	I2C3EN	R/W	0	I2C3 controller clock enable: 0: Disabled 1: Enabled
17	I2C2EN	R/W	0	I2C2 controller clock enable: 0: Disabled 1: Enabled
16	I2C1EN	R/W	0	I2C1 controller clock enable: 0: Disabled 1: Enabled
15	RSV	R/W	0	Reserved
14	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13	DACEN	R/W	0	DAC1/2 module clock enable: 0: Disabled 1: Enabled
12	TIM14EN	R/W	0	TIM14EN controller clock enable: 0: Disabled 1: Enabled
11	TIM13EN	R/W	0	TIM13 controller clock enable: 0: Disabled 1: Enabled
10	TIM12EN	R/W	0	TIM12EN controller clock enable: 0: Disabled 1: Enabled
9	TIM7EN	R/W	0	TIM7 controller clock enable: 0: Disabled 1: Enabled
8	TIM6EN	R/W	0	TIM6 controller clock enable: 0: Disabled 1: Enabled
7	TIM5EN	R/W	0	TIM5 controller clock enable: 0: Disabled 1: Enabled
6	TIM4EN	R/W	0	TIM4 module clock enable: 0: Disabled 1: Enabled
5	TIM3EN	R/W	0	TIM3 controller clock enable: 0: Disabled 1: Enabled
4	TIM2EN	R/W	0	TIM2 controller clock enable: 0: Disabled 1: Enabled
3	UART6EN	R/W	0	UART6 controller clock enable: 0: Disabled 1: Enabled
2	UART5EN	R/W	0	UART5 controller clock enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	UART4EN	R/W	0	UART4 controller clock enable: 0: Disabled 1: Enabled
0	UART3EN	R/W	0	UART3 controller clock enable: 0: Disabled 1: Enabled

6.3.18 APB2 Peripheral Clock Enable Register (RCM_APB2CKENR)

Address: 0x074

Reset value: 0x0000 8000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	R	0	Reserved
15	UART1EN	R/W	1	UART1 controller clock enable: 0: Disabled 1: Enabled
14	USART8EN	R/W	0	USART8 controller clock enable: 0: Disabled 1: Enabled
13	TIM11EN	R/W	0	TIM11 controller clock enable: 0: Disabled 1: Enabled
12	TIM10EN	R/W	0	TIM10 controller clock enable: 0: Disabled 1: Enabled
11	TIM9EN	R/W	0	TIM9 controller clock enable: 0: Disabled 1: Enabled
10	TIM8EN	R/W	0	TIM8 controller clock enable: 0: Disabled 1: Enabled
9	TIM1EN	R/W	0	TIM1 controller clock enable: 0: Disabled 1: Enabled
8	SPI4EN	R/W	0	SPI4 controller clock enable:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Disabled 1: Enabled
7	SPI3EN	R/W	0	SPI3 controller clock enable: 0: Disabled 1: Enabled
6	SPI2EN	R/W	0	SPI2 controller clock enable: 0: Disabled 1: Enabled
5	ADC2EN	R/W	0	ADC2 controller clock enable: 0: Disabled 1: Enabled
4	ADC1EN	R/W	0	ADC1 controller clock enable: 0: Disabled 1: Enabled
3:2	RSV	R	0	Reserved
1	I2S2EN	R/W	0	I2S2 controller clock enable: 0: Disabled 1: Enabled
0	I2S1EN	R/W	0	I2S1 controller clock enable: 0: Disabled 1: Enabled

6.3.19 APB3 Peripheral Clock Enable Register (RCM_APB3CKENR)

Address: 0x078

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	R	0	Reserved
1	CAN2EN	R/W	0	CAN2 controller clock enable: 0: Disabled 1: Enabled
0	CAN1EN	R/W	0	CAN1 controller clock enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.20 AHB Peripheral Reset Enable Register (RCM_AHBRSTR)

Address: 0x07C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	R	0	Reserved
20	LOCKUP_EN	R/W	0	M4 LOCKUP reset enable: 0: Disabled 1: Enabled (a lockup reset will reset the system) Note: This bit is only affected by POR and PDR, other resets do not reset the value of this bit.
19:18	RSV	R	0	Reserved
17	EMCRST	R/W	0	EMC controller reset enable: 0: Disabled 1: Enabled
16	QSPIRST	R/W	0	QSPI controller reset enable: 0: Disabled 1: Enabled
15:11	RSV	R	0	Reserved
10	SRAM3RST	R/W	0	SRAM3 controller reset enable: 0: Disabled 1: Enabled
9	SRAM2RST	R/W	0	SRAM2 controller reset enable: 0: Disabled 1: Enabled
8	SRAM1RST	R/W	0	SRAM1 controller reset enable: 0: Disabled 1: Enabled
7:0	RSV	R	0	Reserved

6.3.21 AHB0 Peripheral Reset Enable Register (RCM_AHB0RSTR)

Address: 0x080

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	R	0	Reserved
12	DCMIRST	R/W	0	DCMI controller reset enable: 0: Disabled 1: Enabled
11	RSV	R	0	Reserved
10	EMACRST	R/W	0	EMAC controller reset enable: 0: Disabled 1: Enabled
9	RSV	R	0	Reserved
8	SDIORST	R/W	0	SDIO controller reset enable: 0: Disabled 1: Enabled
7	RSV	R/W	0	Reserved
6	DMA2RST	R/W	0	DMA2 controller reset enable: 0: Disabled 1: Enabled
5	DMA1RST	R/W	0	DMA1 controller reset enable: 0: Disabled 1: Enabled
4	CRCRST	R/W	0	CRC controller reset enable: 0: Disabled 1: Enabled
3:1	RSV	R	0	Reserved
0	USBRST	R/W	0	USB controller reset enable: 0: Disabled 1: Enabled

6.3.22 AHB1 Peripheral Reset Enable Register (RCM_AHB1RSTR)

Address: 0x084

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15	CORDICRST	R/W	0	CORDIC controller reset enable: 0: Disabled 1: Enabled
14	RSV	R	0	Reserved
13	SHARST	R/W	0	SHA controller reset enable: 0: Disabled 1: Enabled
12	AESRST	R/W	0	AES controller reset enable: 0: Disabled 1: Enabled
11:8	RSV	R	0	Reserved
7	GPIOHRST	R/W	0	GPIOH controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.
6:5	RSV	R	0	Reserved
4	GPIOERST	R/W	0	GPIOE controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.
3	GPIODRST	R/W	0	GPIOD controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.
2	GPIOCRST	R/W	0	GPIO controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.
1	GPIOBRST	R/W	0	GPIOB controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	GPIOARST	R/W	0	GPIOA controller reset enable: 0: Disabled 1: Enabled Note: The default value is 0 in the application.

6.3.23 APB0 Peripheral Reset Enable Register (RCM_APB0RSTR)

Address: 0x088

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	R	0	Reserved
12	USART7RST	R/W	0	USART7 controller reset enable: 0: Disabled 1: Enabled
11	UART2RST	R/W	0	UART2 controller reset enable: 0: Disabled 1: Enabled
10	WWDTRST	R/W	0	WWDT controller reset enable: 0: Disabled 1: Enabled
9:0	RSV	R	0	Reserved

6.3.24 APB1 Peripheral Reset Enable Register (RCM_APB1RSTR)

Address: 0x08C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	R	0	Reserved
20	SPI1RST	R/W	0	SPI1 controller reset enable: 0: Disabled 1: Enabled
19	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
18	I2C3RST	R/W	0	I2C3 controller reset enable: 0: Disabled 1: Enabled
17	I2C2RST	R/W	0	I2C2 controller reset enable: 0: Disabled 1: Enabled
16	I2C1RST	R/W	0	I2C1 controller reset enable: 0: Disabled 1: Enabled
15	RSV	R/W	0	Reserved
14	RSV	R	0	Reserved
13	DACRST	R/W	0	DAC controller reset enable: 0: Disabled 1: Enabled
12	TIM14RST	R/W	0	TIM14EN controller reset enable: 0: Disabled 1: Enabled
11	TIM13RST	R/W	0	TIM13 controller reset enable: 0: Disabled 1: Enabled
10	TIM12RST	R/W	0	TIM12EN controller reset enable: 0: Disabled 1: Enabled
9	TIM7RST	R/W	0	TIM7 controller reset enable: 0: Disabled 1: Enabled
8	TIM6RST	R/W	0	TIM6 controller reset enable: 0: Disabled 1: Enabled
7	TIM5RST	R/W	0	TIM5 controller reset enable: 0: Disabled 1: Enabled
6	TIM4RST	R/W	0	TIM4 controller reset enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	TIM3RST	R/W	0	TIM3 module reset enable: 0: Disabled 1: Enabled
4	TIM2RST	R/W	0	TIM2 controller reset enable: 0: Disabled 1: Enabled
3	UART6RST	R/W	0	UART6 controller reset enable: 0: Disabled 1: Enabled
2	UART5RST	R/W	0	UART5 controller reset enable: 0: Disabled 1: Enabled
1	UART4RST	R/W	0	UART4 controller reset enable: 0: Disabled 1: Enabled
0	UART3RST	R/W	0	UART3 controller reset enable: 0: Disabled 1: Enabled

6.3.25 APB2 Peripheral Reset Enable Register (RCM_APB2RSTR)

Address: 0x090

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	R	0	Reserved
15	UART1RST	R/W	0	UART1 controller reset enable: 0: Disabled 1: Enabled
14	USART8RST	R/W	0	USART8 controller reset enable: 0: Disabled 1: Enabled
13	TIM11RST	R/W	0	TIM11 controller reset enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	TIM10RST	R/W	0	TIM10 controller reset enable: 0: Disabled 1: Enabled
11	TIM9RST	R/W	0	TIM9 controller reset enable: 0: Disabled 1: Enabled
10	TIM8RST	R/W	0	TIM8 controller reset enable: 0: Disabled 1: Enabled
9	TIM1RST	R/W	0	TIM1controller reset enable: 0: Disabled 1: Enabled
8	SPI4RST	R/W	0	SPI4 controller reset enable: 0: Disabled 1: Enabled
7	SPI3RST	R/W	0	SPI3 controller reset enable: 0: Disabled 1: Enabled
6	SPI2RST	R/W	0	SPI2 controller reset enable: 0: Disabled 1: Enabled
5	ADC2RST	R/W	0	ADC2 controller reset enable: 0: Disabled 1: Enabled
4	ADC1RST	R/W	0	ADC1 controller reset enable: 0: Disabled 1: Enabled
3:2	RSV	R	0	Reserved
1	I2S2RST	R/W	0	I2S2 controller reset enable: 0: Disabled 1: Enabled
0	I2S1RST	R/W	0	I2S1 controller reset enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.26 APB3 Peripheral Reset Enable Register (RCM_APB3RSTR)

Address: 0x094

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	R	0	Reserved
1	CAN2RST	R/W	0	CAN2 controller reset enable: 0: Disabled 1: Enabled
0	CAN1RST	R/W	0	CAN1 controller reset enable: 0: Disabled 1: Enabled

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.27 Software Reset Register (RCM_SOFTRSTR)

Address: 0x098

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:0	SOFTRST	R/W	1	Software reset register: When this bit is written with 32'hA5A5_4321, a software reset will be generated to reset the CPU and all IPs on the AHB/APB bus. Also, the eFlash address will be remapped (to 1). When reading: 0: The system is about to perform a software reset. 1: The system does not perform a software reset. When writing: 32'hA5A5_4321: A software reset is generated. Other values: No software reset is generated. Note: Generally not used.

6.3.28 Reset Flag Register (RCM_RFR)

Address: 0x0E0

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	R	0	Reserved
16	RSTM	W	0	Reset clear flag bit: 0: Not cleared 1: Cleared
15:11	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	LOCKUPRSTF	R	0	LOCKUP reset status flag: 0: No reset occurred 1: Reset occurred
9	LVDRSTF	R	0	LVD reset status flag bit: 0: No reset occurred 1: Reset occurred
8	SYSSOFTRSTF	R	0	System software reset status flag: 0: No reset occurred 1: Reset occurred
7	BORRSTF	R	0	BOR status flag: 0: No reset occurred 1: Reset occurred
6	XTHRSTF	R	0	XTH clock reset status flag: 0: No reset occurred 1: Reset occurred
5	WWDTRSTF	R	0	Window watchdog reset status flag: 0: No reset occurred 1: Reset occurred
4	IWDTRSTF	R	0	Independent watchdog reset status flag: 0: No reset occurred 1: Reset occurred
3	SOFTRSTF	R	0	M4 software reset status flag: 0: No reset occurred 1: Reset occurred
2	PINRSTF	R	0	External pin RESETN reset status flag: 0: No reset occurred 1: Reset occurred
1	PORRSTF	R	1	POR01 reset status flag: 0: No reset occurred 1: Reset occurred
0	RSV	R	1	Reserved

Note: This register can only be reset by POR.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6.3.29 External Reset Filter Control Register (RCM_EXRSTFER)

Address: 0x0E4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	R	0	Reserved
0	EXT_FILTER_EN	R/W	0	External reset filter enable: 1: Enabled 0: Disabled

6.3.30 RCM Configuration Protection Register (RCM_RCMPR)

Address: 0x0F0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RCMPR	R/W	0	RCM register write protection: Write 0xA5A5_5A5A to this register to enable write operation of RCM register, and write other values to disable write operation. After configuring the RCM clock reset configuration register, it is recommended to write other values to disable the write operation and protect each configured RCM register value. When reading: 0: Write operation disabled 1: Write operation enabled When writing: 32'hA5A5_5A5A: Write operation enabled Other values: Write operation disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7 GPIO

7.1 Overview

GPIO contains general data input and output interfaces, which can be shared with other functional pins, depending on the chip configuration. With these data interfaces, any number of pins can be configured as interrupt signal inputs.

7.2 Main Features

- The direction of any I/O port can be configured by software.
- Each input pin can be configured with edge or level trigger for the interrupt.
- Input filter

7.3 Functional Description

7.3.1 GPIO Input and Output

Generate output signal and synchronize input signal.

7.3.2 GPIO Interrupt Generation

Capture the input signal and generate an interrupt. In the case of filtering being disabled, the level detection interrupt can be triggered without clock.

7.4 Register Description

GPIOA register base address: 0x4500_0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

GPIOB register base address: 0x4500_0400

GPIOC register base address: 0x4500_0800

GPIOD register base address: 0x4500_0C00

GPIOE register base address: 0x4500_1000

GPIOH register base address: 0x4500_1C00

The registers are listed below:

Table 7-1: List of GPIO Registers

Offset Address	Name	Description
0x00	GPIOx_MODE	Function mode register
0x04	GPIO_SET	Output set register
0x08	GPIO_CLR	Output clear register
0x0C	GPIO_ODATA	Output data register
0x10	GPIO_IDATA	Input data register
0x14	GPIO_IEN	Interrupt enable register
0x18	GPIO_IS	Interrupt trigger mode register
0x1C	GPIO_IBE	Interrupt edge-trigger mode register
0x20	GPIO_IEV	Interrupt level-trigger setting register
0x24	GPIO_IC	Interrupt clear register
0x28	GPIO_RIS	Raw interrupt status register
0x2C	GPIO_MIS	Masked interrupt status register
0x30	GPIO_DBEN	Filter enable register
0x34	GPIO_DBL	Filter length register
0x38	GPIO_LOCK	Configuration lock register
0x3C	GPIO_IM	Input mode register
0x40	GPIO_PULL	Pull-up/down register
0x48	GPIO_SR	Driving speed register
0x4C	GPIO_DS	Driving capability register
0x50	GPIO_AFL	Alternate function low register
0x54	GPIO_AFH	Alternate function high register

Registers are detailed in the following sections.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7.4.1 Function Mode Register (GPIOx_MODE)

Offset address: 0x00

Reset value for GPIOA: 0xABFF FFFF

Reset value for GPIOB: 0xFFFF FE8F

Reset value for GPIOC/GPIOD/GPIOE/GPIOH: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	MODE	R/W	0xFFFF FFFF (except GPIOA and GPIOB)	32-bit register with every 2 bits corresponding to one IO PAD; configure the GPIO mode: 00: General-purpose input mode 01: General-purpose output mode 10: Alternate function mode 11: Analog mode

7.4.2 Output Set Register (GPIO_SET)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SET	W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Invalid operation 1: I/O is set by writing this bit to 1 when I/O is configured as output.

7.4.3 Output Clear Register (GPIO_CLR)

Offset address: 0x08

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	CLR	W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Invalid operation 1: I/O is set by writing this bit to 1 when I/O is configured as output.

7.4.4 Output Data Register (GPIO_ODATA)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	ODATA	R/W	0x0	16-bit register with each bit corresponding to one I/O PAD: When the GPIO is output active, the write operation is directed at the external pin, and the read operation is performed to obtain the external pin output value.

7.4.5 Input Data Register (GPIO_IDATA)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IDATA	R	0x0	16-bit register with each bit corresponding to one IO PAD: When the GPIO is input active, the actual value of the external pin can be read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7.4.6 Interrupt Enable Register (GPIO_IEN)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IEN	R/W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Corresponding pin interrupt disabled 1: Corresponding pin interrupt enabled

7.4.7 Interrupt Trigger Mode Register (GPIO_IS)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IS	R/W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Edge trigger 1: Level trigger

7.4.8 Interrupt Edge Trigger Setting Register (GPIO_IBE)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IBE	R/W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Single-edge trigger 1: Double-edge trigger

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7.4.9 Interrupt Level Trigger Setting Register (GPIO_IEV)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IEV	R/W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Falling-edge/low-level trigger 1: Rising-edge/high-level trigger

7.4.10 Interrupt Clear Register (GPIO_IC)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IC	W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Invalid operation 1: Clear interrupt of the corresponding pin

7.4.11 Raw Interrupt Status Register (GPIO_RIS)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	RIS	R	0x0	16-bit register with each bit corresponding to one IO PAD: 0: No interrupt pending on the corresponding pin 1: With interrupt pending on the corresponding pin

7.4.12 Masked Interrupt Status Register (GPIO_MIS)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	MIS	R	0x0	16-bit register with each bit corresponding to one IO PAD: 0: No interrupt on the corresponding pin output to the system 1: With interrupt on the corresponding pin output to the system

7.4.13 Filter Enable Register (GPIO_DBEN)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	DBEN	R/W	0x0	16-bit register with each bit corresponding to one IO PAD: 0: Input filter disabled 1: Input filter enabled Note: When using this register, only one IO can be enabled for input filtering at a time. Multiple IOs cannot be enabled for input filtering simultaneously.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7.4.14 Filter Length Register (GPIO_DBL)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DBL	R/W	0x0	32-bit register, number of filter cycles

7.4.15 Configuration Lock Register (GPIO_LOCK)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	LOCK	R/W	0x0	16-bit register with each bit corresponding to one IO PAD. Setting it to 1 to lock the configuration of corresponding IO until the next reset. The frozen registers are GPIO_MODE, GPIO_IM, GPIO_PULL, GPIO_SR, GPIO_DS, GPIO_AFH and GPIO_AFL.

7.4.16 Input Mode Register (GPIO_IM)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	IM	R/W	0x0	16-bit register with each bit corresponding to one IO PAD; configure the GPIO input mode: 0: CMOS 1: Schmitt trigger

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7.4.17 Pull-up/down Register (GPIO_PULL)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	PE	R/W	0x0	16-bit register with each bit corresponding to one IO PAD; pull-up/down enable selection: 0: Disabled 1: Enabled
15:0	PS	R/W	0x0	16-bit register with each bit corresponding to one IO PAD; select the pull direction: 0: Pull down 1: Pull up

7.4.18 Driving Speed Register (GPIO_SR)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SR	R/W	0x0	16-bit register with each bit corresponding to one IO PAD; select the GPIO driving speed: 0: High speed 1: Low speed

7.4.19 Driving Capability Register (GPIO_DS)

Offset address: 0x4C

Reset value: 0x5555 5555

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	DS	R/W	0x5555_5555	32-bit register with every 2 bits corresponding to one IO PAD; configure the GPIO output driving capability: 00: 2 mA 01: 4 mA 10: 8 mA 11: 12 mA

7.4.20 Alternate Function Low Register (GPIO_AFL)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	AFL	R/W	0x0	32-bit register with every 4 bits corresponding to one IO pad (0–7), select the GPIO alternate function: 0000: Alternate function 0 0001: Alternate function 1 0010: Alternate function 2 … 1111: Alternate function 15 (Corresponding to AF0–AF15 in the pin alternate function table)

7.4.21 Alternate Function High Register (GPIO_AFH)

Offset address: 0x54

Reset value: 0x0000 0000

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	AFH	R/W	0x0	32-bit register with every 4 bits corresponding to one IO pad (8–15), select the GPIO alternate function: 0000: Alternate function 0 0001: Alternate function 1 0010: Alternate function 2 … 1111: Alternate function 15 (Corresponding to AF0–AF15 in the pin alternate function table)

7.5 Operation Procedure

7.5.1 Input

1. Enable the GPIOx clock in RCM module.
2. Configure the GPIOx_MODE register to program GPIO as input.
3. Use GPIO_IDATA to obtain the input pin level.

7.5.2 Output

1. Enable the GPIOx clock in RCM module.
2. Configure the GPIOx_MODE register to program GPIO as output.
3. Use GPIO_SET/GPIO_CLR or GPIO_ODATA to set the output level.

7.5.3 Interrupt Trigger Mode

1. Enable the GPIOx clock in RCM module.
2. Configure GPIOx_MODE as input.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3. Clearing the interrupt via GPIO_IC register to avoid exceptions.
4. Configure the GPIO_IS register to select the edge or level trigger mode.
5. In edge trigger mode, configure the GPIO_IBE register to determine whether it is single-edge trigger or double-edge trigger.
6. In single-edge trigger mode, configure the GPIO_IEV register to determine whether it is rising-edge trigger or falling-edge trigger.
7. In level trigger mode, configure the GPIO_IEV register to determine whether it is high-level trigger or low-level trigger.
8. Clearing the interrupt via GPIO_IC register.
9. Configure the GPIO_IEN register to enable the interrupt of corresponding bit.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8 Nested Vectored Interrupt Controller (NVIC)

The ARM Cortex-M4 processor and the nested vectored interrupt controller (NVIC) prioritize and handle all exceptions in the handler mode. When an exception occurs, the system automatically pushes the current operating state of the processor on the stack, and then automatically pops it off the stack after executing the interrupt service subroutine (ISR).

Vector fetching is carried out in parallel with pushing the current operating state to improve the interrupt entry efficiency. The processor supports tail chaining and can realize back-to-back interrupt, greatly reducing the overhead caused by repeatedly switching operating states.

All interrupt types are shown in the following table:

Table 8-1 : Cortex-M4 Interrupt Sources

IRQ No.	Peripheral Interrupt	Peripheral Interrupt Description	Vector Address
0	WWDT	Window watchdog interrupt	0x0000_0040
1	LVD	LVD interrupt	0x0000_0044
2	RTC_TAMPER	RTC tamper and timestamp interrupts	0x0000_0048
3	-	Reserved	-
4	EFC	EFC interrupt	0x0000_0050
5	RCM	Clock reset interrupt	0x0000_0054
6	EXTI0	EXTI[0] line interrupt	0x0000_0058
7	EXTI1	EXTI[1] line interrupt	0x0000_005C
8	EXTI2	EXTI[2] line interrupt	0x0000_0060
9	EXTI3	EXTI[3] line interrupt	0x0000_0064
10	EXTI4	EXTI[4] line interrupt	0x0000_0068
11	DMA1_CH0	DMA controller 1 channel 0 global interrupt	0x0000_006C
12	DMA1_CH1	DMA controller 1 channel 1 global interrupt	0x0000_0070
13	DMA1_CH2	DMA controller 1 channel 2 global interrupt	0x0000_0074
14	DMA1_CH3	DMA controller 1 channel 3 global interrupt	0x0000_0078

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

IRQ No.	Peripheral Interrupt	Peripheral Interrupt Description	Vector Address
15	DMA1_CH4	DMA controller 1 channel 4 global interrupt	0x0000_007C
16	DMA1_CH5	DMA controller 1 channel 5 global interrupt	0x0000_0080
17	DMA1_CH6	DMA controller 1 channel 6 global interrupt	0x0000_0084
18	DMA1_CH7	DMA controller 1 channel 7 global interrupt	0x0000_0088
19	ADC1	ADC1 global interrupt	0x0000_008C
20	ADC2	ADC2 global interrupt	0x0000_0090
21	CAN1	CAN1 global interrupt	0x0000_0094
22	CAN2	CAN2 global interrupt	0x0000_0098
23	EXTI9–5	EXTI[9:5] line interrupt	0x0000_009C
24	TIM1_BRK_TIM9	TIM1 break interrupt / TIM9 global interrupt	0x0000_00A0
25	TIM1_UP_TIM10	TIM1 update interrupt / TIM10 global interrupt	0x0000_00A4
26	TIM1_TRG_COM_TIM11	TIM1 trigger and commutation interrupts / TIM11 global interrupt	0x0000_00A8
27	TIM1_CC	TIM1 capture compare interrupt	0x0000_00AC
28	TIM2	TIM2 global interrupt	0x0000_00B0
29	TIM3	TIM3 global interrupt	0x0000_00B4
30	TIM4	TIM4 global interrupt	0x0000_00B8
31	I2C1	I2C1 global interrupt	0x0000_00BC
32	I2C2	I2C2 global interrupt	0x0000_00C0
34~33	-	Reserved	-
35	SPI1	SPI1 global interrupt	0x0000_00CC
36	SPI2	SPI2 global interrupt	0x0000_00D0
37	UART1	UART1 global interrupt	0x0000_00D4
38	UART2	UART2 global interrupt	0x0000_00D8
39	UART3	UART3 global interrupt	0x0000_00DC
40	EXTI15–10	EXTI[15:10] line interrupt	0x0000_00E0
41	RTC_ALARM	RTC alarm interrupt	0x0000_00E4
42	-	Reserved	-
43	TIM8_BRK_TIM12	TIM8 break interrupt / TIM12 global interrupt	0x0000_00EC
44	TIM8_UP_TIM13	TIM8 update interrupt / TIM13 global interrupt	0x0000_00F0
45	TIM8_TRG_COM_TIM14	TIM8 trigger and commutation interrupt / TIM14 global interrupt	0x0000_00F4

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

IRQ No.	Peripheral Interrupt	Peripheral Interrupt Description	Vector Address
46	TIM8_CC	TIM8 capture compare interrupt	0x0000_00F8
47	SDIO	SDIO global interrupt	0x0000_00FC
49~48	-	Reserved	-
50	TIM5	TIM5 global interrupt	0x0000_0108
51	SPI3	SPI3 global interrupt	0x0000_010C
52	UART4	UART4 global interrupt	0x0000_0110
53	UART5	UART5 global interrupt	0x0000_0114
54	TIM6	TIM6 global interrupt	0x0000_0118
55	TIM7	TIM7 global interrupt	0x0000_011C
56	DMA2_CH0	DMA2 channel 2 global interrupt	0x0000_0120
57	DMA2_CH1	DMA2 channel 2 global interrupt	0x0000_0124
58	DMA2_CH2	DMA2 channel 2 global interrupt	0x0000_0128
59	DMA2_CH3	DMA2 channel 2 global interrupt	0x0000_012C
60	DMA2_CH4	DMA2 channel 2 global interrupt	0x0000_0130
61	EMAC	EMAC global interrupt	0x0000_0134
62	SPI4	SPI4 global interrupt	0x0000_0138
63	-	Reserved	-
64	TS	Temperature sensor (TS) interrupt	0x0000_0140
65	OPA1	OPA1 interrupt	0x0000_0144
66	OPA2	OPA2 interrupt	0x0000_0148
67	DAC	DAC global interrupt	0x0000_014C
68	DMA2_CH5	DMA2 channel 2 global interrupt	0x0000_0150
69	DMA2_CH6	DMA2 channel 2 global interrupt	0x0000_0154
70	DMA2_CH7	DMA2 channel 2 global interrupt	0x0000_0158
71	UART6	UART6 global interrupt	0x0000_015C
72	I2C3	I2C3 global interrupt	0x0000_0160
73	USB CONTROLLER	USB controller global interrupt	0x0000_0164
75~74	-	Reserved	-
76	USART7	USART7 global interrupt	0x0000_0170
77	USART8	USART8 global interrupt	0x0000_0174
78	DCMI	DCMI global interrupt	0x0000_0178
79	AES	AES global interrupt	0x0000_017C
80	SHA	SHA global interrupt	0x0000_0180
81	FPU	FPU global interrupt	0x0000_0184

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

IRQ No.	Peripheral Interrupt	Peripheral Interrupt Description	Vector Address
82	ACMP1	ACMP1 interrupt	0x0000_0188
83	ACMP2	ACMP2 interrupt	0x0000_018C
84	ACMP3	ACMP3 interrupt	0x0000_0190
85	I2S1	I2S1 global interrupt	0x0000_0194
86	I2S2	I2S2 global interrupt	0x0000_0198
89–87	-	Reserved	-
90	QSPI	QSPI global interrupt	0x0000_01A8
91	-	Reserved	-
92	-	Reserved	-
93	IWDT	IWDT global interrupt	0x0000_01B4
106–94	-	Reserved	-
107	LPUART	LPUART global interrupt	0x0000_01EC
108	-	Reserved	-
109	LPTIM1	LPTIM1 global interrupt	0x0000_01F4
110	LPTIM2	LPTIM2 global interrupt	0x0000_01F8
118–111	-	Reserved	-

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9 System Configuration Controller (SYSCFG)

Register base address: 0x40B0_2000

The registers are listed below:

Table 9-1: List of SYSCFG Registers

Offset Address	Name	Description
0x04	SYSCFG_EMACMR	EMAC mode control register
0x08	SYSCFG_ADCETSR	ADC external trigger selection register
0x0C	SYSCFG_TIMCFGR	TIM break control register
0x10	SYSCFG_EXTICR0	External interrupt configuration register 0
0x14	SYSCFG_EXTICR1	External interrupt configuration register 1
0x18	SYSCFG_EXTICR2	External interrupt configuration register 2
0x1C	SYSCFG_EXTICR3	External interrupt configuration register 3
0x20	SYSCFG_EXTIWR	External interrupt wakeup configuration register
0x24	SYSCFG_MISCCR	MISC control register
0x28	SYSCFG_SCRVR	Systick counter reference value register

9.1 Register Description

9.1.1 EMAC Mode Control Register (SYSCFG_EMACMR)

Address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	R	0	Reserved
2:0	EMACMR	R/W	0	EMAC mode control register (connected to external MAC PHY): 3’b000: MII

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				3’b001: RGMII 3’b010: Reserved 3’b011: Reserved 3’b100: RMII 3’b101: Reserved 3’b110: Reserved 3’b111: Reserved

9.1.2 ADC External Trigger Selection Register (SYSCFG_ADCETSR)

Address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	R	0	Reserved
6:4	INJ_TRIG_SEL	R/W	0	External trigger source selection in ADC injection mode: 000: EXTI15 001: EXTI9 010: EXTI5 011: LPTIM1_OUT 100: LPTIM2_OUT 101: Reserved 110: Reserved 111: Reserved
3	RSV	R	0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2:0	RGL_TRIG_SEL	R/W	0	External trigger source selection in ADC regular mode: 000: EXTI11 001: EXTI10 010: EXTI6 011: LPTIM1_OUT 100: LPTIM2_OUT 101: Reserved 110: Reserved 111: Reserved

9.1.3 TIM Break Control Register (SYSCFG_TIMCFGR)

Address: 0x0C

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	R	0	Reserved
4	TIMx_DEBUG_STOP	R/W	0	TIMx debug control signal: 0: Invalid 1: Valid
3:2	RSV	R	0	Reserved
1	TIM8_BRKR	R/W	1	TIM8 break mode control register: 1: OCx and OCxn being 0 at break 0: OCx and OCxn being OCxp and OCxnp respectively at break
0	TIM1_BRKR	R/W	1	TIM1 break mode control register: 1: OCx and OCxn being 0 at break 0: OCx and OCxn being OCxp and OCxnp respectively at break

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9.1.4 External Interrupt Configuration Register 0 (SYSCFG_EXTICR0)

Address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	R	0	Reserved
27	EXTI3_WK_FLAG	R	0	EXTI3 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
26	EXTI2_WK_FLAG	R	0	EXTI2 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
25	EXTI1_WK_FLAG	R	0	EXTI1 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
24	EXTI0_WK_FLAG	R	0	EXTI0 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
23	EXTI3_WK_EDGE_SEL	R/W	0	EXTI3 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
22	EXTI2_WK_EDGE_SEL	R/W	0	EXTI2 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
21	EXTI1_WK_EDGE_SEL	R/W	0	EXTI1 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
20	EXTI0_WK_EDGE_SEL	R/W	0	EXTI0 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
19	EXTI3_WKEN	R/W	0	EXTI3 wakeup enable in Stop mode: 0: Disabled 1: Enabled
18	EXTI2_WKEN	R/W	0	EXTI2 wakeup enable in Stop mode: 0: Disabled 1: Enabled
17	EXTI1_WKEN	R/W	0	EXTI1 wakeup enable in Stop mode: 0: Disabled 1: Enabled
16	EXTI0_WKEN	R/W	0	EXTI0 wakeup enable in Stop mode: 0: Disabled 1: Enabled
15:12	EXTI3	R/W	0	External interrupt configuration bit: 4’b0000: PA[3] pin 4’b0001: PB[3] pin 4’b0010: PC[3] pin 4’b0011: PD[3] pin 4’b0100: PE[3] pin 4’b0101: Reserved 4’b0110: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4’b0111: Reserved 4’b1000: Reserved
11:8	EXTI2	R/W	0	External interrupt configuration bit: 4’b0000: PA[2] pin 4’b0001: PB[2] pin 4’b0010: PC[2] pin 4’b0011: PD[2] pin 4’b0100: PE[2] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
7:4	EXTI1	R/W	0	External interrupt configuration bit: 4’b0000: PA[1] pin 4’b0001: PB[1] pin 4’b0010: PC[1] pin 4’b0011: PD[1] pin 4’b0100: PE[1] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: PH[1] pin 4’b1000: Reserved
3:0	EXTI0	R/W	0	External interrupt configuration bit: 4’b0000: PA[0] pin 4’b0001: PB[0] pin 4’b0010: PC[0] pin 4’b0011: PD[0] pin 4’b0100: PE[0] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: PH[0] pin 4’b1000: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9.1.5 External Interrupt Configuration Register 1 (SYSCFG_EXTICR1)

Address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	R	0	Reserved
27	EXTI7_WK_FLAG	R	0	EXTI7 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
26	EXTI6_WK_FLAG	R	0	EXTI6 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
25	EXTI5_WK_FLAG	R	0	EXTI5 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
24	EXTI4_WK_FLAG	R	0	EXTI4 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
23	EXTI7_WK_EDGE_SEL	R/W	0	EXTI7 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
22	EXTI6_WK_EDGE_SEL	R/W	0	EXTI6 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
21	EXTI5_WK_EDGE_SEL	R/W	0	EXTI5 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
20	EXTI4_WK_EDGE_SEL	R/W	0	EXTI4 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
19	EXTI7_WKEN	R/W	0	EXTI7 wakeup enable in Stop mode: 0: Disabled 1: Enabled
18	EXTI6_WKEN	R/W	0	EXTI6 wakeup enable in Stop mode: 0: Disabled 1: Enabled
17	EXTI5_WKEN	R/W	0	EXTI5 wakeup enable in Stop mode: 0: Disabled 1: Enabled
16	EXTI4_WKEN	R/W	0	EXTI4 wakeup enable in Stop mode: 0: Disabled 1: Enabled
15:12	EXTI7	R/W	0	External interrupt configuration bit: 4’b0000: PA[7] pin 4’b0001: PB[7] pin 4’b0010: PC[7] pin 4’b0011: PD[7] pin 4’b0100: PE[7] pin 4’b0101: Reserved 4’b0110: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4’b0111: Reserved 4’b1000: Reserved
11:8	EXTI6	R/W	0	External interrupt configuration bit: 4’b0000: PA[6] pin 4’b0001: PB[6] pin 4’b0010: PC[6] pin 4’b0011: PD[6] pin 4’b0100: PE[6] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
7:4	EXTI5	R/W	0	External interrupt configuration bit: 4’b0000: PA[5] pin 4’b0001: PB[5] pin 4’b0010: PC[5] pin 4’b0011: PD[5] pin 4’b0100: PE[5] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
3:0	EXTI4	R/W	0	External interrupt configuration bit: 4’b0000: PA[4] pin 4’b0001: PB[4] pin 4’b0010: PC[4] pin 4’b0011: PD[4] pin 4’b0100: PE[4] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9.1.6 External Interrupt Configuration Register 2 (SYSCFG_EXTICR2)

Address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	R	-	Reserved
27	EXTI11_WK_FLAG	R	0	EXTI11 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
26	EXTI10_WK_FLAG	R	0	EXTI10 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
25	EXTI9_WK_FLAG	R	0	EXTI9 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
24	EXTI8_WK_FLAG	R	0	EXTI8 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
23	EXTI11_WK_EDGE_SEL	R/W	0	EXTI11 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
22	EXTI10_WK_EDGE_SEL	R/W	0	EXTI10 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
21	EXTI9_WK_EDGE_SEL	R/W	0	EXTI9 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
20	EXTI8_WK_EDGE_SEL	R/W	0	EXTI8 wakeup edge selection in Stop mode: 0: Wakeup on rising edge

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Wakeup on falling edge
19	EXTI11_WKEN	R/W	0	EXTI11 wakeup enable in Stop mode: 0: Disabled 1: Enabled
18	EXTI10_WKEN	R/W	0	EXTI10 wakeup enable in Stop mode: 0: Disabled 1: Enabled
17	EXTI9_WKEN	R/W	0	EXTI9 wakeup enable in Stop mode: 0: Disabled 1: Enabled
16	EXTI8_WKEN	R/W	0	EXTI8 wakeup enable in Stop mode: 0: Disabled 1: Enabled
15:12	EXTI11	R/W	0	External interrupt configuration bit: 4'b0000: PA[11] pin 4'b0001: PB[11] pin 4'b0010: PC[11] pin 4'b0011: PD[11] pin 4'b0100: PE[11] pin 4'b0101: Reserved 4'b0110: Reserved 4'b0111: Reserved 4'b1000: Reserved
11:8	EXTI10	R/W	0	External interrupt configuration bit: 4'b0000: PA[10] pin 4'b0001: PB[10] pin 4'b0010: PC[10] pin 4'b0011: PD[10] pin 4'b0100: PE[10] pin 4'b0101: Reserved 4'b0110: Reserved 4'b0111: Reserved 4'b1000: Reserved
7:4	EXTI9	R/W	0	External interrupt configuration bit: 4'b0000: PA[9] pin 4'b0001: PB[9] pin 4'b0010: PC[9] pin

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4'b0011: PD[9] pin 4'b0100: PE[9] pin 4'b0101: Reserved 4'b0110: Reserved 4'b0111: Reserved 4'b1000: Reserved
3:0	EXTI8	R/W	0	External interrupt configuration bit: 4'b0000: PA[8] pin 4'b0001: PB[8] pin 4'b0010: PC[8] pin 4'b0011: PD[8] pin 4'b0100: PE[8] pin 4'b0101: Reserved 4'b0110: Reserved 4'b0111: Reserved 4'b1000: Reserved

9.1.7 External Interrupt Configuration Register 3 (SYSCFG_EXTICR3)

Address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	R	0	Reserved
27	EXTI15_WK_FLAG	R	0	EXTI15 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
26	EXTI14_WK_FLAG	R	0	EXTI14 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
25	EXTI13_WK_FLAG	R	0	EXTI13 wakeup source flag in Stop mode: 0: Not woken up by setting this bit

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Woken up by setting this bit
24	EXTI12_WK_FLAG	R	0	EXTI12 wakeup source flag in Stop mode: 0: Not woken up by setting this bit 1: Woken up by setting this bit
23	EXTI15_WK_EDGE_SEL	R/W	0	EXTI15 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
22	EXTI14_WK_EDGE_SEL	R/W	0	EXTI14 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
21	EXTI13_WK_EDGE_SEL	R/W	0	EXTI13 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
20	EXTI12_WK_EDGE_SEL	R/W	0	EXTI12 wakeup edge selection in Stop mode: 0: Wakeup on rising edge 1: Wakeup on falling edge
19	EXTI15_WKEN	R/W	0	EXTI15 wakeup enable in Stop mode: 0: Disabled 1: Enabled
18	EXTI14_WKEN	R/W	0	EXTI14 wakeup enable in Stop mode: 0: Disabled 1: Enabled
17	EXTI13_WKEN	R/W	0	EXTI13 wakeup enable in Stop mode: 0: Disabled 1: Enabled
16	EXTI12_WKEN	R/W	0	EXTI12 wakeup enable in Stop mode: 0: Disabled 1: Enabled
15:12	EXTI15	R/W	0	External interrupt configuration bit: 4'b0000: PA[15] pin 4'b0001: PB[15] pin 4'b0010: PC[15] pin

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				4’b0011: PD[15] pin 4’b0100: PE[15] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
11:8	EXTI14	R/W	0	External interrupt configuration bit: 4’b0000: PA[14] pin 4’b0001: PB[14] pin 4’b0010: PC[14] pin 4’b0011: PD[14] pin 4’b0100: PE[14] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
7:4	EXTI13	R/W	0	External interrupt configuration bit: 4’b0000: PA[13] pin 4’b0001: PB[13] pin 4’b0010: PC[13] pin 4’b0011: PD[13] pin 4’b0100: PE[13] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved
3:0	EXTI12	R/W	0	External interrupt configuration bit: 4’b0000: PA[12] pin 4’b0001: PB[12] pin 4’b0010: PC[12] pin 4’b0011: PD[12] pin 4’b0100: PE[12] pin 4’b0101: Reserved 4’b0110: Reserved 4’b0111: Reserved 4’b1000: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9.1.8 External Interrupt Wakeup Configuration Register (SYSCFG_EXTIWR)

Address: 0x20

Reset value: 0x0002 0000

Bit	Name	Attribute	Reset Value	Description
31:18	RSV	R	0	Reserved
17	INT_SEL	R/W	1	Interrupt response mode: 0: Reserved 1: One of multiple GPIOs selected as interrupt Note: This bit shall not be modified and must be 1.
16	IESEL	R/W	0	Port interrupt mode selection bit: 1: Stop mode 0: Run mode Notes: ● When the system is in Run mode, the system clock will not be turned off, and IESEL can be set to 0. At this time, the external signal source that triggers the port interrupt will generate an interrupt signal after synchronization with the system clock, which can filter out glitches of the external signal source. ● When the system is in Stop mode, the system clock will be turned off, and IESEL can be set to 1. At this time, the external signal source that triggers the port interrupt will generate an interrupt signal directly without filtering out glitches of the external signal source.
15	EXTI15_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI15 (INT_SEL = 1),

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
14	EXTI14_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI14 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
13	EXTI13_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI13 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
12	EXTI12_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI12 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
11	EXTI11_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI11 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
10	EXTI10_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI10 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
9	EXTI9_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI9 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
8	EXTI8_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI8 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
7	EXTI7_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI7 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
6	EXTI6_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI6 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
5	EXTI5_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI5 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: No operation
4	EXTI4_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI4 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
3	EXTI3_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI3 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
2	EXTI2_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI2 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
1	EXTI1_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI1 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation
0	EXTI0_CLR	R/W	0	When the IESEL bit in the register is 1 and the wakeup source is EXTI0 (INT_SEL = 1), write this bit to 1 after the system wakes up to clear the GPIO wakeup interrupt. 1: Clear GPIO wakeup interrupt after system wakeup 0: No operation

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9.1.9 MISC Control Register (SYSCFG_MISCCR)

Address: 0x24

Reset value: 0x0000 0030

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	R	0	Reserved
5	EFC_VOL_PD_EN	R/W	1	EFC power-down ready signal (Standby mode): 0: Invalid 1: Valid Note: Generally not modified.
4	EFC_CLK_PD_EN	R/W	1	EFC clock stop ready signal (Stop mode): 0: Invalid 1: Valid Note: Generally not modified.
3	RSV	R	0	Reserved
2	LVD_INT_EN	R/W	0	LVD interrupt enable: 0: LVD interrupt disabled 1: LVD interrupt enabled
1	RSV	R		Reserved
0	NMIEN	R/W	0	NMI enable: 0: NMI disabled 1: NMI enabled (LVD interrupt)

9.1.10 SysTick Calibration Reference Value Register (SYSCFG_SCRVR)

Address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	R	0	Reserved
23:0	STCALIB	R/W	0	SysTick calibration reference value

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10 Cyclic Redundancy Check Calculation Unit (CRC)

10.1 Overview

The CRC controller can perform CRC calculation using various polynomials based on CRC32 and CRC16 standards.

10.2 Main Features

- Uses the following polynomials:

  $x^{16} + x^{12} + x^5 + 1$

  $x^{16} + x^{15} + x^2 + 1$

  $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$

- Can input byte, half-word and full-word data
- The input data can be reversed by bit, half-word, byte, initial value and result value.

10.3 Functional Description

10.3.1 CRC Calculation

CRC calculation can be performed in multiple formats.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10.3.2 Common CRC Formats

Table 10-1: Common CRC Formats

Name	Polynomial	Initial Value	Input Data Reversed by Byte	Output Data Reversed	Result XOR
CRC-16 / CCITT	1021	0	Y	Y	0
CRC-16 / CCITT-FALSE	1021	FFFF	N	N	0
CRC-16 / X25	1021	FFFF	Y	Y	FFFF
CRC-16 / XMODEM	1021	0	N	N	0
CRC-16 / IBM	8005	0	Y	Y	0
CRC-16 / MAXIM	8005	0	Y	Y	FFFF
CRC-16 / USB	8005	FFFF	Y	Y	FFFF
CRC-16 / MODBUS	8005	FFFF	Y	Y	0
CRC-32	04C11DB7	FFFFFFFF	Y	Y	FFFFFFFF
CRC-32 / MPEG-2	04C11DB7	FFFFFFFF	N	N	0

Note: XOR function for results is not provided in this module. If necessary, please take out the results and then calculate them separately.

10.4 Register Description

Register base address: 0x40A0_0000

The registers are listed below:

Table 10-2: List of CRC Registers

Offset Address	Name	Description
0x00	CRC_DATA	Data register
0x04	CRC_CFG	Configuration register
0x08	CRC_INIT	Initial value register

10.4.1 Data Register (CRC_DATA)

Offset address: 0x00

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	DATA	R/W	0x0	This register is used to write new data to the CRC calculator, and it holds the previous CRC calculation result when it is read.

10.4.2 Configuration Register (CRC_CFG)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
30:9	RSV	-	-	Reserved
8	INIT_REV	R/W	0x0	Initial value reverse: 0: Not reversed 1: Reversed
7	DOUT_REV	R/W	0x0	Output data reverse: 0: Not reversed 1: Reversed
6	RSV	-	-	Reserved
5	DIN_REV	R/W	0x0	Input data reverse: 0: Not reversed 1: Reversed
4:3	WIDTH_DIN	R/W	0x0	Input data width selection: 00: 8 bits 01: 16 bits 10/11: 32 bits
2:1	POL	R/W	0x0	CRC polynomial selection: 00: CRC16-1021 ($x^{16} + x^{12} + x^5 + 1$); 01: CRC16-8005 ($x^{16} + x^{15} + x^2 + 1$); 10/11: CRC32-04C11DB7 ($x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$)
0	RESET	W	0x0	Writing 1 to this bit will clear the calculation result and load the initial value, and this bit will be automatically cleared.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10.4.3 Initial Value Register (CRC_INIT)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	INIT	R/W	0x0	Write the initial value.

10.5 Operation Procedure

1. Configure the CRC_INIT register to determine the initial value.
2. Configure the CRC_CFG register to select the CRC polynomial, data width and data reverse order, and load the initial value.
3. Write data to the CRC_DATA register, which can be written continuously.
4. Fetch the CRC calculation result from the CRC_DATA register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11 Direct Memory Access Controller (DMA)

11.1 Overview

The DMA is used to provide high-speed data transfer between peripherals and memory as well as from memory to memory. Data can be quickly moved by DMA without any CPU actions, which keeps the CPU resources free for other operations, thus improving the system efficiency.

11.2 Main Features

- Controllable data transmission among multiple modules
- Memory-to-memory, memory-to-peripheral, peripheral-to-memory, and peripheral-to-peripheral data transfers
- Provided with 8 DMA channels
- Configurable bit width and block length of data transfer
- Each channel with 4 x 32-bit FIFO
- Block length up to 4095
- Invariant transmission and incremental transmission of source address
- Invariant transmission and incremental transmission of destination address

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11.3 Functional Description

11.3.1 Flow Control

DMA supports transfers where the DMA itself acts as a flow controller. For source and destination transfer, the following modes are supported:

- Memory-to-memory
- Memory-to-peripheral
- Peripheral-to-memory
- Peripheral-to-peripheral

11.3.2 Handshake Signal

When both the source and destination are set as peripheral, it is necessary to specify the handshake signal number used by the peripheral. DMA will initiate a transfer only when the handshake signal is valid.

The handshake signal can be generated not only by peripheral hardware, but also by software writing DMA register.

The DMA handshake signals are assigned as follows:

Table 11-1: DMA1 Handshake Signal

	HS0	HS1	HS2	HS3	HS4	HS5	HS6	HS7	HS8	HS9	HS10	HS11	HS12	HS13
Peripheral Request	SPI3_RX	TIM2_UP	SPI3_RX	SPI2_RX	SPI2_TX	SPI3_TX	I2C1_TX	SPI3_TX	QSPI_RX	QSPI_TX	TIM12_CH4	TIM12_UP	TIM12_TRIG	TIM12_CH1
	I2C1_RX	TIM2_CH3	TIM7_UP	TIM4_CH2	TIM7_UP	I2C1_RX	TIM4_UP	I2C1_RX	TIM12_CH2	TIM12_CH3	TIM13_CH3	TIM13_CH4	TIM13_UP	TIM13_TRIG
	TIM4_CH1	UART3_RX	I2S2_RX	I2S1_RX	I2S1_TX	I2S2_TX	TIM2_CH2	TIM4_CH3	TIM13_CH1	TIM13_CH2	TIM14_CH2	TIM14_CH3	TIM14_CH4	TIM14_UP
	I2S2_RX	TIM5_CH4	I2C3_RX	UART3_TX	I2C3_TX	TIM2_CH1	TIM2_CH4	TIM2_CH4	TIM14_TRIG	TIM14_CH1	USART7_RX	USART6_TX;		
	UART5_RX	TIM5_TRIG	UART4_RX	TIM5_TRIG	UART4_TX	UART2_RX	UART2_TX	TIM2_UP	TIM4_CH4	TIM4_TRIG	TIM2_TRIG			
	TIM5_CH3	TIM6_UP	TIM3_CH4	TIM5_CH4	TIM3_TRIG	TIM3_CH2	TIM5_UP	UART5_TX						
	TIM5_UP		TIM3_UP	I2C2_RX	TIM3_CH1	DAC1	DAC2	TIM3_CH3						
			TIM5_CH1		TIM5_CH2									
			I2C2_RX		UART3_TX									

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 11-2: DMA2 Handshake Signal

	HS0	HS1	HS2	HS3	HS4	HS5	HS6	HS7	HS8	HS9	HS10	HS11	HS12	HS13
Peripheral Request	ADC1	DCMI	TIM8_CH1	ADC2	ADC1	QSPI_TX	TIM1_CH1	DCMI	SPI4_RX	SPI4_TX	TIM9_CH1	TIM9_CH2	CORDIC_IN	CORDIC_OUT
	SPI1_RX	UART6_RX	TIM8_CH2	SPI1_TX	TIM1_CH4	AES_OUT	TIM1_CH2	SHA_IN	TIM9_UP	TIM9_TRIG	TIM10_TRIG	TIM10_CH1	TIM9_CH3	TIM9_CH4
	TIM1_TRIG	TIM1_CH1	TIM8_CH3	TIM1_CH1	TIM1_TRIG	SPI1_TX	TIM1_CH3	UART1_TX	TIM10_CH4	TIM10_UP	TIM11_UP	TIM11_TRIG	TIM10_CH2	TIM10_CH3
		TIM8_UP	ADC2	TIM1_CH1	TIM1_COM	UART1_RX	QSPI_RX	UART6_TX	TIM11_CH3	TIM11_CH4	USART8_RX	USART8_TX	TIM11_CH1	TIM11_CH2
			SPI1_RX	TIM8_CH2	TIM8_CH3	TIM1_UP	UART6_TX	TIM8_CH4						
			UART1_RX					TIM8_TRIG						
			UART6_RX					TIM8_COM						
			TIM1_CH2											

Notes:

- Note that the same handshake signal can only be responded by one of multiple peripherals at a time, that is, DMA cannot be triggered by multiple peripherals at the same time.
- The handshake signal (peripheral) and the 8 channels of DMA can be freely configured according to the application.
- The handshake signal is fixed and can only be set from the corresponding peripheral in the table.
- The 8 channels of DMA can be freely configured by software to adapt to different peripheral applications.

11.4 Register Description

DMA1 register base address: 0x4070_0000

DMA2 register base address: 0x4080_0000

The registers are listed below:

Table 11-3: List of DMA Registers

Offset Address	Name	Description
0x00	DMA_SAR0	Channel 0 source address register
0x08	DMA_DAR0	Channel 0 destination address register
0x18	DMA_CTL0	Channel 0 control register
0x40	DMA_CFG0	Channel 0 configuration register
0x58	DMA_SAR1	Channel 1 source address register
0x60	DMA_DAR1	Channel 1 destination address register
0x70	DMA_CTL1	Channel 1 control register
0x98	DMA_CFG1	Channel 1 configuration register
0xB0	DMA_SAR2	Channel 2 source address register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0xB8	DMA_DAR2	Channel 2 destination address register
0xC8	DMA_CTL2	Channel 2 control register
0xF0	DMA_CFG2	Channel 2 configuration register
0x108	DMA_SAR3	Channel 3 source address register
0x110	DMA_DAR3	Channel 3 destination address register
0x120	DMA_CTL3	Channel 3 control register
0x148	DMA_CFG3	Channel 3 configuration register
0x160	DMA_SAR4	Channel 4 source address register
0x168	DMA_DAR4	Channel 4 destination address register
0x178	DMA_CTL4	Channel 4 control register
0x1A0	DMA_CFG4	Channel 4 configuration register
0x1B8	DMA_SAR5	Channel 5 source address register
0x1C0	DMA_DAR5	Channel 5 destination address register
0x1D0	DMA_CTL5	Channel 5 control register
0x1F8	DMA_CFG5	Channel 5 configuration register
0x210	DMA_SAR6	Channel 6 source address register
0x218	DMA_DAR6	Channel 6 destination address register
0x228	DMA_CTL6	Channel 6 control register
0x250	DMA_CFG6	Channel 6 configuration register
0x268	DMA_SAR7	Channel 7 source address register
0x270	DMA_DAR7	Channel 7 destination address register
0x280	DMA_CTL7	Channel 7 control register
0x2A8	DMA_CFG7	Channel 7 configuration register
0x2C0	DMA_RAWTFR	Raw transfer interrupt register
0x2C8	DMA_RAWBLOCK	Raw block transfer interrupt register
0x2D0	DMA_RAWSRCTRAN	Raw source transfer interrupt register
0x2D8	DMA_RAWDSTTRAN	Raw destination transfer interrupt register
0x2E0	DMA_RAWERR	Raw error interrupt register
0x2E8	DMA_STATUSTFR	Transfer interrupt status register
0x2F0	DMA_STATUSBLOCK	Block transfer interrupt status register
0x2F8	DMA_STATUSSRCTRAN	Source transfer interrupt status register
0x300	DMA_STATUSDSTTRAN	Destination transfer interrupt status register
0x308	DMA_STATUSERR	Error interrupt status register
0x310	DMA_MASKTFR	Transfer interrupt mask register
0x318	DMA_MASKBLOCK	Block transfer interrupt mask register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x320	DMA_MASKSRCTRAN	Source transfer interrupt mask register
0x328	DMA_MASKDSTTRAN	Destination transfer interrupt mask register
0x330	DMA_MASKERR	Error interrupt mask register
0x338	DMA_CLEARTFR	Transfer interrupt clear register
0x340	DMA_CLEARBLOCK	Block transfer interrupt clear register
0x348	DMA_CLEARSRCTRAN	Source transfer interrupt clear register
0x350	DMA_CLEARDSTTRAN	Destination transfer interrupt clear register
0x358	DMA_CLEARERR	Error interrupt clear register
0x360	DMA_STATUSINT	Interrupt status register
0x368	DMA_REQSRCREG	Source transfer request signal register
0x370	DMA_REQDSTREG	Destination transfer Req signal register
0x378	DMA_SGLREQSRCREG	Source transfer single signal register
0x380	DMA_SGLREQDSTREG	Destination transfer single signal register
0x388	DMA_LSTSRCREG	Source transfer last signal register
0x390	DMA_LSTDSTREG	Destination transfer last signal register
0x398	DMA_CFGREG	DMA module enable register
0x3A0	DMA_CHENREG	Channel enable register

11.4.1 Source Address Register (DMA_SARx)

Offset address: 0x00/0x58/0xB0/0x108/0x160/0x1B8/0x210/0x268, x = 0−7

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SAR	R/W	0x0	DMA transfer source address, automatically updated

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.2 Destination Address Register (DMA_DARx)

Offset address: 0x08/0x60/0xB8/0x110/0x168/0x1C0/0x218/0x270, x = 0−7

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	DAR	R/W	0x0	DMA transfer destination address, automatically updated

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.3 Control Register (DMA_CTLx)

Offset address: 0x18/0x70/0xC8/0x120/0x178/0x1D0/0x228/0x280, x = 0−7

Reset value: 0x0030 4801

Bit	Name	Attribute	Reset Value	Description
31:22	RSV	-	-	Reserved
21:20	TT_FC	R/W	0x3	Transfer type and flow control: 0x0: Memory-to-memory 0x1: Memory-to-peripheral 0x2: Peripheral-to-memory 0x3: Peripheral-to-peripheral
19:17	RSV	-	-	Reserved
16:14	SRC_MSIZE	R/W	0x1	Burst length of source transfer, the number of data read each time the source peripheral request handshake signal is available: 0x0: 1 0x1: 4 0x2: 8 0x3: 16 Others: Reserved
13:11	DST_MSIZE	R/W	0x1	Burst length of destination transfer, the number of data written each time the destination peripheral request handshake signal is available: 0x0: 1 0x1: 4 0x2: 8 0x3: 16 Others: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10:9	SINC	R/W	0x0	Source address incrementation: 0x0: Increment 0x1: Decrement 0x2/0x3: Invariant
8:7	DINC	R/W	0x0	Destination address incrementation: 0x0: Increment 0x1: Decrement 0x2/0x3: Invariant
6	RSV	-	-	Reserved
5:4	SRC_TR_WIDTH	R/W	0x0	Source transfer data width: 0x0: 8 bits 0x1: 16 bits 0x2: 32 bits
3	RSV	-	-	Reserved
2:1	DST_TR_WIDTH	R/W	0x0	Destination transfer data width: 0x0: 8 bits 0x1: 16 bits 0x2: 32 bits
0	INT_EN	R/W	0x1	Interrupt enable: 0x0: Disabled 0x1: Enabled

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.4 Control Register (DMA_CTLHx)

Offset address: 0x1C/0x74/0xCC/0x124/0x17C/0x1D4/0x22C/0x284, x = 0−7

Reset value: 0x0000 0002

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
11:0	BLOCK_TS	R/W	0x2	The block transfer source data size is in the unit of source transfer width, the total number of transfer is determined by the source transfer, and the number of destination transfer varies automatically according to the bit widths of source and destination transfers. Once the transfer begins, the readback value is the total number of data that has been read from the source.

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.5 Configuration Register (DMA_CFGx)

Offset address: 0x40/0x98/0xF0/0x148/0x1A0/0x1F8/0x250/0x2A8, x = 0−7

Reset value: 0x0000_0e00 + 0x20 * x (x = 0−7)

Bit	Name	Attribute	Reset Value	Description
31	RELOAD_DST	R/W	0x0	Auto-reload a destination transfer: 0x0: Disabled 0x1: Enabled; when a block transfer ends, reset DARx to its initial value and start a new block transfer.
30	RELOAD_SRC	R/W	0x0	Auto-reload a source transfer: 0x0: Disabled 0x1: Enabled; when a block transfer ends, reset SARx to its initial value and start a new block transfer.
29:20	MAX_ABRST	R/W	0x0	Maximum AMBA burst length for DMA transfers on this channel. 0 indicates no software limit on the maximum AMBA burst length for DMA transfers on this channel.
19	SRC_HS_POL	R/W	0x0	Please use 0x0: handshake signal active high

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
18	DST_HS_POL	R/W	0x0	Please use 0x0: handshake signal active high
17:12	RSV	-	-	Reserved
11	HS_SEL_SRC	R/W	0x1	Source transfer handshake signal selection: 0x0: Hardware handshake 0x1: Software handshake
10	HS_SEL_DST	R/W	0x1	Destination transfer handshake signal selection: 0x0: Hardware handshake 0x1: Software handshake
9	FIFO_EMPTY	R	0x1	FIFO empty indication for this channel: 0x0: FIFO non-empty 0x1: FIFO empty
8	CH_SUSP	R/W	0x0	Suspend transfer on this channel: 0x0: Normal transfer 0x1: Transfer suspended
7:5	CH_PRIOR	R/W	Different for each channel	Specify the channel priority, with 0 as the lowest
4:0	RSV	-	-	Reserved

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.6 Configuration Register (DMA_CFGHx)

Offset address: 0x44/0x9C/0xF4/0x14C/0x1A4/0x1FC/0x254/0x2AC, x = 0–7

Reset value: 0x0000 0004

Bit	Name	Attribute	Reset Value	Description
31:15	RSV	-	-	Reserved
14:11	DEST_PER	R/W	0x0	Destination peripheral handshake signal number: This item is ignored when the destination is specified as memory in TT_FC.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10:7	SRC_PER	R/W	0x0	Source peripheral handshake signal number: This item is ignored when the source is specified as memory in TT_FC.
6:5	RSV	-	-	Reserved
4:2	PROTCTL	R/W	0x1	Drive HPROT[3:1]
1	FIFO_MODE	R/W	0x0	Specify how much available data/space is needed for a burst transfer: 0x0: Transfer can be initiated as long as there is one available data/space. 0x1: Destination transfer can be initiated when the available data is greater than or equal to half the FIFO depth; source transfer can be initiated when the space is greater than or equal to half the FIFO depth; except at the end of a burst or block transfer.
0	FCMODE	R/W	0x0	Flow control mode / data prefetch: 0x0: Prefetch enabled, read when source data transfer is available. 0x1: Prefetch disabled, source data transfer not initiated until destination data transfer is completed.

Note: This register must be set when the channel is disabled (CH_EN = 0 for DMA_CHENREG).

11.4.7 Raw Transfer Interrupt Register (DMA_RAWTFR)

Offset address: 0x2C0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	RAWTFR	R/W	0x0	Bit x indicates the raw transfer interrupt status of channel x. This interrupt is triggered when the channel completes all transfers. Note: Writing directly to this register is not recommended during normal use.

11.4.8 Raw Block Transfer Interrupt Register (DMA_RAWBLOCK)

Offset address: 0x2C8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	RAWBLOCK	R/W	0x0	Bit x indicates the raw block transfer interrupt status of channel x. This interrupt is triggered when the channel completes a block transfer. Note: Writing directly to this register is not recommended during normal use.

11.4.9 Raw Source Transfer Interrupt Register (DMA_RAWSRCTRAN)

Offset address: 0x2D0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	RAWSRCTRAN	R/W	0x0	Bit x indicates the raw source transfer interrupt status of channel x. This interrupt is triggered when the channel completes a burst/single transfer in response to a handshake signal from the source peripheral. Note: Writing directly to this register is not recommended during normal use.

11.4.10 Raw Destination Transfer Interrupt Register (DMA_RAWDSTTRAN)

Offset address: 0x2D8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	RAWDSTTRAN	R/W	0x0	Bit x indicates the raw destination transfer interrupt status of channel x. This interrupt is triggered when the channel completes a burst/single transfer in response to a handshake signal from the destination peripheral. Note: Writing directly to this register is not recommended during normal use.

11.4.11 Raw Error Interrupt Register (DMA_RAWERR)

Offset address: 0x2E0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	RAWERR	R/W	0x0	Bit x indicates the raw error interrupt status of channel x.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				This interrupt is triggered when the channel receives an error response from HRSP during transfer, causing the transfer to be canceled and the channel to be shut down. Note: Writing directly to this register is not recommended during normal use.

11.4.12 Transfer Interrupt Status Register (DMA_STATUSTFR)

Offset address: 0x2E8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	STATUSTFR	R	0x0	Transfer interrupt output status of channel x: This interrupt is triggered when the channel completes all transfers, and this register will not be set to 1 if the interrupt is masked.

Note: Writing to this register is prohibited.

11.4.13 Block Transfer Interrupt Status Register (DMA_STATUSBLOCK)

Offset address: 0x2F0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	STATUSBLOCK	R	0x0	Bit x indicates the block transfer interrupt output status of channel x. This interrupt is triggered when the channel completes a block transfer. This register will not be set to 1 if the interrupt is masked.

Note: Writing to this register is prohibited.

11.4.14 Source Transfer Interrupt Status Register (DMA_STATUSSRCTRAN)

Offset address: 0x2F8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	STATUSSRCTRAN	R	0x0	Bit x indicates the source transfer interrupt output status of channel x. This interrupt is triggered when the channel completes a burst/single transfer in response to a handshake signal from the source peripheral. This register will not be set to 1 if the interrupt is masked.

Note: Writing to this register is prohibited.

11.4.15 Destination Transfer Interrupt Status Register (DMA_STATUSDSTRAN)

Offset address: 0x300

Reset value: 0x0000 0000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	STATUSDSTTRAN	R	0x0	Bit x indicates the destination transfer interrupt output status of channel x. This interrupt is triggered when the channel completes a burst/single transfer in response to a handshake signal from the destination peripheral. This register will not be set to 1 if the interrupt is masked.

Note: Writing to this register is prohibited.

11.4.16 Error Interrupt Status Register (DMA_STATUSERR)

Offset address: 0x308

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	STATUSERR	R	0x0	Bit x indicates the error interrupt output status of channel x. This interrupt is triggered when the channel receives an error response from HRSP during transfer, causing the transfer to be canceled and the channel to be shut down. This register will not be set to 1 if the interrupt is masked.

Note: Writing to this register is prohibited.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11.4.17 Transfer Interrupt Mask Register (DMA_MASKTFR)

Offset address: 0x310

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	MASKTFR_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	MASKTFR	R/W	0x0	Bit x indicates the transfer interrupt mask status of channel x: 0x0: Masked 0x1: Not masked

For example:

- To set bit 0 to 1, write 0x101 to this register without affecting bit 1.
- To set bit 1 to 0, write 0x200 to this register without affecting bit 0.
- To set bit 0 and bit 1 to 1, write 0x303 to this register.

11.4.18 Block Transfer Interrupt Mask Register (DMA_MASKBLOCK)

Offset address: 0x318

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	MASKBLOCK_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	MASKBLOCK	R/W	0x0	Bit x indicates the block transfer interrupt mask status of channel x: 0x0: Masked 0x1: Not masked

11.4.19 Source Transfer Interrupt Mask Register (DMA_MASKSRCTRAN)

Offset address: 0x320

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	MASKSRCTRAN_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	MASKSRCTRAN	R/W	0x0	Bit x indicates the source transfer interrupt mask status of channel x: 0x0: Masked 0x1: Not masked

11.4.20 Destination Transfer Interrupt Mask Register (DMA_MASKDSTTRAN)

Offset address: 0x328

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:8	MASKDSTTRAN_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	MASKDSTTRAN	R/W	0x0	Bit x indicates the destination transfer interrupt mask status of channel x: 0x0: Masked 0x1: Not masked

11.4.21 Error Interrupt Mask Register (DMA_MASKERR)

Offset address: 0x330

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	MASKERR_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	MASKERR	R/W	0x0	Bit x indicates the error interrupt mask status of channel x: 0x0: Masked 0x1: Not masked

11.4.22 Transfer Interrupt Clear Register (DMA_CLEARTFR)

Offset address: 0x338

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	CLEARTFR	W	0x0	Writing 1 to bit x clears the transfer interrupt for channel x.

Note: Reading this register is prohibited.

11.4.23 Block Transfer Interrupt Clear Register (DMA_CLEARBLOCK)

Offset address: 0x340

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CLEARBLOCK	W	0x0	Writing 1 to bit x clears the block transfer interrupt for channel x.

Note: Reading this register is prohibited.

11.4.24 Source Transfer Interrupt Clear Register (DMA_CLEARSRCTRAN)

Offset address: 0x348

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CLEARSRCTRAN	W	0x0	Writing 1 to bit x clears the source transfer interrupt for channel x.

Note: Reading this register is prohibited.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11.4.25 Destination Transfer Interrupt Clear Register (DMA_CLEARDSTTRAN)

Offset address: 0x350

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CLEARDSTTRAN	W	0x0	Writing 1 to bit x clears the destination transfer interrupt for channel x.

Note: Reading this register is prohibited.

11.4.26 Error Interrupt Clear Register (DMA_CLEARERR)

Offset address: 0x358

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CLEARERR	W	0x0	Writing 1 to bit x clears the error interrupt for channel x.

Note: Reading this register is prohibited.

11.4.27 Interrupt Status Register (DMA_STATUSINT)

Offset address: 0x360

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	ERR	R	0x0	Each bit or calculation result of STATUSERR
3	DSTTRAN	R	0x0	Each bit or calculation result of STATUSDSTTRAN

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	SRCTRAN	R	0x0	Each bit or calculation result of STATUSSRCTRAN
1	BLOCK	R	0x0	Each bit or calculation result of STATUSBLOCK
0	TFR	R	0x0	Each bit or calculation result of STATUSTFR

Note: Writing to this register is prohibited.

11.4.28 Destination Transfer Request Signal Register (DMA_REQSRCREG)

Offset address: 0x368

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	SRC_REQ_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	SRC_REQ	R/W	0x0	Bit x is the request handshake signal for source transfer of channel x.

11.4.29 Destination Transfer Request Signal Register (DMA_REQDSTREG)

Offset address: 0x370

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	DST_REQ_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	DST_REQ	R/W	0x0	Bit x is the request handshake signal for destination transfer of channel x.

11.4.30 Source Transfer Single Signal Register (DMA_SGLREQSRCREG)

Offset address: 0x378

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	SRC_SGLREQ_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	SRC_SGLREQ	R/W	0x0	Bit x is the single handshake signal for source transfer of channel x.

11.4.31 Destination Transfer Single Signal Register (DMA_SGLREQDSTREG)

Offset address: 0x380

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:8	DST_SGLREQ_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	DST_SGLREQ	R/W	0x0	Bit x is the single handshake signal for destination transfer of channel x.

11.4.32 Source Transfer Last Signal Register (DMA_LSTSRCRE)

Offset address: 0x388

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	LSTSRC_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	LSTSRC	R/W	0x0	Bit x is the last handshake signal for source transfer of channel x.

11.4.33 Destination Transfer Last Signal Register (DMA_LSTDSTREG)

Offset address: 0x390

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:8	LSTDST_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0
7:0	LSTDST	R/W	0x0	Bit x is the last handshake signal for destination transfer of channel x.

11.4.34 DMA Module Enable Register (DMA_CFGREG)

Offset address: 0x398

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	DMA_EN	R/W	0x0	DMA enable: 0x0: DMA module function disabled 0x1: DMA module function enabled; this bit shall be enabled before enabling the channel.

11.4.35 Channel Enable Register (DMA_CHENREG)

Offset address: 0x3A0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	CH_EN_WE	W	0x0	Bit (x + 8) is the write activation bit for channel x: 0x0: Write disabled 0x1: Write enabled Valid for once, read as 0

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	CH_EN	R/W	0x0	Bit x is the enable bit for channel x. This bit will be automatically set to 0 upon completion of transfer.

For example:

- To set bit 0 to 1, write 0x101 to this register without affecting bit 1.
- To set bit 1 to 0, write 0x200 to this register without affecting bit 0.
- To set bit 0 and bit 1 to 1, write 0x303 to this register.

11.5 Operation Procedure

11.5.1 Basic Hardware Flow Control

1. Configure the DMA_SARx register to specify the source address.
2. Configure the DMA_DARx register to specify the destination address.
3. Configure the DMA_CTLx register:
   A. to enable interrupt if required.
   B. to set the transfer data width and size.
   C. to select the flow control type; note that SRAM has no handshake signal, and SRAM is identified as memory in TT_FC.
   D. to select whether the address is incremented or decremented.
   E. to select the burst transfer length.
4. Configure the DMA_CFGx register:
   A. to select the channel priority.
   B. to set whether to automatically restart the transmission.
   C. to select the handshake signal.
   D. to set the HS_SEL field to 0.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. Configure the DMA_MASKBLOCK register to enable interrupt for corresponding channel if required.
6. Configure the DMA_CFGREG register to enable DMA.
7. Configure the DMA_CHENREG register to enable the channel.
8. Wait for the interrupt or query DMA_CHENREG.
9. Clear the interrupt.

11.5.2 Software Flow Control

1. Configure the DMA_SARx register to specify the source address.
2. Configure the DMA_DARx register to specify the destination address.
3. Configure the DMA_CTLx register:
   A. to enable interrupt if required.
   B. to set the transfer data width and size.
   C. to select the flow control type.
   D. to select whether the address is incremented or decremented.
   E. to select the burst transfer length.
4. Configure the DMA_CFGx register:
   A. to select the channel priority.
   B. to set the HS_SEL field to 1.
5. Configure the DMA_MASKBLOCK register to enable interrupt for corresponding channel if required.
6. Configure the DMA_CFGREG register to enable DMA.
7. Configure the DMA_CHENREG register to enable the channel.
8. Software writes to the software handshake register to trigger the transmission.
9. Wait for the interrupt or query DMA_CHENREG.
10. Clear the interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12 Digital Camera Interface (DCMI)

12.1 Overview

The DCMI can receive 8/10/12/14-bit data streams from the camera and supports multiple formats and JPEG transmission.

12.2 Main Features

- 8/10/12/14-bit DVP interface
- Hardware / embedded code synchronization
- Continuous / single frame mode
- Window cropping function
- Built-in DMA data transfer

12.3 Pin Description

Table 12-1: DCMI Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
DCMI_PIXCLK	PA6	Input	Digital camera clock signal input
DCMI_HSYNC	PA4	Input	Digital camera line synchronization signal input
DCMI_VSYNC	PB7	Input	Digital camera column synchronization signal input
DCMI_D0	PA9, PC6	Input	Digital camera data signal input
DCMI_D1	PA10, PC7	Input	Digital camera data signal input
DCMI_D2	PC8, PE0	Input	Digital camera data signal input
DCMI_D3	PC9, PE1	Input	Digital camera data signal input
DCMI_D4	PC11, PE4	Input	Digital camera data signal input
DCMI_D5	PB6	Input	Digital camera data signal input

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
DCMI_D6	PB8, PE5	Input	Digital camera data signal input
DCMI_D7	PB9, PE6	Input	Digital camera data signal input
DCMI_D8	PC10	Input	Digital camera data signal input
DCMI_D9	PC12	Input	Digital camera data signal input
DCMI_D10	PB5	Input	Digital camera data signal input
DCMI_D11	PD2	Input	Digital camera data signal input
DCMI_D12	PD0	Input	Digital camera data signal input
DCMI_D13	PD1	Input	Digital camera data signal input

12.4 Functional Description

12.4.1 Data Capture

12.4.1.1 8-bit Data

When the capture data width is set to 8 bits, every 4 captured data will fill one word of memory. When one word of data is captured or the end of line (non-JPEG mode) or end of frame is detected, the one word of data will be written to FIFO. The data storage format is shown in the table below:

Table 12-2: 8-bit Data Storage Format

Bit	31:24	23:16	15:8	7:0
Data	Data 4	Data 3	Data 2	Data 1

12.4.1.2 10-bit Data

When the capture data width is set to 10 bits, every 2 captured data will fill one word of memory. When one word of data is captured or the end of line (non-JPEG mode) or end of frame is detected, the one word of data will be written to FIFO. The data storage format is shown in the table below:

Unicmicro

Table 12-3: 10-bit Data Storage Format

Bit	31:26	25: 16	15:10	9:0
Data	6'b0	Data 2	6'b0	Data 1

12.4.1.3 12-bit Data

When the capture data width is set to 12 bits, every 2 captured data will fill one word of memory. When one word of data is captured or the end of line (non-JPEG mode) or end of frame is detected, the one word of data will be written to FIFO. The data storage format is shown in the table below:

Table 12-4: 12-bit Data Storage Format

Bit	31:28	27: 16	15:12	11:0
Data	4'b0	Data 2	4'b0	Data 1

12.4.1.4 14-bit Data

When the capture data width is set to 14 bits, every 2 captured data will fill one word of memory. When one word of data is captured or the end of line (non-JPEG mode) or end of frame is detected, the one word of data will be written to FIFO. The data storage format is shown in the table below:

Table 12-5: 14-bit Data Storage Format

Bit	31:30	29: 16	15:14	13:0
Data	2'b0	Data 2	2'b0	Data 1

12.4.2 Hardware Synchronization Mode

In hardware synchronization mode, DCMI uses VSYNC and HSYNC signals to control signal acquisition. When both VSYNC and HSYNC are invalid, DCMI will receive the data transmitted, with valid HSYNC indicating the completion of one-line transmission and valid VSYNC indicating the completion of one-frame transmission. The polarity of VSYNC, HSYNC and pixel clock is programmable.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.4.3 JPEG Mode

When the JPEG mode is enabled, the functions of window cropping and embedded code synchronization are disabled, and HSYNC is used only for data enable and will not trigger the end-of-line write operation into the FIFO.

12.4.4 Embedded Code Synchronization Mode

With the embedded code synchronization mode on, the specific values contained in the data stream are used instead of the VSYNC and HSYNC signals. This mode is available only when the 8-bit data input is selected.

Set the feature code for detection in DCMI_ESC register and DCMI_ESCNASK register, then synchronization can be triggered when FF 00 00 xx appears in the data stream with XX the same as the feature code to start or stop the data acquisition.

The DCMI_ESCMASK register can be set to mask certain bits in the feature code for detection, and synchronization can be triggered as long as the unmasked bits in the data match the values set in the DCMI_ESC register. For example, if FS in DCMI_ESC is set to AA and FSM in DCMI_ESCMASK is set to F0, the frame synchronization can be triggered as long as FF 00 00 AX (X = 0–F) appears in the data stream.

12.4.5 Window Cropping

The image acquisition range can be limited within a window area by setting the DCMI_CROPSTART register and DCMI_CROPSIZE register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.4.6 Single-frame Capture

By using the single-frame capture mode, DCMI can automatically turn off capture once a frame has been captured.

12.5 Register

Register base address: 0x4020_0000

The registers are listed below:

Table 12-6: List of DCMI Registers

Offset Address	Name	Description
0x00	DCMI_CTRL	Control register
0x04	DCMI_STATUS	Status register
0x08	DCMI_INTRAW	Raw interrupt register
0x0C	DCMI_INTEN	Interrupt enable register
0x10	DCMI_INTMASKED	Interrupt status register
0x14	DCMI_INTCLR	Interrupt clear register
0x18	DCMI_ESC	Embedded code register
0x1C	DCMI_ESCMASK	Embedded code mask register
0x20	DCMI_CROPSTART	Cropped window starting point register
0x24	DCMI_CROPSIZE	Cropped window size register
0x28	DCMI_DATA	Data register
0x2C	DCMI_MSTCTRL	Data output control register
0x30	DCMI_MSTADR	Data output address register

12.5.1 Control Register (DCMI_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	INPUT_OFFSET	R/W	0x0	Select the input source when the input data width is 8 bits:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Input signal from D0–D7 ports 1: Input signal from D2–D9 ports
14	DCMI_EN	R/W	0x0	DCMI enable: 0: Disabled 1: Enabled
13:12	RSV	-	-	Reserved
11:10	WIDTH	R/W	0x0	Data input width selection: 0: 8 bits 1: 10 bits 2: 12 bits 3: 14 bits
9:8	FRAME_RATE	R/W	0x0	Frame rate control: 0: Capture all frames 1: Capture half the frames 2/3: Capture 1/4 the frames
7	VSYNC_POL	R/W	0x0	Select the vertical synchronization VSYNC polarity (this bit indicates the level of VSYNC pin when the data is invalid on the parallel interface): 0: Blanking at low level (VSYNC is active low, no data is transmitted) 1: Blanking at high level (VSYNC is active high, no data is transmitted)
6	HSYNC_POL	R/W	0x0	Select the horizontal synchronization HSYNC polarity (this bit indicates the level of HSYNC pin when the data is invalid on the parallel interface): 0: Blanking at low level (HSYNC is active low, no data is transmitted) 1: Blanking at high level (HSYNC is active high, no data is transmitted)
5	CLK_POL	R/W	0x0	Select the clock polarity (this bit configures the capture edge of the pixel clock): 0: Active at falling edge

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Active at rising edge
4	ESC_EN	R/W	0x0	Select hardware or embedded code synchronization: 0: Hardware synchronization 1: Embedded code synchronization
3	JPEG_EN	R/W	0x0	JPEG mode enable: 0: JPEG mode disabled 1: JPEG mode enabled, disabling the functions of frame rate reduction, embedded code synchronization and window cropping
2	CROP_EN	R/W	0x0	Window cropping enable: 0: Disabled 1: Enabled
1	SINGLE_FRAME	R/W	0x0	Single-frame capture enable: 0: Single-frame capture disabled 1: Single-frame capture enabled, and capture is automatically turned off once a frame has been captured
0	CAPTURE_EN	R/W	0x0	Capture enable: 0: Disabled 1: Enabled

12.5.2 Status Register (DCMI_STATUS)

Offset address: 0x04

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2	FIFO_EMPTY	R	0x0	FIFIO status: 0: Empty 1: Not empty
1	VSYNC_STATUS	R	0x1	VSYNC status: 0: Transmitting data 1: Blanking

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	HSYNC_STATUS	R	0x1	HSYNC status: 0: Transmitting data 1: Blanking

12.5.3 Raw Interrupt Register (DCMI_INTRAW)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	MST_ERR	R	0x0	Set when transmission error occurs in output data
4	HSYNC	R	0x0	Set when HSYNC is blanked
3	VSYNC	R	0x0	Set when VSYNC is blanked
2	ESC_ERR	R	0x0	Set when an error embedded code is received
1	OVERFLOW	R	0x0	Set when a FIFO overflow occurs
0	FRAME_END	R	0x0	Set when one frame is captured

12.5.4 Interrupt Enable Register (DCMI_INTEN)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	MST_ERR_EN	R/W	0x0	Enable data transmission error interrupt
4	HSYNC_EN	R/W	0x0	Enable HSYNC interrupt
3	VSYNC_EN	R/W	0x0	Enable VSYNC interrupt
2	ESC_ERR_EN	R/W	0x0	Enable embedded code error interrupt
1	OVERFLOW_EN	R/W	0x0	Enable FIFO overflow interrupt
0	FRAME_END_EN	R/W	0x0	Enable frame capture completion interrupt

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.5.5 Interrupt Status Register (DCMI_INTMASKED)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	MST_ERR_MASKED	R	0x0	Data transmission error interrupt status
4	HSYNC_MASKED	R	0x0	HSYNC interrupt status
3	VSYNC_MASKED	R	0x0	VSYNC interrupt status
2	ESC_ERR_MASKED	R	0x0	Embedded code error interrupt status
1	OVERFLOW_MASKED	R	0x0	FIFO overflow interrupt status
0	FRAME_END_MASKED	R	0x0	Frame capture completion interrupt status

12.5.6 Interrupt Clear Register (DCMI_INTCLR)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	MST_ERR_CLR	W	0x0	Writing 1 to this bit clears the transmission error interrupt.
4	HSYNC_CLR	W	0x0	Writing 1 to this bit clears the HSYNC interrupt.
3	VSYNC_CLR	W	0x0	Writing 1 to this bit clears the VSYNC interrupt.
2	ESC_ERR_CLR	W	0x0	Writing 1 to this bit clears the embedded code error interrupt.
1	OVERFLOW_CLR	W	0x0	Writing 1 to this bit clears the FIFO overflow interrupt.
0	FRAME_END_CLR	W	0x0	Writing 1 to this bit clears the frame capture completion interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.5.7 Embedded Code Register (DCMI_ESC)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	FE	R/W	0x0	Set the embedded code for detecting the end of a frame.
23:16	LE	R/W	0x0	Set the embedded code for detecting the end of a line.
15:8	LS	R/W	0x0	Set the embedded code for detecting the start of a line.
7:0	FS	R/W	0x0	Set the embedded code for detecting the start of a frame.

12.5.8 Embedded Code Mask Register (DCMI_ESCMASK)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	FEM	R/W	0x0	Set the embedded code detection mask bit for detecting the end of a frame: Bits set to 0 will not be detected. Bits set to 1 will be detected.
23:16	LEM	R/W	0x0	Set the embedded code detection mask bit for detecting the end of a line: Bits set to 0 will not be detected. Bits set to 1 will be detected.
15:8	LSM	R/W	0x0	Set the embedded code detection mask bit for detecting the start of a line: Bits set to 0 will not be detected. Bits set to 1 will be detected.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	FSM	R/W	0x0	Set the embedded code detection mask bit for detecting the start of a frame: Bits set to 0 will not be detected. Bits set to 1 will be detected.

12.5.9 Cropping Window Starting Point Register (DCMI_CROPSTART)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	Reserved
28:16	VERTICAL_STARTPOINT	R/W	0x0	Set the number of lines to skip before starting capture.
15:14	RSV	-	-	Reserved
13:0	HORIZONAL_STARTPOINT	R/W	0x0	Set the number of pixel clocks to skip before each line starts capturing.

12.5.10 Cropping Window Size Register (DCMI_CROPSIZE)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	Reserved
28:16	VERTICAL_SIZE	R/W	0x0	Set the number of lines captured per frame.
15:14	RSV	-	-	Reserved
13:0	HORIZONAL_SIZE	R/W	0x0	Set the number of pixel clocks captured per line.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

12.5.11 Data Register (DCMI_DATA)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DATA	R	0x0	Read this register to fetch data from FIFO.

12.5.12 Data Output Control Register (DCMI_MSTCTRL)

Offset address: 0x2C

Reset value: 0x0000 0016

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5	ADRRST	R/W	0x0	Set whether to reset the transmission address upon completion of a frame transmission: 0: Do not reset 1: The transmission address is automatically reset to the original set address after a frame is transmitted.
4:3	WIDTH	R/W	0x2	Set the data transmission width: 0: 8 bits 1: 16 bits 2/3: 32 bits
2:1	ADRINCR	R/W	0x3	Set whether to enable address auto-increment: 0/1: Disabled, data is output to the same address 2: Enabled, data is output to the decremented address 3: Enabled, data is output to the incremented address

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	MST_EN	R/W	0x0	Set whether to enable automatic data output: 0: Disabled 1: Enabled

12.5.13 Data Output Address Register (DCMI_MSTADR)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MST_ADR	R/W	0x0	Set the destination address of data output, note that it shall be aligned with the data transmission width.

12.6 Operation Procedure

12.6.1 Capturing Data Using DCMI and Saving Data to Memory

1. Connect the external camera.
2. Configure the corresponding IO pins as I2C_SCL and I2C_SDA, and check the I2C bus status to ensure it is idle.
3. Configure the corresponding IO pins as camera_pdn and camera_rst respectively.
4. Configure the corresponding IO pins as DCMI_PCLK, DCMI_VSYNC, DCMI_HSYNC and DCMI_D0−D13 respectively to alternate DCMI functions.
5. camera pdn = 0 indicates the low-power mode is turned off, which can be reset by setting the camera RST hardware. Note: It shall be configured according to datasheet of the selected camera, taking ov2640 as an example, the same below.
6. Soft reset the camera via I2C. Read the camera ID.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Configure the camera to UXGA, RGB565 mode via I2C bus. Refer to the sample code for specific parameter configuration.
8. Configure via I2C bus the camera screen selection size of 320 (width) * 240 (height).
9. Configure the camera clock frequency via I2C bus.
10. Enable DCMI clock and release DCMI reset.
11. Enable dcmi_frameend interrupt and DCMI hardware interrupt. When dcmi_frameend interrupt is triggered, it means that a frame of data collected by DCMI from external camera has been completely saved to the destination memory area.
12. Configure DCMI RX data width as 8 bits and configure the input signal as D0–D7. Note: This step shall be carried out referring to the hardware design and data transmission mode.
13. Configure DCMI VSYNC to be low, DCMI HSYNC to be low and DCMI PCLK to be high, and enable hardware synchronization.
14. Set the cropping window starting point (DCMI_CROPSTART) and size (DCMI_CROPSIZE) which shall be less than or equal to the camera screen selection size.
15. Enable window cropping crop_en = 1.
16. Configure the frame rate as capturing all frames, enable continuous capture, and disable JPEG reception.
17. Set the DCMI data output destination address mst_adr to be the legal address of the destination memory area.
18. Set the address to be reset automatically after one frame transmission; the DCMI output data transfer width is 8 bits; set the output address to be automatically incremented by 1. Note: The data width shall be configured according to the actual application.
19. Enable automatic data output: mst_en = 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20. Enable DCMI and DCMI capture, so that DCMI stores the captured data to the designated memory area.

12.6.2 Capturing Data Using DCMI and Transferring It Directly to Destination Address

1. Connect the external camera.
2. Configure the corresponding IO pins as I2C_SCL and I2C_SDA, and check the I2C bus status to ensure it is idle.
3. Configure the corresponding IO pins as camera_pdn and camera_rst respectively.
4. Configure the corresponding IO pins as DCMI_PCLK, DCMI_VSYNC, DCMI_HSYNC and DCMI_D0–D13 respectively to alternate DCMI functions.
5. camera pdn = 0 indicates the low-power mode is turned off, which can be reset by setting the camera RST hardware. Note: It shall be configured according to datasheet of the selected camera, taking ov2640 as an example, the same below.
6. Soft reset the camera via I2C. Read the camera ID.
7. Configure the camera to UXGA, RGB565 mode via I2C bus. Refer to the sample code for specific parameter configuration.
8. Configure via I2C bus the camera screen selection size of 320 (width) * 240 (height).
9. Configure the camera clock frequency via I2C bus.
10. Enable DCMI clock and release DCMI reset.
11. Configure DCMI RX data width as 8 bits and configure the input signal as D0–D7. Note: This step shall be carried out referring to the hardware design and data transmission mode.
12. Configure DCMI VSYNC to be low, DCMI HSYNC to be low and DCMI PCLK to be high, and

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

enable hardware synchronization.

13. Set the cropping window starting point (DCMI_CROPSTART) and size (DCMI_CROPSIZE) which shall be less than or equal to the camera screen selection size.
14. Enable window cropping crop_en = 1.
15. Configure the frame rate as capturing all frames, enable continuous capture, and disable JPEG reception.
16. Set the DCMI data output destination address mst_adr as the destination peripheral address, e.g. EMC I80 data input register address.
17. The DCMI output data transfer width is 16 bits; set the output address to be unchanged. Note: The data width shall be configured according to the actual application.
18. Enable automatic data output: mst_en = 1.
19. Enable DCMI and DCMI capture, so that DCMI transfers the captured data to the designated peripheral address.

12.6.3 Capturing Data Using DCMI and Transferring It to Destination Address via DMA

1. Connect the external camera.
2. Configure the corresponding IO pins as I2C_SCL and I2C_SDA, and check the I2C bus status to ensure it is idle.
3. Configure the corresponding IO pins as camera_pdn and camera_rst respectively.
4. Configure the corresponding IO pins as DCMI_PCLK, DCMI_VSYNC, DCMI_HSYNC and DCMI_D0−D13 respectively to alternate DCMI functions.
5. camera pdn = 0 indicates the low-power mode is turned off, which can be reset by setting

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the camera RST hardware. Note: It shall be configured according to datasheet of the selected camera, taking ov2640 as an example, the same below.

6. Soft reset the camera via I2C. Read the camera ID.
7. Configure the camera to UXGA, RGB565 mode via I2C bus. Refer to the sample code for specific parameter configuration.
8. Configure via I2C bus the camera screen selection size of 320 (width) * 240 (height).
9. Configure the camera clock frequency via I2C bus.
10. Enable DCMI clock and release DCMI reset.
11. Configure DCMI RX data width as 8 bits and configure the input signal as D0–D7. Note: This step shall be carried out referring to the hardware design and data transmission mode.
12. Configure DCMI VSYNC to be low, DCMI HSYNC to be low and DCMI PCLK to be high, and enable hardware synchronization.
13. Set the cropping window starting point (DCMI_CROPSTART) and size (DCMI_CROPSIZE) which shall be less than or equal to the camera screen selection size.
14. Enable window cropping crop_en = 1.
15. Configure the frame rate as capturing all frames, enable continuous capture, and disable JPEG reception.
16. Set the DCMI data output destination address mst_adr as the destination peripheral address, e.g. EMC I80 data input register address.
17. The DCMI output data transfer width is 16 bits; set the output address to be unchanged. Note: The data width shall be configured according to the actual application.
18. Disable automatic data output: mst_en = 0. In this mode, the output address is reset, and the configuration of output data width and output address is invalid. By default, DMA mode

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

adopts 32-bit data width transmission internally.

19. Enable DCMI and DCMI capture.

20. Enable DMA1 clock and release DMA1 reset. Example: DMA carries DCMI data to EMC.

21. Initialize DMA and channel configuration, for example, configure both the source address and destination address to be unchanged for DMA1 channel 1 transfer from peripheral to memory. Both the source and destination DMA burst lengths are 1. The source data transfer width is 32 bits, and the destination data transfer width is 16 bits. Both source and destination transfers are hardware synchronized. The source handshake signal is signal 1, and the destination handshake signal is an arbitrary value (memory).

22. Enable DMA module. Enable DMA1 interrupt and hook the interrupt handler function.

23. Enable DMA1 to transfer data, in this case: For DMA1 channel 1, the source address is set by DCMI_DATA register, and the destination register is set by EMC_DATA register. Configure the transfer data length as 1/4 the total bytes of a row of pixels. See DMA chapter introduction or sample code for details of DMA configuration.

24. Enter DMA interrupt after each line of data transmission to perform line counting and start the next line of DMA data transmission. When the transmission of a frame is completed, the line count is cleared, at this point the captured data of a frame has been completely transferred to the destination peripheral, and the first DMA transfer of a new frame of data is started.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

13 External Memory Controller (EMC)

13.1 Overview

The module supports SRAM, NOR FLASH and I80 interface functions. The EMC is responsible for controlling the transmission of information between the internal local bus and the external memory module (synchronous or asynchronous). Also this module can be used as TFT-LCD controller to support TFT-LCD with 8080 interface.

13.2 Main Features

- Off-chip SRAM and NOR Flash extensions
- 8080 TFT-LCD control
- Write protection
- Configurable 8-/16-bit external interface
- Configurable waiting time for write and read operations

13.3 Pin Description

Table 13-1: EMC Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
EX_NCE (csn0)	PD7	Output	Chip select signal
EX_RS (rs)	PD6	Output	Data and command selection
EX_WR (wen)	PD5	Output	Write enable
EX_RD (oen)	PD4	Output	Read enable
EX_BA[1] (EBn[1])	PE1	Output	Valid bit control for higher 8 bits of data

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
EX_BA[0] (EBn[0])	PE0	Output	Valid bit control for lower 8 bits of data
EX_RST	-	Output	Control with a GPIO
EX_D0	PD14	Input/output	Data input/output
EX_D1	PD15	Input/output	Data input/output
EX_D2	PD0	Input/output	Data input/output
EX_D3	PD1	Input/output	Data input/output
EX_D4	PE7	Input/output	Data input/output
EX_D5	PE8	Input/output	Data input/output
EX_D6	PE9	Input/output	Data input/output
EX_D7	PE10	Input/output	Data input/output
EX_D8	PE11	Input/output	Data input/output
EX_D9	PE12	Input/output	Data input/output
EX_D10	PE13	Input/output	Data input/output
EX_D11	PE14	Input/output	Data input/output
EX_D12	PE15	Input/output	Data input/output
EX_D13	PD8	Input/output	Data input/output
EX_D14	PD9	Input/output	Data input/output
EX_D15	PD10	Input/output	Data input/output
MEM_A0	PB10	Output	Address line
MEM_A1	PB11	Output	Address line
MEM_A2	PB12	Output	Address line
MEM_A3	PB13	Output	Address line
MEM_A4	PC0	Output	Address line
MEM_A5	PC1	Output	Address line
MEM_A6	PC2	Output	Address line
MEM_A7	PC3	Output	Address line
MEM_A8	PC4	Output	Address line
MEM_A9	PC5	Output	Address line
MEM_A10	PE2	Output	Address line
MEM_A11	PE3	Output	Address line
MEM_A12	PE4	Output	Address line
MEM_A13	PE5	Output	Address line
MEM_A14	PE6	Output	Address line
MEM_A15	PD3	Output	Address line
MEM_A16	PD11	Output	Address line

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
MEM_A17	PD12	Output	Address line
MEM_A18	PD13	Output	Address line

13.4 Functional Description

13.4.1 Chip Select

The CSen signal in the EMC_CSCR0 register corresponds to a peripheral chip select enable. Setting CSen in the register does not mean that the external chip select signal CSnx is active low, which is effective only when CSen in the register is 1 and a read/write operation is performed on the corresponding peripheral address area.

13.4.2 EBn Signal Timing

For CSnx write operation, the setup time and cancellation time of EBn signal can be flexibly configured. However, for read operation, only the setup time is configured. The cancellation time of read operation is always at the end of the rising edge.

The timing diagram of write operation is shown in the following figure:

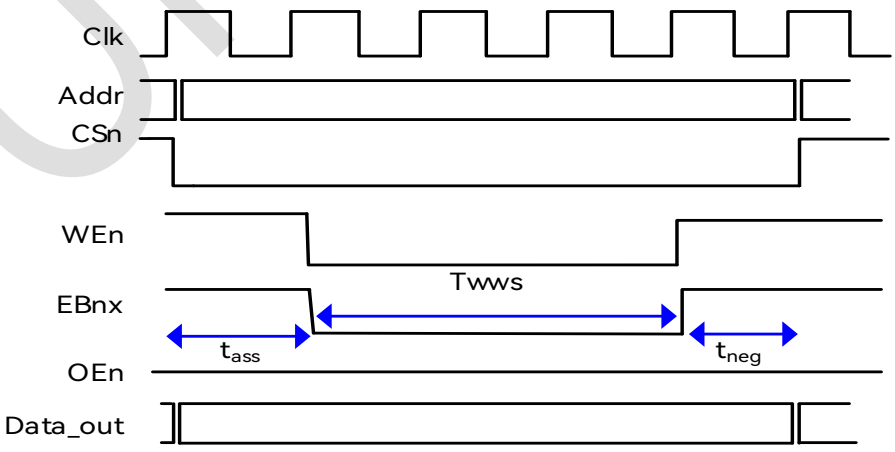

Figure 13-1: Write Operation Timing Diagram for Off-chip Memory Controller

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The timing diagram of read operation is shown in the following figure:

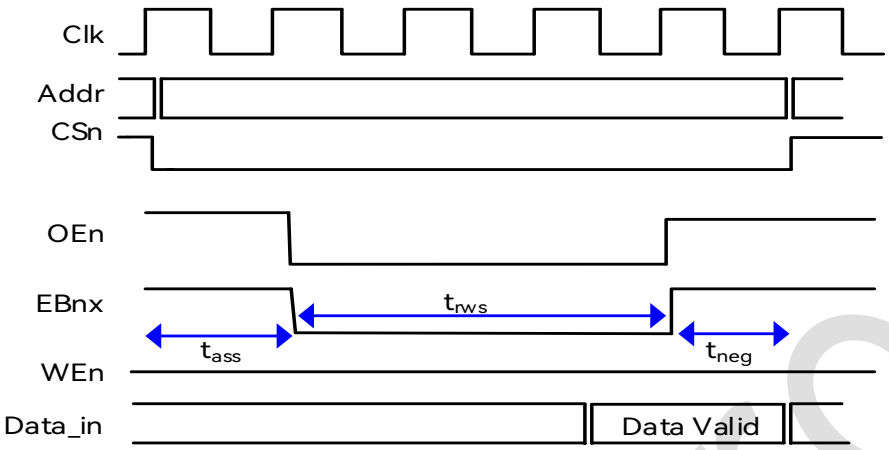

Figure 13-2: Read Operation Timing Diagram for Off-chip Memory Controller

The corresponding time of read/write timing of off-chip memory is shown in the table below:

Table 13-2: Corresponding Time of Read/Write Timing of Off-chip Memory

Wats/Rats/Aats	**t_{ass}**	**Wnts**	**t_{neg}**	**Wws/Rws**	**t_{wws}/t_{rws}**
000	0	000	0	0000	1
001	1	001	1	0001	2
010	2	010	2	0010	3
011	3	011	3	0011	4
100	4	100	4	0100	5
101	5	101	5	0101	6
110	6	110	6	0110	7
111	7	111	7	0111	8
-	-	-	-	1000	9
				1001	10
				1010	11
				1011	12
				1100	13
				1101	14
				1110	15
				1111	16

Note: For TFT reading, t_{neg} is valid, that is, t_{neg} time is included in the timing.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

13.4.3 Byte Enable Pin

PS in the EMC_CSCR0 register is set according to the data bit width of the externally connected peripheral. The EMC controller supports peripherals with external data of 8/16 bits, which can be accessed by CPU in a bit width higher than that of the peripheral data. For example, if the peripheral data width is 8 bits, when the CPU initiates a 32-bit write command, the EMC controller will internally divide the 32-bit command into four independent write operations. An error will occur when the CPU initiates an access below the minimum specified data width of the peripheral, and the software shall avoid initiating such an operation.

The following table shows the correspondence between EBn and peripheral data lines.

Table 13-3: Correspondence between EBn and Peripheral Data Line

Minimum Access Data Bit Width Supported by Peripheral (Unit: Bit)	Peripheral Data Bit Width (Unit: Bit)	EBnx	External Data Interface Signal
8	16	EBnx[1]	Data_in / Data_out[15:8]
		EBnx[0]	Data_in / Data_out[7:0]
16	16	EBnx[1]	-
		EBnx[0]	Data_in / Data_out[15:0]
8	8	EBnx[1]	-
		EBnx[0]	Data_in / Data_out[7:0]

13.4.4 TFT_LCD Control

The EMC module supports 8080 TFT-LCD control. The 8080 TFT-LCD timing interface is shown in the following figure:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

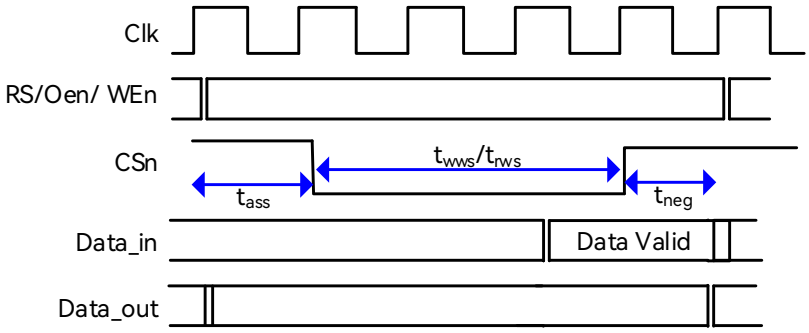

Figure 13-3: I8080 TFT_LCD Timing Interface

RS in EMC module is used to select command operation or data transfer operation, depending on whether the register for this operation is EMC_CMD or EMC_DATA.

Note: For TFT reading, t_{neg} is valid, that is, t_{neg} time is included in the read timing.

13.5 Register Description

EMC register base address: 0x6A00_0000

The registers are listed below:

Table 13-4: List of EMC Registers

Offset Address	Name	Description
0x00	EMC_CSCR0	Chip select control register 0
0x10	EMC_TFTS	TFT selection interface register
0x14	EMC_CMD	CMD channel register in TFT mode
0x18	EMC_DATA	DATA channel register in TFT mode
0x1C	EMC_SEG0	Segment register 0

Segment start address: 0x6800_0000

Table 13-5: Chip Select Segment Start Address

Offset Address	Name	Description
0x800000*x	UM_SEG_ADDRx	Start address of the chip select segment x, which takes the value of 0–3.

Registers are detailed in the following sections.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

13.5.1 Chip Select Control Register (EMC_CSCR0)

Offset address: 0x00

Reset value: 0x7770 2FF0

Bit	Name	Attribute	Reset Value	Description
31	RSV	-	-	Reserved
30:28	WATS	R/W	0x7	Write signal valid time selection: Time interval between CSn signal being valid and EBn signal being valid
27	RSV	-	-	Reserved
26:24	WNTS	R/W	0x7	Write signal invalid time selection: Time interval between EBn signal being invalid and CSn signal being invalid
23	RSV	-	-	Reserved
22:20	RATS	R/W	0x7	Read signal valid time selection: Time interval between CSn signal being valid and EBn/OEn signal being valid
19:15	RSV	-	-	Reserved
14	RO	R/W	0	RO bit is used to restrict write access to the address field under the corresponding chip select: 1: Only read access is allowed. 0: Both read and write accesses are allowed. When accessing a chip-selected memory space, the chip select logic will compare the RO bit with the internal read/write signal. If a violation is detected by the chip select logic, the access will be ignored. 1: Only read access is allowed; the chip select logic will ignore the write access. 0: Both read and write accesses are allowed. This bit does not work in TFT mode.
13	PS	R/W	0x1	External port bit width selection: 0: 16-bit port 1: 8-bit port
12	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
11:8	WWS	R/W	0xF	WWS filed determines the number of clock cycles for the write wait state.
7:4	RWS	R/W	0xF	RWS filed determines the number of clock cycles for the read wait state.
3:1	RSV	-	-	Reserved
0	CSEN	R/W	0x0	The CSen bit is used to enable chip select logic: If the chip select is disabled, the external chip select signal will be active high. 1: Chip select enabled 0: Chip select disabled

13.5.2 TFT Mode Channel Register (EMC_TTS)

Offset address: 0x10

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1:0	TFTS	R/W	3	TFT mode selection: 00: Peripheral controlled by CSn0 is in TFT mode, used for 8080 TFT-LCD control 01: Peripheral controlled by CSn0 is in SRAM mode, used for external extension of SRAM 10: Peripheral controlled by CSn0 is in NORFlash mode, used for external extension of NOR Flash

13.5.3 TFT Command Channel Register (EMC_CMD)

Offset address: 0x14

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CMD	W	0x0	If the peripheral is selected in TFT mode, reading from or writing to this register will interpret the data as the content of a command (Cmd), initiating a TFT command operation.

13.5.4 TFT Data Channel Register (EMC_DATA)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Default Value	Description
31:0	DATA	R/W	0x0	If the peripheral is selected in TFT mode, reading from or writing to this register will interpret the data as the content of data, initiating a TFT Data operation.

13.5.5 Segment Register (EMC_SEG0)

Offset address: 0x1C

Reset value: 0x0FFF 8000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	Reserved
27:15	SIZE	R/W	0x1FFF	Size setting bit of the memory area (SEG0) corresponding to CSen0: SEG0 (capacity: byte) = (Size + 1) * 1K
14:0	START	R/W	0	Start address of the memory area (SEG0) corresponding to CSen0, aligned in 1 KB: that is, the address here is in unit of 1 KB, and the smallest unit of each SEG is 1 KB.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

13.6 Operation Procedure

13.6.1 Off-chip Memory Read Operation

1. Configure the IO pins as EMC CSn, WEn, RS, OEn, D0−D15, A0−A18, DQM0 and DQM1 to alternate EMC functions.
2. Enable EMC clock and release EMC reset.
3. Configure EMC CSN0 to non-TFT mode.
4. Set bit[30:28], bit[26:24], bit[22:20], bit[11:8] and bit[7:4] in the EMC_CSCR0 register, the setting value can refer to the default value or be configured according to the peripheral parameters used.
5. Set EMC_CSCR0[14] to read-only/read-write allowed. The external port bit width is set to 16 bits or 8 bits according to the peripheral parameters.
6. Obtain the start address UM_SEG_ADDR0 of the chip-selected memory area.
7. Set the segment register EMC_SEG0.
8. Enable CSN0.
9. Initiate a read operation on the corresponding legal chip-selected memory area.

13.6.2 Off-chip Memory Write Operation

1. Configure the IO pins as EMC CSn, WEn, RS, OEn, D0−D15, A0−A18, DQM0 and DQM1 to alternate EMC functions.
2. Enable EMC clock and release EMC reset.
3. Configure EMC CSN0 to non-TFT mode.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Set bit[30:28], bit[26:24], bit[22:20], bit[11:8] and bit[7:4] in the EMC_CSCR0 register, the setting value can refer to the default value or be configured according to the peripheral parameters used.
5. Set EMC_CSCR0[14] to read-write allowed. The external port bit width is set to 16 bits or 8 bits according to the peripheral parameters.
6. Obtain the start address UM_SEG_ADDR0 of the chip-selected memory area.
7. Set the segment register EMC_SEG0.
8. Enable CSN0.
9. Initiate a write operation on the corresponding legal chip-selected memory area.

13.6.3 TFT_LCD Operation

1. Configure the IO pins as EMC CSn, WEn, RS, OEn and D0–D15 respectively to alternate EMC functions.
2. Enable EMC clock and release EMC reset.
3. Configure EMC CSN0 to TFT mode.
4. Set bit[30:28], bit[26:24], bit[22:20], bit[11:8] and bit[7:4] in the EMC_CSCR0 register, the setting value can refer to the default value or be configured according to the peripheral parameters used.
5. Set EMC_CSCR0[14] to read-write allowed. The external port bit width is set to 16 bits or 8 bits according to the peripheral parameters.
6. Enable CSN0.
7. Write data to EMC_CMD to initiate peripheral command transfer.
8. Read and write data to EMC_DATA to initiate peripheral data transfer.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14 Hardware Acceleration Co-processor (CORDIC)

14.1 Overview

The CORDIC provides hardware acceleration of certain mathematical functions such as $m \cdot \sin\theta$, $m \cdot \cos\theta$, atan2(y, x), $\sqrt{x^2+y^2}$, $y \cdot x$, y/x, $\sinh w$, $\cosh w$, $tanh^{-1}(y/x)$, ln(x), $\sqrt{x}$, etc.

14.2 Main Features

- 24-bit CORDIC rotation engine
- AHB interface supporting 16-bit and 32-bit fixed point input and output formats
- DMA available
- Configurable input/output data address

14.3 Functional Description

14.3.1 Data Format and Input/Output

This module adopts 24-bit data internally for operation, and the data input and output can be selected in either 32-bit or 16-bit format for data transfer. The data format range in this section applies to all function modes.

When the 32-bit format is selected, the higher 24 bits are the input and output data, and the lower 8 bits are meaningless. If signed, the signed 32-bit format can be used for operation in

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the system. The decimal point of the data is located between bit [23] and bit [22]. If the data is signed, the whole data is represented in the range of [-1, (2^{23} - 1) / 2^{23}], i.e. [0x80000000, 0x7FFFFFFF]. If the data is not signed, it is represented in the range of [0, (2^{24} - 1) / 2^{23}], i.e. [0x0, 0xFFFFFF00]. The following is an example of data conversion [applicable to positive and negative numbers]:

Input Data x [DEC]	Calculation Formula	Converted Data [DEC]	Converted Data [HEX]
0.25	x * 2^{31}	536870912	0x20000000

When the 16-bit format is selected, the 16-bit data is the higher 16 bits of the internal data, and the lower 8 bits of the input data are filled with 0. If the data is signed, the signed 16-bit format can be used for operation in the system, and the whole data is represented in the range of [-1, (2^{15} - 1) / 2^{15}], i.e. [0x8000, 0x7FFF]. If the data is not signed, it is represented in the range of [0, (2^{16} - 1) / 2^{15}], i.e. [0x0, 0xFFFF].

Input Data x [DEC]	Calculation Formula	Converted Data [DEC]	Converted Data [HEX]
0.25	x * 2^{15}	8192	0x00002000

If two data inputs are required for an operation, the operation will start automatically when both data are written; if only one data is required, writing one data will automatically starts the operation.

14.3.2 Function Mode 0: $m \cdot \sin\theta$ / $m \cdot \cos\theta$

When the function mode 0: : $m \cdot \sin\theta$ / $m \cdot \cos\theta$ is selected, the input and output data are as follows:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 14-1: Input and Output Table of Cordic Controller Function Mode 0

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	θ	m	$m \cdot \sin\theta$	$m \cdot \cos\theta$
Sign bit	True	True	True	True
Remarks	The unit is $\pi\ rad$.	1 can be represented by 0x7FFFFF (XX) (32-bit mode) or 0x7FFF (16-bit mode), where no multiplication is performed.	-	-

Relevant calculation: $\tan\theta = \sin\theta \div \cos\theta$

Notes:

1. The sign bit indicates whether the data is signed, and the MSB bit of the signed data represents the sign.

   True: signed; False: Not signed.

2. For the data input range, please refer to Chapter 14.3.1 Data Format and Input/Output.

3. The above notes are applicable to all function modes.

14.3.3 Function Mode 1: Atan2(y, x) / $\sqrt{(x^2 + y^2)}$

When the function mode 1: Atan2(y, x) / $\sqrt{(x^2 + y^2)}$ is selected, the input and output data are as follows:

Table 14-2: Input and Output Table of Cordic Controller Function Mode 1

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	y	x	atan2 (y, x)	$\sqrt{x^2 + y^2}$
Sign bit	True	True	True	False
Remarks	-	-	The unit is $\pi\ rad$.	Bit [23] indicates that the decimal point is preceded by 0 or 1, not the sign, and the range is [0, $\sqrt{2}$].

Relevant calculation: $\tan^{-1} x = atan2(x \cdot 2^{-n}, 2^{-n})$

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14.3.4 Function Mode 2: $y \cdot x$

When the function mode 2: $y \cdot x$ is selected, the input and output data are as follows:

Table 14-3: Input and Output Table of Cordic Controller Function Mode 2

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	y	x	$y \cdot x$	-
Sign bit	True	True	True	-
Remarks	-	-	-	-

14.3.5 Function Mode 3: y/x

When the function mode 3: y/x is selected, the input and output data are as follows:

Table 14-4: Input and Output Table of Cordic Controller Function Mode 3

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	y	x	y/x	-
Sign bit	True	True	True	-
Remarks	It is required that $\|y\| \leq \|x\|$.		-	-

14.3.6 Function Mode 4: $\sinh w$ / $\cosh w$

When the function mode 4: $\sinh w$ / $\cosh w$ is selected, the input and output data are as follows:

Table 14-5: Input and Output Table of Cordic Controller Function Mode 4

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-1} \cdot w$	-	$2^{-1} \cdot \sinh w$	$2^{-1} \cdot \cosh w$
Sign bit	True	-	True	False
Remarks	$w \in [-1.1181, 1.1181]$	-	Range in [0, 1.366]	Range in [0, 1.693]

Relevant calculation: $e^w = \sinh w + \cosh w$

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14.3.7 Function Mode 5: $\tanh^{-1}(y/x)$

When the function mode 5: $\tanh^{-1}(y/x)$ is selected, the input and output data are as follows:

Table 14-6: Input and Output Table of Cordic Controller Function Mode 5

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	y	x	$2^{-1} \cdot \theta$	-
Sign bit	True	False	True	-
Remarks	It is able to take the point (y,x) from the ranges of $y \in [-1,1)$ and $x \in [0,2)$, or enter $y = 2^{-n} \cdot \tanh\theta$ and $x = 2^{-n}$, requiring that $(y/x) \in [-0.8069, 0.8069]$.		-	-

14.3.8 Function Mode 6: $\ln(x)$

When the function mode 6: $\ln(x)$ is selected, the input and output data format varies depending on the range of input data, requiring a numerical range SCALE in the control register, as follows:

When $x \in [0.1069, 1)$, make SCALE = 0.

Table 14-7: Input and Output Table 1 of Cordic Controller Function Mode 6

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	x	-	$2^{-2} \cdot \ln x$	-
Sign bit	False	-	True	-
Remarks	$x \in [0.1069, 1)$	-	-	-

When $x \in (1, 3)$, make SCALE = 1.

Table 14-8: Input and Output Table 2 of Cordic Controller Function Mode 6

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-1} \cdot x$	--	$2^{-2} \cdot \ln x$	-
Sign bit	False	--	True	-
Remarks	$x \in (1, 3)$	-	-	-

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When $x \in [3,7)$, make SCALE = 2.

Table 14-9: Input and Output Table 3 of Cordic Controller Function Mode 6

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-2} \cdot x$	-	$2^{-2} \cdot \ln x$	-
Sign bit	False	-	True	-
Remarks	$x \in [3,7)$	-	-	-

When $x \in [7,9.35]$, make SCALE = 3.

Table 14-10: Input and Output Table 4 of Cordic Controller Function Mode 6

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-3} \cdot x$	-	$2^{-2} \cdot \ln x$	-
Sign bit	False	-	True	-
Remarks	$x \in [7, 9.35]$	-	-	-

Relevant calculation: $w^t = e^{t \ln w}$

14.3.9 Function Mode 7: $\mathrm{Sqr(x)}$

When the function mode 7: $\mathrm{Sqr(x)}$, i.e. Sqr(x), is selected, the input and output data format varies depending on the range of input data, requiring a numerical range SCALE in the control register, as follows:

When $x \in [0.1069, 1)$, make SCALE = 0.

Table 14-11: Input and Output Table 1 of Cordic Controller Function Mode 7

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	x	-	$\sqrt{x}$	-
Sign bit	False	-	False	-
Remarks	$x \in [0.1069, 1)$	-	-	-

When $x \in (1, 3)$, make SCALE = 1.

Table 14-12: Input and Output Table 2 of Cordic Controller Function Mode 7

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-1} \cdot x$	-	$2^{-1} \cdot \sqrt{x}$	-
Sign bit	False	-	False	-

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Remarks	$x \in (1, 3)$	-	-	-

When $x \in [3, 7)$, make SCALE = 2.

Table 14-13: Input and Output Table 3 of Cordic Controller Function Mode 7

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-2} \cdot x$	-	$2^{-2} \cdot \sqrt{x}$	-
Sign bit	False	-	False	-
Remarks	$x \in [3,7)$	-	-	-

When $x \in [7, 9.35]$, make SCALE = 3.

Table 14-14: Input and Output Table 4 of Cordic Controller Function Mode 7

	Input Data 1	Input Data 2	Output Data 1	Output Data 2
Content	$2^{-3} \cdot x$	-	$2^{-3} \cdot \sqrt{x}$	-
Sign bit	False	-	False	-
Remarks	$x \in [7, 9.35]$	-	-	-

14.4 Register Description

Register base address: 0x4540_0000

The registers are listed below:

Table 14-15: List of Cordic Controller Registers

Offset Address	Name	Description
0x00	CORDIC_CTRL	Control register
0x04	CORDIC_DIN1	Input register 1
0x08	CORDIC_DIN2	Input register 2
0x0C	CORDIC_DOUT1	Output register 1
0x10	CORDIC_DOUT2	Output register 2

Registers are detailed in the following sections.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14.4.1 Control Register (CORDIC_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	DATA_READY	R	0x0	0: No currently-available operation result 1: With currently-available operation result
30:22	RSV	-	-	Reserved
21:20	SCALE	R/W	0x0	Scaling factor of argument calculated via ln(x) and $\sqrt{x}$ is determined by the value of x.
19:18	RSV	-	-	Reserved
17	DMA_OUT	R/W	0x0	Output data DMA enable: 0: Output data DMA disabled 1: Output data DMA enabled
16	DMA_IN	R/W	0x0	Input data DMA enable: 0: Input data DMA disabled 1: Input data DMA enabled
15:14	RSV	-	-	Reserved
13	MERGE_OUT	R/W	0x0	Output data merge, available only when the 16-bit bus mode is used for data output: 0: Data is output in two parts. 1: The lower 16 bits of the data output register 1 is read as the first argument, and the higher 16 bits is read as the second argument.
12	MERGE_IN	R/W	0x0	Input data merge, available only when the 16-bit bus mode is used for data input: 0: Data is input in two parts. 1: The lower 16 bits of the data input register 1 is written with the first

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				argument, and the higher 16 bits is written with the second argument.
11	ADDR_OUT	R/W	0x0	Output address mode: 0: Read data from two data output registers respectively. 1: Both arguments can be read sequentially from data output register 1.
10	ADDR_IN	R/W	0x0	Input address mode: 0: Two arguments shall be written into two data input registers respectively. 1: Both arguments can be written sequentially into data input register 1.
9	WIDTH_OUT	R/W	0x0	Data output transfer mode: 0: 32-bit bus transfer mode adopted for data output 1: 16-bit bus transfer mode adopted for data output
8	WIDTH_IN	R/W	0x0	Data input transfer mode: 0: 32-bit bus transfer mode adopted for data input 1: 16-bit bus transfer mode adopted for data input
7:4	ITERATION	R/W	12	(Number of iterations in operation / 2) The more number of iterations, the more accurate it is and the longer it takes. The available range is [4, 12].
3	RSV	-	-	Reserved
2:0	MODE	R/W	0x0	Function modes 0: Mode 0 selected $m \cdot \sin\theta$ / $m \cdot \cos\theta$ 1: Mode 1 selected atan2(y,x) / $\sqrt{x^2 + y^2}$ 2: Mode 2 selected $y \cdot x$ 3: Mode 3 selected

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				y/x 4: Mode 4 selected $\sinh w / \cosh w$ 5: Mode 5 selected $tanh^{-1}(y/x)$ 6: Mode 6 selected $\ln(x)$ 7: Mode 7 selected $\sqrt{x}$

14.4.2 Data Input Register 1 (CORDIC_DIN1)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DIN1	W	0x0	Write input data to this register.

14.4.3 Data Input Register 2 (CORDIC_DIN2)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DIN2	W	0x0	Write input data to this register.

14.4.4 Data Output Register 1 (CORDIC_DOUT1)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DOUT1	R	0x0	Read the operation result from this register.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14.4.5 Data Output Register 2 (CORDIC_DOUT2)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DOUT2	R	0x0	Read the operation result from this register.

14.5 Operation Procedure

1. Enable CORDIC module clock and release the reset.
2. Set the input/output address format, data format, etc.
3. Set the number of operations (the higher the number, the more accurate the operation result).
4. Set the operation mode and the input/output data transfer mode.
5. Input the appropriate data to CORDIC_DIN1 and CORDIC_DIN2 according to the set operation mode.
6. Wait for CORDIC_CTRL[31] to be set.
7. Read the two output data from CORDIC_DOUT1 and CORDIC_DOUT2.

14.6 Calculation Example

The relevant calculation background in this section is the default configuration of CORDIC_CTRL:

[input/output calculation in other modes can refer to this mode].

Default configuration: Input/output data format: 32-bit

Data input and output: Input through two input registers

Data transfer mode: CPU

Calculation function mode: Function mode 0

Calculation method for converting input data 1 into the hardware-specified format:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

0.25 [DEC] = 0.25 * 2^{31} [DEC]

= 536,870,912 [DEC]

= 0x20000000 [HEX]

Calculation method for converting input data 2 into the hardware-specified format: 0.00390625 [DEC] = 0.00390625 * 2^{31} [DEC]

= 8388608 [DEC]

= 0x00800000 [HEX]

Input the above calculation results into CORDIC_DIN1 and CORDIC_DIN2 respectively, then the hardware can start the calculation, the results of which can be obtained from CORDIC_DOUT1 and CORDIC_DOUT2.

Verify the calculation results:

Theoretical calculation result of output data 1:

m * sinθ = 0.00390625 * (sin (0.25 * π)) [DEC]

= 0.00390625 * 0.7071067811865 [DEC]

= 0.00276213586400995 12671907982 [DEC] * 2^{31}

= 5931641.601515722055569 [DEC]

= 5A8279 [HEX]

Theoretical calculation result of output data 2:

m * cosθ = 0.00390625 * (cos (0.25 * π)) [DEC]

= 0.00390625 * 0.7071067811865 [DEC]

= 0.00276213586400995 [DEC]

= 5,931,641.601515722055569 [DEC] * 2^{31}

= 5A8279 [HEX]

The result of hardware automatic calculation is 0x5a8300 [error: 135, mainly from the decimal point].

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15 Advanced Encryption and Decryption Algorithm Accelerator (AES)

15.1 Overview

The AES module encrypts or decrypts data using AES-128 or AES-256 algorithm defined in Federal information processing standards (FIPS) publication 197.

15.2 Main Features

- Cipher key lengths of 128 or 256 bits
- Support multiple chaining modes: CBC, ECB, ETR, CCM, CMAC and GCM
- Support encryption and decryption
- All data is in big-endian order according to AES regulations.
- Separate interfaces for key and message: low-capacity interface for key, and FIFO interacting with the interface for data input and output
- Using DMA for data input and output
- Instant key expansion available, no additional storage space required
- For a message, only packet-based encryption and decryption are supported, while message switching is not supported.
- Three-key mode supported

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.3 Functional Description

15.3.1 Encryption and Decryption

Once the key is loaded, the algorithm core can be started by enabling the AES_GO control bit, which will automatically read the data and perform the algorithm. Upon completion of the algorithm, the AES_DONE bit is enabled and the data is automatically output.

15.4 Register Description

Register base address: 0x4510_0000

The registers are listed below:

Table 15-1: List of AES Registers

Offset Address	Name	Description
0x00	AES_MSGCFG	Message control register
0x04	AES_CTXCFG	Key control register
0x08	AES_MSGTOTALBYTES	Message length register
0x0C	AES_MSGAADBYTES	Additional message length register
0x10	AES_GCMAADINFO	GCM mode additional message length register
0x14	AES_CTXKEYSEL	Key selection register
0x20	AES_CTXKEY0	Key register 0
0x24	AES_CTXKEY1	Key register 1
0x28	AES_CTXKEY2	Key register 2
0x2C	AES_CTXKEY3	Key register 3
0x30	AES_CTXKEY4	Key register 4
0x34	AES_CTXKEY5	Key register 5
0x38	AES_CTXKEY6	Key register 6
0x3C	AES_CTXKEY7	Key register 7
0x40	AES_CTXCBCKEY0	CBC mode key register 0
0x44	AES_CTXCBCKEY1	CBC mode key register 1
0x48	AES_CTXCBCKEY2	CBC mode key register 2

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x4C	AES_CTXCBCKEY3	CBC mode key register 3
0x50	AES_CTXCTR0	CTR mode operator register 0
0x54	AES_CTXCTR1	CTR mode operator register 1
0x58	AES_CTXCTR2	CTR mode operator register 2
0x5C	AES_CTXCTR3	CTR mode operator register 3
0x60	AES_CTXIV0	Initialization vector register 0
0x64	AES_CTXIV1	Initialization vector register 1
0x68	AES_CTXIV2	Initialization vector register 2
0x6C	AES_CTXIV3	Initialization vector register 3
0x70	AES_CTXMAC0	Message authentication code register 0
0x74	AES_CTXMAC1	Message authentication code register 1
0x78	AES_CTXMAC2	Message authentication code register 2
0x7C	AES_CTXMAC3	Message authentication code register 3
0x80	AES_INGRESSFIFO	Input FIFO register
0x84	AES_INGFSTATUS	Input FIFO status register
0x88	AES_ENGRESSFIFO	Output FIFO register
0x8C	AES_ENGFSTATUS	Output FIFO status register
0x90	AES_DMAINGLEN	DMA input length register
0x94	AES_INGDBCFG	DMA input burst transfer setting register
0x98	AES_DMAENGLEN	DMA output length register
0x9C	AES_ENGDBCFG	DMA output burst transfer setting register
0xA0	AES_DONESTATUS	Done status register
0xA4	AES_INGDMADONE	DMA input done status register
0xA8	AES_ENGDMADONE	DMA output done status register

Registers are detailed in the following sections.

15.4.1 Message Control Register (AES_MSGCFG)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13:12	KEY_SIZE	R/W	0x0	Key size selection: 0: 128 bits 2: 256 bits 1 & 3: Reserved
11:8	MAC_LEN	R/W	0x0	Set the message authentication code or initialization vector length (only for CCM, CMA and XCBC modes)
7:4	ALG_MODE	R/W	0x0	Chaining mode selection: 0: ECB 1: CBC 2: CTR 3: CCM 4: CMAC 5: GCM
3	DIR	R/W	0x0	Encryption/decryption selection: 0: Decryption 1: Encryption
2	MSG_BEGIN	R/W	0x0	The first byte of the message contained in the current calculation; the required key will be automatically obtained before the calculation.
1	MSG_END	R/W	0x0	The last byte of the message contained in the current calculation; if the calculation result is the message authentication code or initialization vector, it will be automatically stored in the key storage space upon completion of the calculation.
0	AES_GO	R/W	0x0	Start encryption/decryption, after which this bit and other control bits shall remain unchanged until AES_DONE is set.

15.4.2 Key Control Register (AES_CTXCFG)

Offset address: 0x04

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:4	CTX_INDEX	R/W	0x0	Context page number
3	INV_KEY_STR	R/W	0x0	Store the last round key to the key storage space
2	INV_KET_RET	R/W	0x0	Retrieve the last round key
1	CTX_STR	R/W	0x0	Store the current key to the key storage space
0	CTX_RET	R/W	0x1	Retrieve the current key

15.4.3 Message Length Register (AES_MSGTOTALBYTES)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	Reserved
27:0	TOTAL_BYTES	R/W	0x0	Set the total length of message, including all packets while excluding additional message. This is required in CCM and GCM modes.

15.4.4 Additional Message Length Register (AES_MSGAADBYTES)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	AAD_LEN	R/W	0x0	Set the additional message length in the current calculation. This is required in CCM and GCM modes.
15:0	MSG_LEN	R/W	0x0	Set the length of message in the current calculation, including message and additional message.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.5 GCM Mode Additional Message Length Register (AES_GCMAADINFO)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	Reserved
27:0	AAD_LEN_TOT	R/W	0x0	Set the total length of additional message, including all packets. This is required in GCM mode.

15.4.6 Key Selection Register (AES_CTXKEYSEL)

Offset address: 0x14

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2:0	KEY_SEL	R/W	0x1	Key source selection: 1: Key set by CPU (register) 2: OTP KEY1 4: OTP KEY2

15.4.7 Key Register 0 (AES_CTXKEY0)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_0	W	0x0	Key [31:0] Note: ECB mode key register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.8 Key Register 1 (AES_CTXKEY1)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_1	W	0x0	Key [63:32]

15.4.9 Key Register 2 (AES_CTXKEY2)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_2	W	0x0	Key [95:64]

15.4.10 Key Register 3 (AES_CTXKEY3)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_3	W	0x0	Key [127:96]

15.4.11 Key Register 4 (AES_CTXKEY4)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_4	W	0x0	Key [159:128]

15.4.12 Key Register 5 (AES_CTXKEY5)

Offset address: 0x34

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_5	W	0x0	Key [191:160]

15.4.13 Key Register 6 (AES_CTXKEY6)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_6	W	0x0	Key [223:192]

15.4.14 Key Register 7 (AES_CTXKEY7)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	KEY_7	W	0x0	Key [255:224]

15.4.15 CBC Mode Key Register 0 (AES_CTXCBCKEY0)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CBC_KEY_0	W	0x0	CBC key [31:0] (XCBC-3K K2)

15.4.16 CBC Mode Key Register 1 (AES_CTXCBCKEY1)

Offset address: 0x44

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CBC_KEY_1	W	0x0	CBC key [63:32] (XCBC-3K K2)

15.4.17 CBC Mode Key Register 2 (AES_CTXCBCKEY2)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CBC_KEY_2	W	0x0	CBC key [95:64] (XCBC-3K K2)

15.4.18 CBC Mode Key Register 3 (AES_CTXCBCKEY3)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CBC_KEY_3	W	0x0	CBC key [127:96] (XCBC-3K K2)

15.4.19 CTR Mode Operator Register 0 (AES_CTXCTR0)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCM_CTR_0	W	0x0	CTR/CCM operator [31:0] (XCBC-3K K1)

15.4.20 CTR Mode Operator Register 1 (AES_CTXCTR1)

Offset address: 0x54

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CCM_CTR_1	W	0x0	CTR/CCM operator [63:32] (XCBC-3K K1)

15.4.21 CTR Mode Operator Register 2 (AES_CTXCTR2)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCM_CTR_2	W	0x0	CTR/CCM operator [95:64] (XCBC-3K K1)

15.4.22 CTR Mode Operator Register 3 (AES_CTXCTR3)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCM_CTR_3	W	0x0	CTR/CCM operator [127:96] (XCBC-3K K1)

15.4.23 Initialization Vector Register 0 (AES_CTXIV0)

Offset address: 0x60

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	IV_0	W	0x0	Initialization vector [31:0]

15.4.24 Initialization Vector Register 1 (AES_CTXIV1)

Offset address: 0x64

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31:0	IV_1	W	0x0	Initialization vector [63:32]

15.4.25 Initialization Vector Register 2 (AES_CTXIV2)

Offset address: 0x68

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	IV_2	W	0x2	Initialization vector [95:64]

15.4.26 Initialization Vector Register 3 (AES_CTXIV3)

Offset address: 0x6C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	IV_3	W	0x0	Initialization vector [127:96]

15.4.27 Message Authentication Code Register 0 (AES_CTXMAC0)

Offset address: 0x70

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MAC_0	W	0x0	Message authentication code [31:0] in CMAC/XCBC/CCM mode

15.4.28 Message Authentication Code Register 1 (AES_CTXMAC1)

Offset address: 0x74

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MAC_1	W	0x0	Message authentication code [63:32] in CMAC/XCBC/CCM mode

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.29 Message Authentication Code Register 2 (AES_CTXMAC2)

Offset address: 0x78

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MAC_2	W	0x0	Message authentication code [95:64] in CMAC/XCBC/CCM mode

15.4.30 Message Authentication Code Register 3 (AES_CTXMAC3)

Offset address: 0x7C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	MAC_3	W	0x0	Message authentication code [127:96] in CMAC/XCBC/CCM mode

15.4.31 Input FIFO (AES_INGRESSFIFO)

Offset address: 0x80

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	INGRESS_FIFO	W	0x0	AES input data FIFO ingress

15.4.32 Input FIFO Status Register (AES_INGFSTATUS)

Offset address: 0x84

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1	ING_FULL	R	0x0	Input data FIFO full
0	ING_EMPTY	R	0x1	Input data FIFO empty

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.33 Output FIFO (AES_ENGRESSFIFO)

Offset address: 0x88

Reset value: uncertain

Bit	Name	Attribute	Reset Value	Description
31:0	ENGRESS_FIFO	R	-	AES output data FIFO egress

15.4.34 Output FIFO Status Register (AES_ENGFSTATUS)

Offset address: 0x8C

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1	ENG_FULL	R	0x0	Output data FIFO full
0	ENG_EMPTY	R	0x1	Output data FIFO empty

15.4.35 DMA Input Length Register (AES_DMAINGLEN)

Offset address: 0x90

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	INGRESS_DMA_DATA_LENGTH	R/W	0x0	Set the total length of data input from DMA to AES for generating handshake signal. This is required even if CPU is used instead of DMA to transfer data.

15.4.36 DMA Input Burst Transfer Setting Register (AES_INGDBCFG)

Offset address: 0x94

Reset value: 0xC000 0012

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31	PLAINTEXT_REQ	R/W	0x1	This bit sets whether the current AES mode requires inputting data.
30	DMA_EN	R/W	0x1	This bit sets whether the input data is transferred via DMA or CPU: 0: CPU 1: DMA
29:6	RSV	-	-	Reserved
5:3	DST_MSIZE	R/W	0x2	DMA destination burst transfer length: 0: 1 1: 4 2: 8 3: 16 4: 32 5: 64 6: 128 7: 256
2:0	DST_TR_WIDTH	R/W	0x2	DMA destination transfer width: 0: 8 bits 1: 16 bits 2: 32 bits

15.4.37 DMA Output Length Register (AES_DMAENGLEN)

Offset address: 0x98

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ENGRESS_DMA_DATA_LENGTH	R/W	0x0	Set the total length of data output by DMA from AES for generating handshake signal.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.4.38 DMA Output Burst Transfer Setting Register (AES_ENGDBCFG)

Offset address: 0x9C

Reset value: 0xC000 0012

Bit	Name	Attribute	Reset Value	Description
31	CIPERTEXT_REQ	R/W	0x1	This bit sets whether the current AES mode requires outputting data.
30	DMA_EN	R/W	0x1	This bit sets whether the input data is transferred via DMA or CPU: 0: CPU 1: DMA
29:6	RSV	-	-	Reserved
5:3	SRC_MSIZE	R/W	0x2	DMA source burst transfer length: 0: 1 1: 4 2: 8 3: 16 4: 32 5: 64 6: 128 7: 256
2:0	SRC_TR_WIDTH	R/W	0x2	DMA source transfer width: 0: 8 bits 1: 16 bits 2: 32 bits

15.4.39 Done Status Register (AES_DONESTATUS)

Offset address: 0xA0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	MAC_VAILD	R	0x0	Indicating whether the MAC authentication is successful, this bit is valid only after AED_DONE is set, and automatically cleared when AES_GO is set.
0	AES_DONE	R	0x0	Indicating that the calculation has been completed, this bit is automatically cleared when AES_GO is set.

15.4.40 DMA Input Done Status Register (AES_INGDMADONE)

Offset address: 0xA4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	DMA_INGRESS_DONE	R	0x0	Indicating that DMA input has been completed, this bit is cleared when it is read.

15.4.41 DMA Output Done Status Register (AES_ENGDMADONE)

Offset address: 0xA8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	DMA_ENGRESS_DONE	R	0x0	Indicating that DMA output has been completed, this bit is cleared when it is read.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15.5 Operation Procedure

15.5.1 AES Encryption Model

1. Configure the AED_CTXKEYSEL register and populate the AES_CTXKEYx register.
2. Input data to AES_INGRESSFIFO and retrieve data from AES_ENGRESSFIFO.
3. Configure AES_MSGCFG to set the direction as encryption, and finally set AES_MSGCFG[0].
4. When the AES module completes the operation of current packet, the AES_DONE bit will be set and an interrupt will be generated.
5. The query or interrupt determines the completion flag.
6. If the current packet is not the last one, repeat steps 2 and 3.
7. If the current packet is the last one and the encryption mode that can generate message authentication code is used, retrieve the message authentication code from AES_CTXMAC.

15.5.2 AES Decryption Model

1. Configure the AED_CTXKEYSEL register and populate the AES_CTXREGx register.
2. Configure DMA to input data to AES_INGRESSFIFO and retrieve data from AES_ENGRESSFIFO.
3. Configure AES_MSGCFG to set the direction as decryption, and finally set AES_MSGCFG[0].
4. When the AES module completes the operation of current packet, the AES_DONE bit will be set and an interrupt will be generated.
5. The query or interrupt determines the completion flag.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. If the encryption mode that can generate message authentication code is used, retrieve the message authentication code from AES_CTXMAC.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

16 Secure Hash Algorithm Accelerator (SHA)

16.1 Hash Overview

The SHA module can implement the SHA-256 algorithm defined in FIPS publication 180-4.

16.2 Main Features

- 256-bit ICV length
- Information in external storage shall be in little-endian order with double words (4 bytes) aligned
- 32 x 32-bit input data FIFO
- Using DMA for data input

16.3 Functional Description

16.3.1 Encryption and Decryption

Enable the SHA_GO control bit to start the algorithm core, and input the data to start the calculation. The SHA_DONE bit is enabled upon completion of the algorithm.

16.3.2 OTP Value Comparison

When the SHA operation result is the same as the OTP ICV value, only the ICV match can be known by the SHA module, and the specific ICV value cannot be read out.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

To use the OTP comparison function, please write the OTP ICV to the OTP storage area in little-endian order, and the OTP ICV in SHA-256 format shall be written to the [383:128] bits of the OTP storage area, and write 0 to the [127:0] of the OTP storage area.

16.4 Register Description

SHA register base address: 0x4520_0000

The registers are listed below:

Table 16-1: List of SHA Registers

Offset Address	Name	Description
0x00	SHA_CMD	Control command register
0x04	SHA_MODE	Bit mode register
0x08	SHA_MSGCFG	Data setting register
0x18	SHA_MSGTOTALBYTES	Data length register
0x1C	SHA_DATAIN	Data input register
0x20	SHA_ICV00	ICV read register 00
0x24	SHA_ICV01	ICV read register 01
0x28	SHA_ICV02	ICV read register 02
0x2C	SHA_ICV03	ICV read register 03
0x30	SHA_ICV04	ICV read register 04
0x34	SHA_ICV05	ICV read register 05
0x38	SHA_ICV06	ICV read register 06
0x3C	SHA_ICV07	ICV read register 07
0x40	SHA_ICV08	ICV read register 08
0x44	SHA_ICV09	ICV read register 09
0x48	SHA_ICV10	ICV read register 10
0x4C	SHA_ICV11	ICV read register 11
0x50	SHA_VERIRESULT	Verification result register
0x54	SHA_IRQEN	Interrupt enable register
0x58	SHA_IRQCLR	Interrupt clear register
0x5C	SHA_IRQSTATUS	Interrupt status register
0x60	SHA_DMABURSTSIZE	DMA burst length register

Registers are detailed in the following sections.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

16.4.1 Control Command Register (SHA_CMD)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1	SHA_DONE	R	0x0	Indicating the completion of current segment algorithm
0	SHA_GO	R/W	0x0	Writing 1 starts the algorithm, upon completion of which it is automatically cleared, and writing 0 has no effect. Read to know whether the algorithm is currently in progress or not.

16.4.2 Bit Mode Register (SHA_MODE)

Offset address: 0x04

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1:0	MODE	R/W	0x1	SHA algorithm mode selection: 1: SHA-256 2: SHA-384 Other values: Invalid Note: If it is set to other invalid values and the SHA algorithm is activated, the SHA accelerator shall be reset and set to a valid value to calculate correctly again.

16.4.3 Data Setting Register (SHA_MSGCFG)

Offset address: 0x08

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:27	RSV	-	-	Reserved
26:12	NUM_BYTES	R/W	0x0	Set the length of current segment. If MSG_END is not enabled, this field must be set to a multiple of the algorithm block size. For SHA-256, the block size is 64 bytes. For SHA-384, the block size is 128 bytes.
11:2	RSV	-	-	Reserved
1	MSG_BEGIN	R/W	0x0	Indicating that the current segment is the first one of the whole message.
0	MSG_END	R/W	0x0	Indicating that the current segment is the last one of the whole message.

16.4.4 Data Length Register (SHA_MSGTOTALBYTES)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	Reserved
25:0	TOTAL_BYTES	R/W	0x0	Length of the whole message, valid only when MSG_END is enabled.

16.4.5 Data Input Register (SHA_DATAIN)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DATA_IN	R/W	0x0	Enter the data to be calculated here.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

16.4.6 ICV Read Register (SHA_ICVn)

Offset address: 0x20−0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ICV	R	0x0	When the algorithm result does not match the OTP ICV, the results can be read from these registers. For SHA-256, SHA_ICV 04−11 stores the algorithm result. For SHA-384, SHA_ICV 00−11 stores the algorithm result. SHA_ICV 00 stores ICV[31:0]. SHA_ICV 01 stores ICV[63:32]. SHA_ICV 02 stores ICV[95:64]. SHA_ICV 03 stores ICV[127:96]. SHA_ICV 04 stores ICV[159:128]. SHA_ICV 05 stores ICV[191:160]. SHA_ICV 06 stores ICV[223:192]. SHA_ICV 07 stores ICV[255:224]. SHA_ICV 08 stores ICV[287:256]. SHA_ICV 09 stores ICV[319:288]. SHA_ICV 10 stores ICV[351:320]. SHA_ICV 11 stores ICV[383:352]. Note: The result of SHA-256[255:0] is stored in SHA_ICV[128:383].

16.4.7 Verification Result Register (SHA_VERIRESULT)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	VERI_VLD	R	0x0	Indicating that ICV verification has been performed; triggered by SHA_DONE + SHA_MSG_END and cleared by enabling SHA_GO.
0	VERI_FAIL	R	0x0	Indicating whether ICV matches the OTP ICV value: 0: Matches 1: Does not match

16.4.8 Interrupt Enable Register (SHA_IRQEN)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	IRQ_EN	R/W	0x0	Algorithm done interrupt enable: 0: Disabled 1: Enabled

16.4.9 Interrupt Clear Register (SHA_IRQCLR)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	IRQ_CLR	W	0x0	Write 1 to clear this interrupt.

16.4.10 Interrupt Status Register (SHA_IRQSTATUS)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	IRQ_STATUS	R	0x0	Interrupt status: 0: No interrupt occurred 1: Interrupt occurred

16.4.11 DMA Burst Length Register (SHA_DMABURSTSIZE)

Offset address: 0x60

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2:0	DMA_BURST_SIZE	R/W	0x3	DMA burst length: 0x0: 1 0x1: 4 0x2: 8 0x3: 16 0x4: 32 It shall be consistent with the setting in DMA module. For example, in DMA burst 4, if the data operation unit is a byte, it means 4 bytes are operated on. If the data operation unit is 32-bit data, it means 4 units of 32-bit data are operated on.

16.5 Operation Procedure

16.5.1 SHA Algorithm Application (Input Message via DMA)

1. Enable SHA clock and release SHA reset.
2. Enable DMA1 clock and release DMA1 reset.
3. Initialize DMA and channel configuration, for example, configure the source address to be automatically incremented by 1 and the destination address to be unchanged for DMA1

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

channel 2 transfer from memory to peripheral. Both the source and destination DMA burst lengths are 1. The source data transfer width is 32 bits, and the destination data transfer width is 32 bits. Both source and destination transfers are hardware synchronized. The source handshake signal is an arbitrary value (memory), and the destination handshake signal is signal 7.

4. Enable DMA module.
5. Configure SHA_MODE[1:0] to select SHA-256 or SHA-384 mode.
6. Configure SHA_MSGCFG[1] and SHA_MSGCFG[0], if the message length is less than 64 bytes (for SHA-256) or 128 bytes (for SHA-384), it will be a single segment, with both BEGIN and END set to 1. If the length exceeds a single segment, it shall be set according to the first segment, the middle segment and the last segment respectively.
7. Configure SHA_MSGCFG[26:12] for the length of current segment.
8. Configure the SHA_MSGTOTALBYTES register for the length of the entire message.
9. Obtain the plaintext to be encrypted. For example, plaintext to be encrypted can be stored in the legal address space of SRAM.
10. Enable the SHA_CMD register, and SHA_GO = 1 initiate an SHA algorithm.
11. Initiate DMA to transfer data to SHA, for example, regarding DMA1 channel 2 transfer, configure the destination address SHA_DATA_IN to be not automatically incremented and the source address as the first address of plaintext to be automatically incremented. Configure the length of the transferred data (according to DMA1 source data width, if the source data width is 32 bits, the length of current segment shall be divided by 32). See DMA chapter or sample code for DMA configuration.
12. Wait for SHA_DONE to be set.
13. Check the encryption result register SHA_ICVn, where SHA_ICV04-11 stores the algorithm result corresponding to SHA-256 and SHA_ICV00-11 stores the algorithm result corresponding to SHA0384.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14. Store the encryption result in OTP SHA ICV in advance (see EFC chapter or sample code for details) to check whether the verification result register SHA_VERIRESULT meets expectations.
15. Repeat steps 6–12 if there are multiple message segments.

16.5.2 SHA Algorithm Application (Input Message via CPU)

1. Enable SHA clock and release SHA reset.
2. Configure SHA_MODE[1:0] to select SHA-256 or SHA-384 mode.
3. Configure SHA_MSGCFG[1] and SHA_MSGCFG[0], if the message length is less than 64 bytes (for SHA-256) or 128 bytes (for SHA-384), it will be a single segment, with both BEGIN and END set to 1. If the length exceeds a single segment, it shall be set according to the first segment, the middle segment and the last segment respectively.
4. Configure SHA_MSGCFG[26:12] for the length of current segment.
5. Configure the SHA_MSGTOTALBYTES register for the length of the entire message.
6. Obtain the plaintext to be encrypted. For example, plaintext to be encrypted can be stored in the legal address space of SRAM.
7. Enable the SHA_CMD register, and SHA_GO = 1 initiate an SHA algorithm.
8. Splice the plaintext into 32-bit words in little-endian order by CPU and loop them into the SHA_DATAIN register.
9. Wait for SHA_DONE to be set.
10. Check the encryption result register SHA_ICVn, where SHA_ICV04-11 stores the algorithm result corresponding to SHA-256 and SHA_ICV00-11 stores the algorithm result corresponding to SHA0384.
11. Store the encryption result in OTP SHA ICV in advance (see EFC chapter or sample code for details) to check whether the verification result register SHA_VERIRESULT meets expectations.
12. Repeat steps 3–9 if there are multiple message segments.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

17 True Random Number Generator (RNG)

17.1 Overview

The RNG is capable of delivering true random numbers.

17.2 Main Features

The RNG delivers true random numbers from random seeds.

17.3 Functional Description

17.3.1 RNG

RNG is capable of delivering true random numbers from random seeds.

17.4 Register Description

Register base address: 0x40B0_D000

The registers are listed below:

Table 17-1: List of RNG Registers

Offset Address	Name	Description
0x00	RNG_DATA	Data register
0x04	RNG_SEED	Random seed register
0x08	RNG_CR	Control register

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

17.4.1 Data Register (RNG_DATA)

Offset address: 0x00

Reset value: 0xXXXX XXXX (the data read is random)

Bit	Name	Attribute	Reset Value	Description
31:0	DATA	R/W	0xXXXX XXXX	Read data to obtain random number

17.4.2 Random Seed Register (RNG_SEED)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SEED	W	0x0	Reserved

17.4.3 Control Register (RNG_CR)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	EN	R/W	0x0	Enable random seed to automatically change with time: 1: Enabled 0: Disabled

17.5 Operation Procedure

1. Configure the RNG_CR register to enable the random number generator.
2. Configure the RNG_SEED register to write a random seed.
3. Read the RNG_DATA register for random values, which can be read continuously.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18 Advanced Control Timer (TIM1 & TIM8)

18.1 Overview

The advanced-control timer consists of a 32-bit auto-reload counter driven by a programmable prescaler.

It may be used for a variety of purposes, including measuring the pulse lengths of input signals (input capture) or generating output waveforms (output compare, complementary PWM with dead-time insertion).

18.2 Main Features

- 32-bit up, down, up/down auto-reload counter
- 16-bit programmable prescaler allowing real-time adjustment of the counter clock division
- 4 independent channels for input capture, output compare, PWM generation, and one-pulse output
- Complementary output with programmable dead-time
- Support cascading with other timers
- Repetition counter to update the timer registers only after a given number of cycles of the counter
- Break inputs, break signal filtering and polarity selection, combinatorial configuration of break signals
- Interrupt or DMA event can be generated in the following cases:
  - Counter overflow/underflow, counter initialization (triggered by software or hardware)
  - Trigger event (counter start, stop, initialization, or count by internal/external trigger)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Input capture
- Output compare
- Break input

- Support incremental quadrature encoder and Hall sensor
- Trigger input for external clock

18.3 System Block Diagram

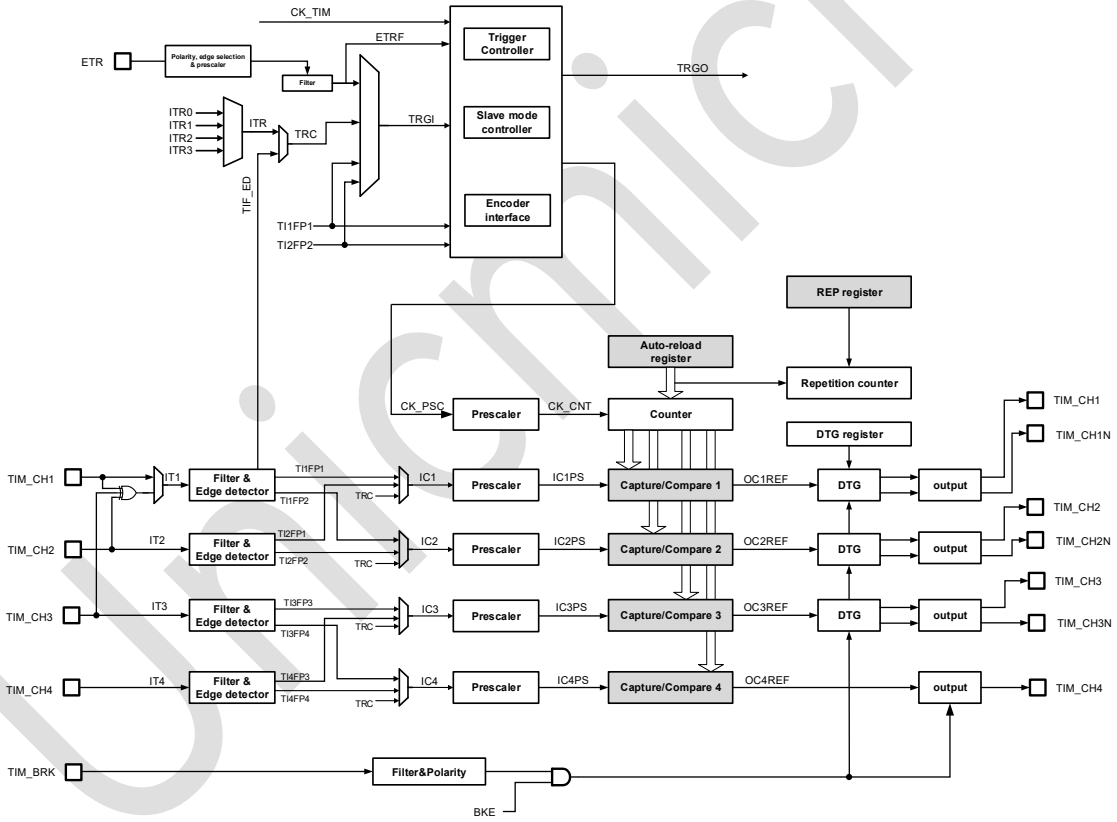

Figure 18-1: TIM1 & TIM8 System Block Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.4 Pin Description

Table 18-1: TIM1& TIM8 Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
TIM1_BKIN	PA6, PB12, PE15	Input	Break input
TIM1_ETR	PA12, PE7	Input	External trigger input
TIM1_CH1	PA8, PE9	Input/output	Channel input capture / PWM output signal
TIM1_CH1N	PA7, PB13, PE8	Output	PWM output inversion
TIM1_CH2	PA9, PE11	Input/output	Channel input capture / PWM output signal
TIM1_CH2N	PB0, PB14, PE10	Output	PWM output inversion
TIM1_CH3	PA10, PA4, PE13	Input/output	Channel input capture / PWM output signal
TIM1_CH3N	PB1, PB15, PE12	Output	PWM output inversion
TIM1_CH4	PA11, PA5, PE14	Input/output	Channel input capture / PWM output signal
TIM8_BKIN	PA6	Input	Break input
TIM8_ETR	PA0	Input	External trigger input
TIM8_CH1	PC6	Input/output	Channel input capture / PWM output signal
TIM8_CH1N	PA5, PA7	Output	PWM output inversion
TIM8_CH2	PC7	Input/output	Channel input capture / PWM output signal
TIM8_CH2N	PB0, PB14	Output	PWM output inversion
TIM8_CH3	PC8	Input/output	Channel input capture / PWM output signal
TIM8_CH3N	PB1, PB15	Output	PWM output inversion
TIM8_CH4	PC9	Input/output	Channel input capture / PWM output signal

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.5 Timer Interconnection

Table 18-2: Timer Interconnection

TIMx Interconnect								
Source (TIMx Trigger Output)	**Inputs (TIMx Trigger Input)**							
TIM1_TRGO	–	TIM2_ITR0	TIM3_ITR0	TIM4_ITR0	–	TIM8_ITR0	–	–
TIM2_TRGO	TIM1_ITR1	–	TIM3_ITR1	TIM4_ITR1	TIM5_ITR0	TIM8_ITR1	TIM9_ITR0	–
TIM3_TRGO	TIM1_ITR2	TIM2_ITR2	–	TIM4_ITR2	TIM5_ITR1	–	TIM9_ITR1	–
TIM4_TRGO	TIM1_ITR3	TIM2_ITR3	TIM3_ITR3	–	TIM5_ITR2	TIM8_ITR2	–	TIM12_ITR0
TIM5_TRGO	TIM1_ITR0	–	TIM3_ITR2	–	–	TIM8_ITR3	–	TIM12_ITR1
TIM8_TRGO	–	TIM2_ITR1	–	TIM4_ITR3	TIM5_ITR3	–	–	–
TIM9_TRGO	–	–	–	–	–	–	–	–
TIM10_TRGO	–	–	–	–	–	–	TIM9_ITR2	–
TIM11_TRGO	–	–	–	–	–	–	TIM9_ITR3	–
TIM12_TRGO	–	–	–	–	–	–	–	–
TIM13_TRGO	–	–	–	–	–	–	–	TIM12_ITR2
TIM14_TRGO	–	–	–	–	–	–	–	TIM12_ITR3

18.6 Functional Description

18.6.1 Time-base Unit

The main block of the time-base unit is a 32-bit counter with its related auto-reload register. The counter can count up, down, or both up and down. The counter clock can be divided by a 16-bit prescaler.

The counter, the auto-reload register and the prescaler register can be written or read by software, which is true even when the counter is running.

The time-base unit includes:

- Counter register (TIM_CNT)
- Prescaler register (TIM_PSC)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Auto-reload register (TIM_ARR)
- Repetition counter register (TIM_RCR)

The auto-reload register is preloaded, which is controlled by the auto-reload preload enable (ARPE) bit in the register. When ARPE = 0, write to the ARR register, and the written data is directly transferred to the shadow register. When ARPE = 1, the data written to the TIM_ARR register is transferred to the shadow register when an update event (TIM_CNT overflow or underflow) occurs. The update event of ARR can also be actively triggered by software via register operation.

The counter TIM_CNT is clocked by the prescaler output TIM_PSC, which is enabled only when the counter enable bit (CEN) in the register is set. When CNT = ARR, this round of counting is over and the update event is sent.

TIM_PSC is a synchronous prescaler that can divide the counter clock frequency by any factor between 1 and 65536. The PSC register is also buffered, and overwriting PSC does not actually overwrite the shadow register unless a new update event occurs. Thus the PSC register can be changed in real time on the fly, and the new prescaler ratio is taken into account at the next update event.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

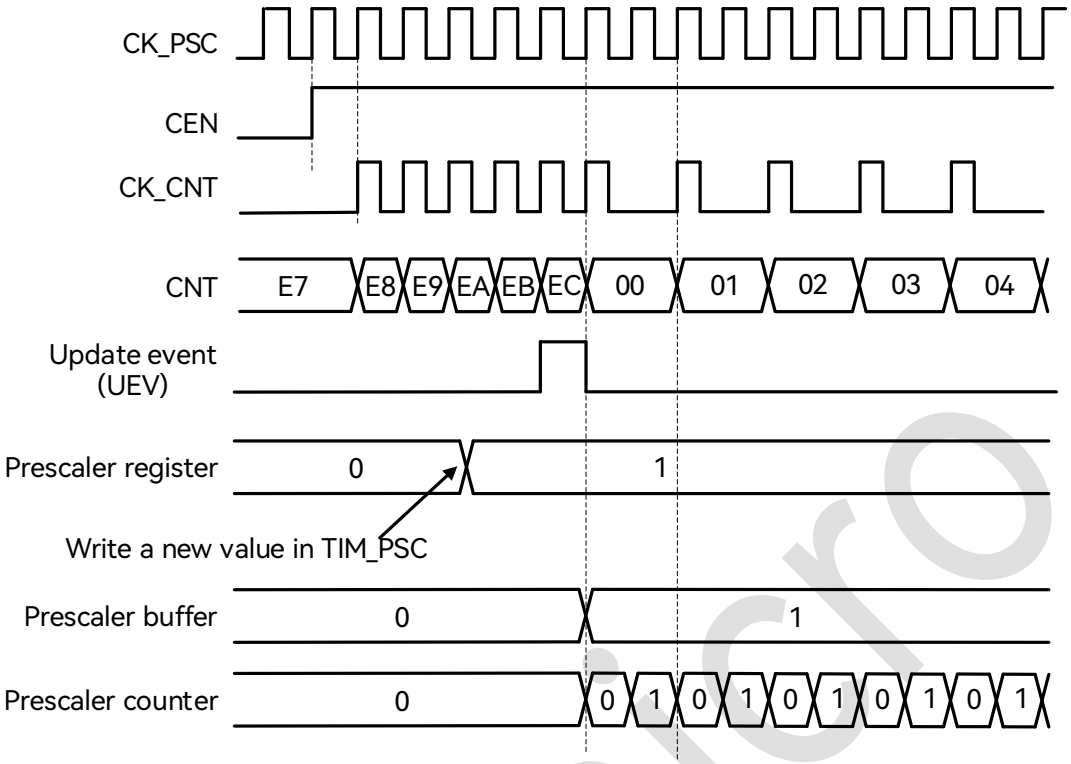

Figure 18-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 2

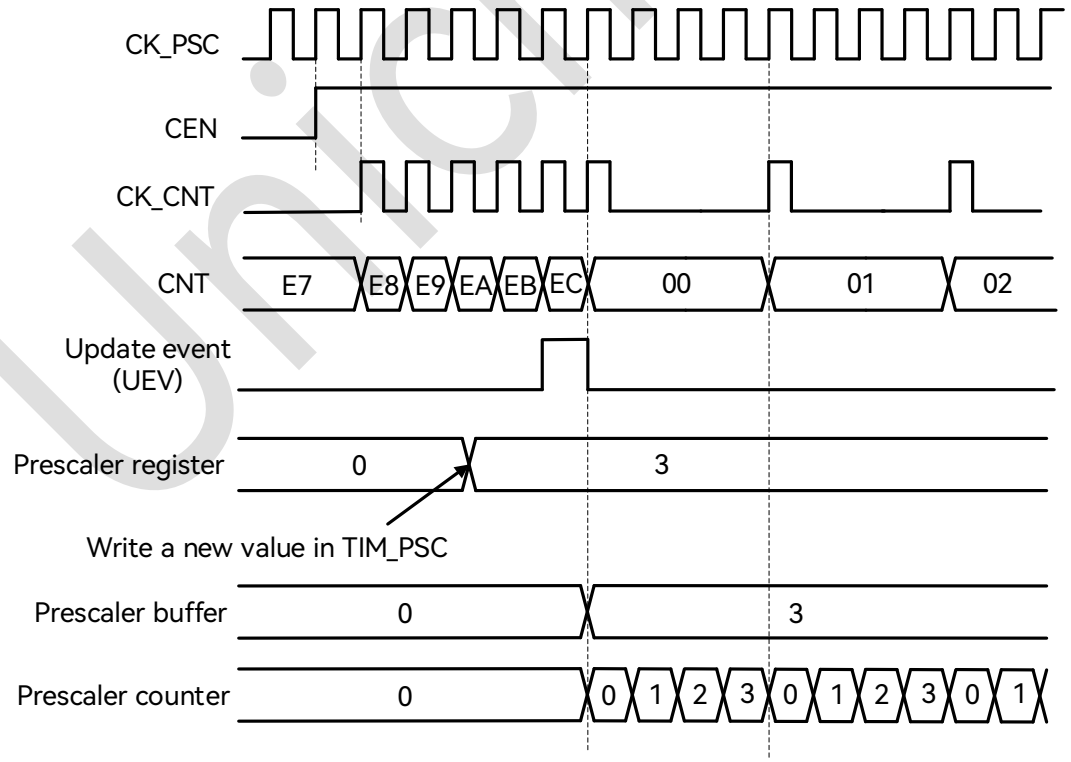

Figure 18-3: Counter Timing Diagram with Prescaler Division Changing from 1 to 4

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.2 Counter Operation Mode

The counter supports up-counting mode, down-counting mode and center-aligned mode.

18.6.2.1 Up-counting Mode

In up-counting mode, the counter counts from 0 to the auto-reload value, i.e. CNT = ARR, generating an overflow event, and then restarts counting from 0.

If the repetition counter is enabled, the counter repeats the above process a number of times (RCR + 1) as defined in RCR before generating an underflow event.

The software can directly trigger an update event by setting the UG bit in the register, at which time the CNT and the prescaler registers are automatically cleared. Whether setting the UG register triggers UIF (update interrupt flag) is determined by the setting of the URS register.

The update event can be disabled by setting the UDIS bit in the register to avoid updating the shadow register while writing new values in the preload registers.

When an update event occurs, the following registers are updated and the UIF bit is set:

- The repetition counter register RCR is reloaded with the content of TIM_RCR register.
- The auto-reload shadow register ARR is reloaded with the content of TIM_ARR register.
- The prescaler shadow register PSC is reloaded with the content of TIM_PSC register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

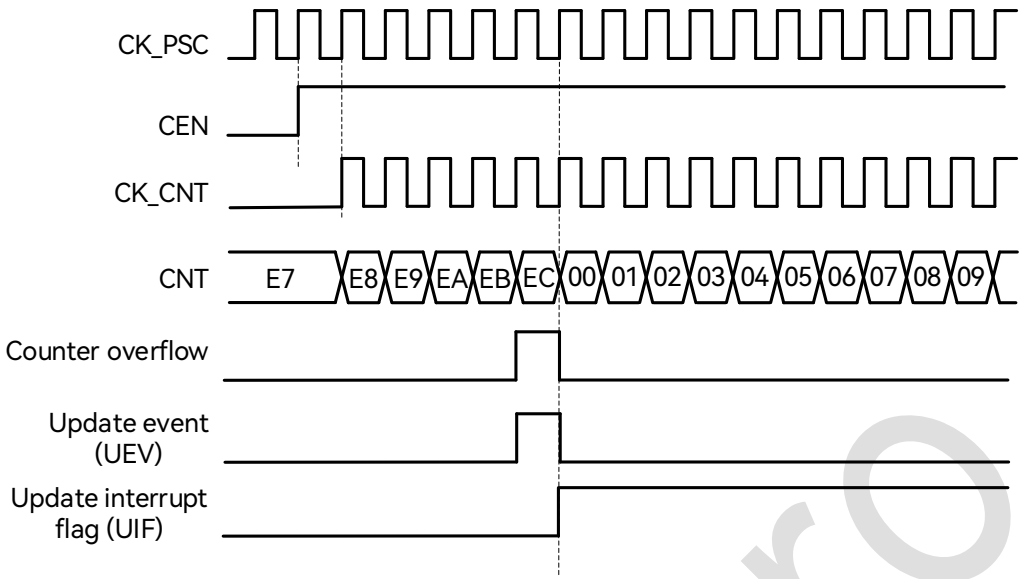

Figure 18-4: Up-counting Waveform Diagram, Internal Clock not Divided

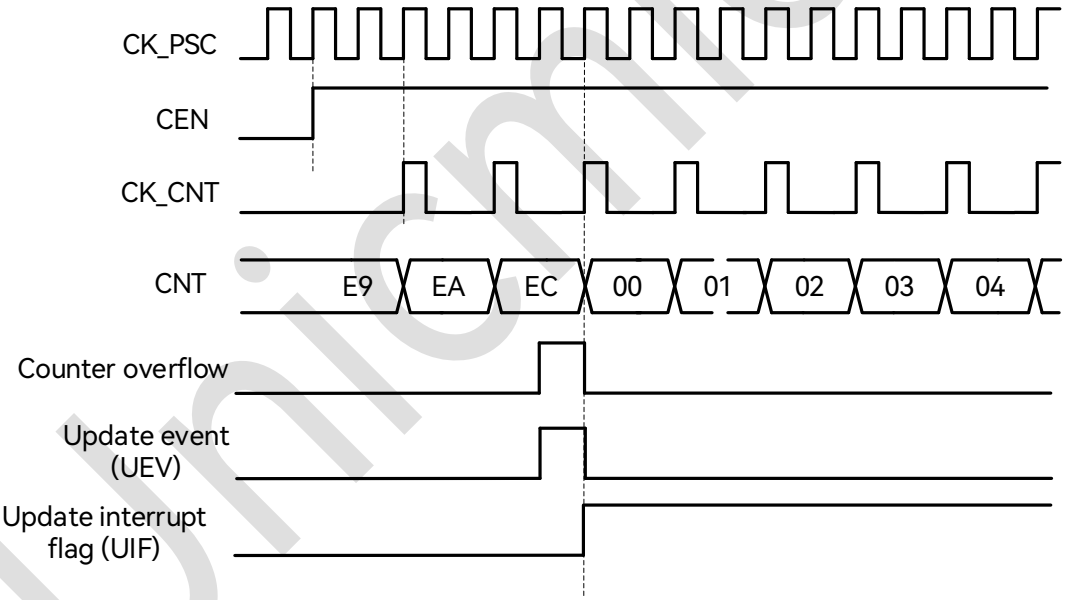

Figure 18-5: Up-counting Waveform Diagram, Internal Clock Divided by 2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

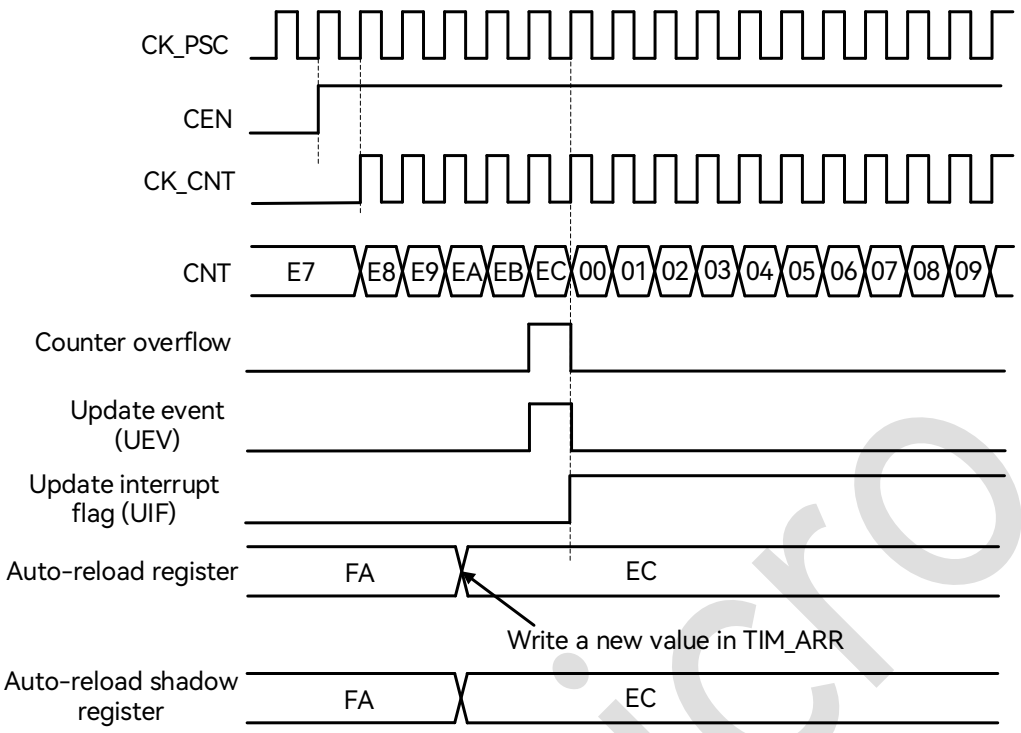

Figure 18-6: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded)

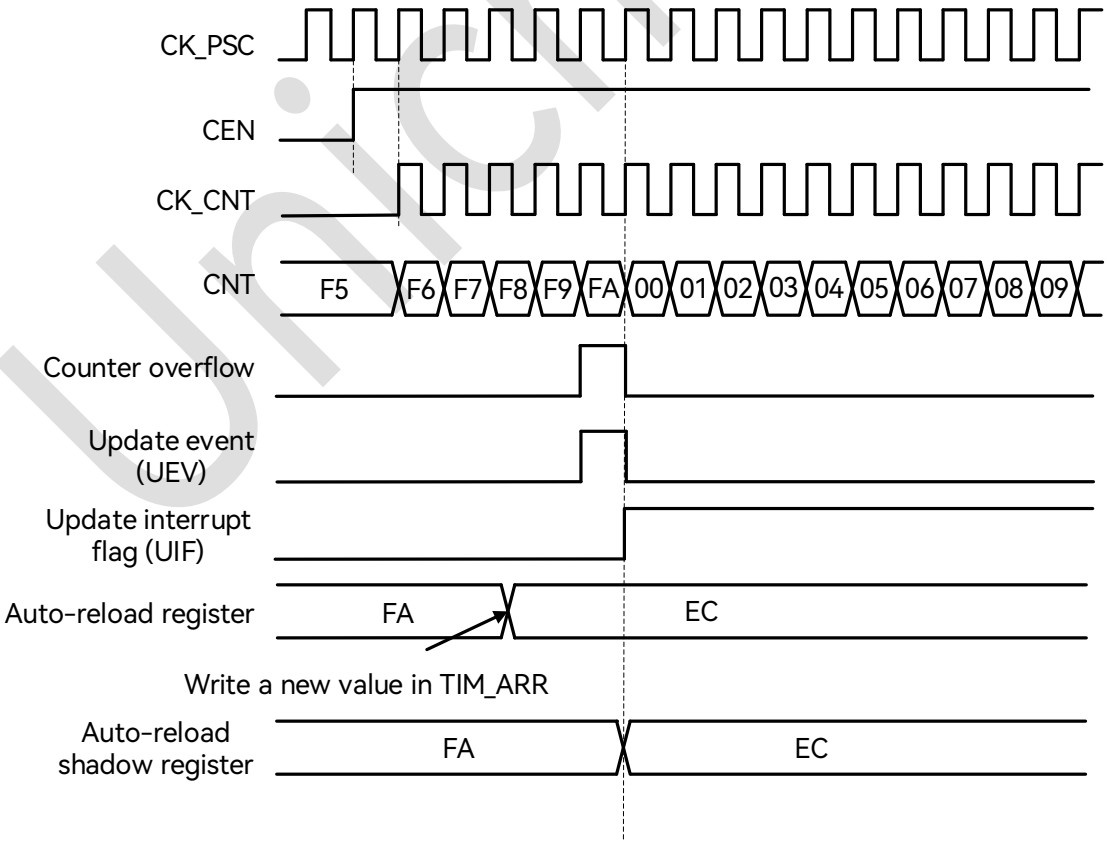

Figure 18-7: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.2.2 Down-counting Mode

In down-counting mode, the counter counts from the auto-reload value down to 0, generating an underflow event, and then restarts counting from the auto-reload value.

If the repetition counter is enabled, the counter repeats the above process a number of times (RCR + 1) as defined in RCR before generating an underflow event.

The software can directly trigger an update event by setting the UG bit in the register, at which time the CNT and the prescaler registers are automatically cleared. Whether setting the UG register triggers UIF (update interrupt flag) is determined by the setting of the URS register.

The update event can be disabled by setting the UDIS bit in the register to avoid updating the shadow register while writing new values in the preload registers.

When an update event occurs, the following registers are updated and the UIF bit is set:

- The repetition counter register RCR is reloaded with the content of TIM_RCR register.
- The auto-reload shadow register ARR is reloaded with the content of TIM_ARR register.
- The prescaler shadow register PSC is reloaded with the content of TIM_PSC register.

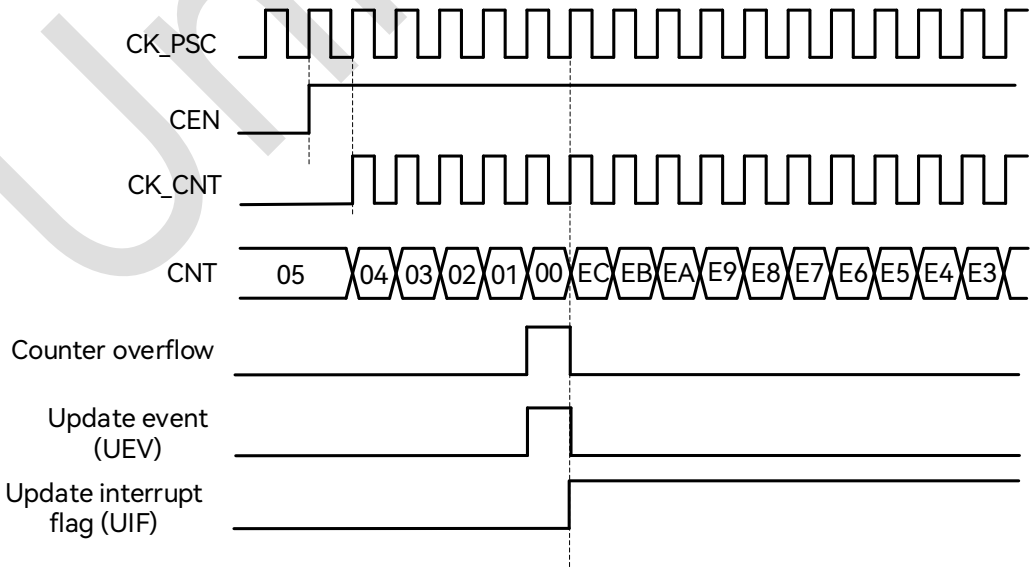

Figure 18-8: Down-counting Waveform Diagram, Internal Clock not Divided

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

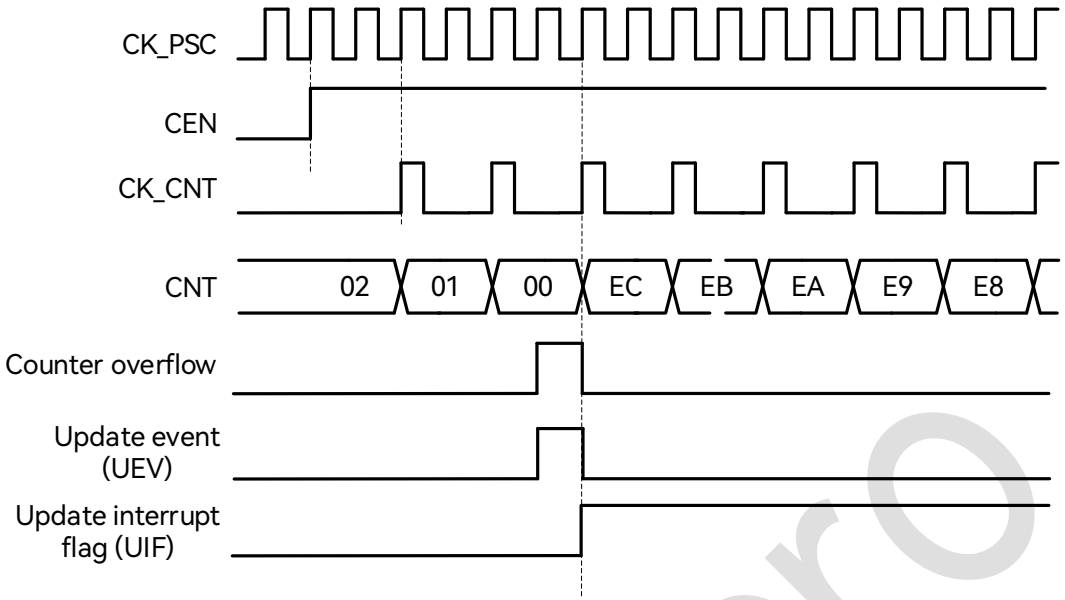

Figure 18-9: Down-counting Waveform Diagram, Internal Clock Divided by 2

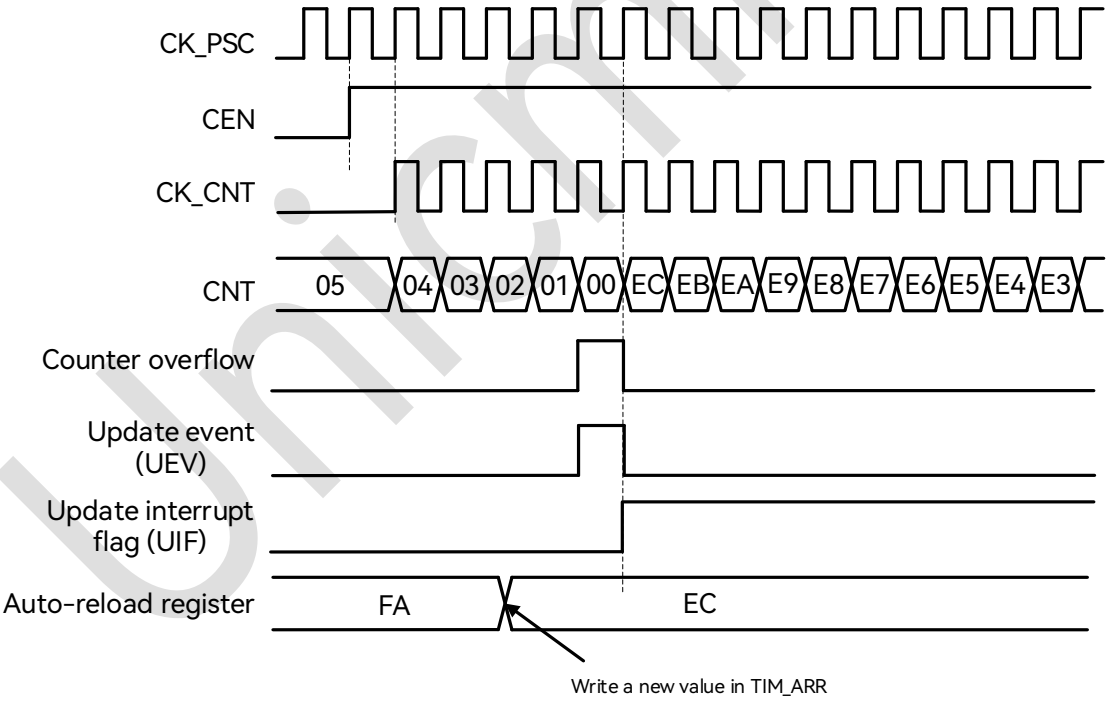

Figure 18-10: Down-counting Waveform Diagram, Internal Clock Divided by 2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

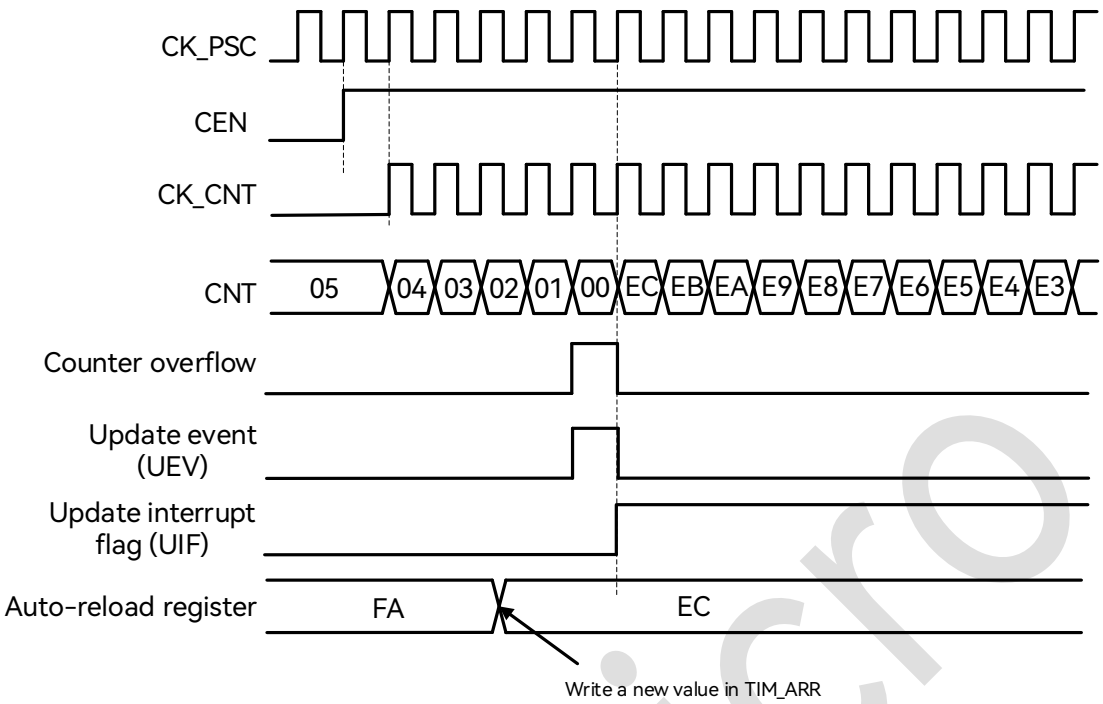

Figure 18-11: Down-counting Waveform Diagram, Update Event when Repetition Counter is not Used

18.6.2.3 Center-aligned Counting Mode

In center-aligned mode, the counter counts from 0 to the auto-reload value - 1 and generates a counter overflow event, then counts from the auto-reload value down to 1 and generates a counter underflow event, and then restarts counting from 0.

The CMS[1:0] bits are used for enabling the center-aligned mode and selecting the output compare mode herein. The center-aligned mode is active when CMS! = 00. The output compare interrupt flag of channels configured in output is set when: the counter counts down (CMS = 01), the counter counts up (CMS = 10), the counter counts up and down (CMS = 11).

In this mode, the DIR direction bit in the register cannot be written by software. It is updated by hardware and gives the current direction of the counter.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The counter updates the shadow registers of ARR, PSC and RCR at each counter overflow and underflow.

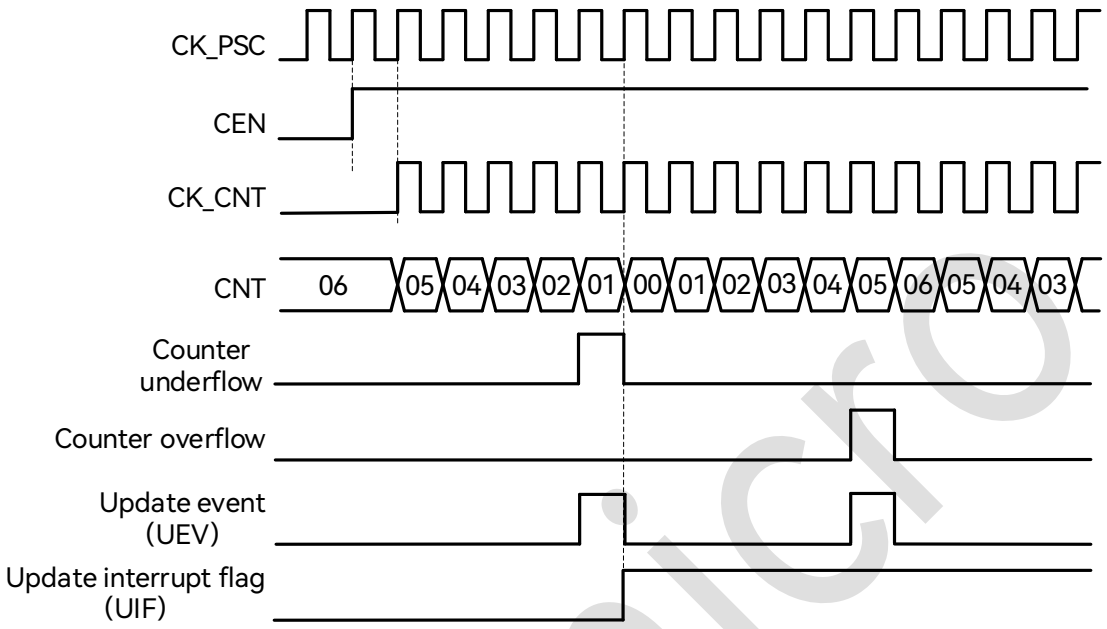

Figure 18-12: Center-aligned Counter Timing Diagram, TIM_PCS = 0, TIM_ARR = 0x6

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

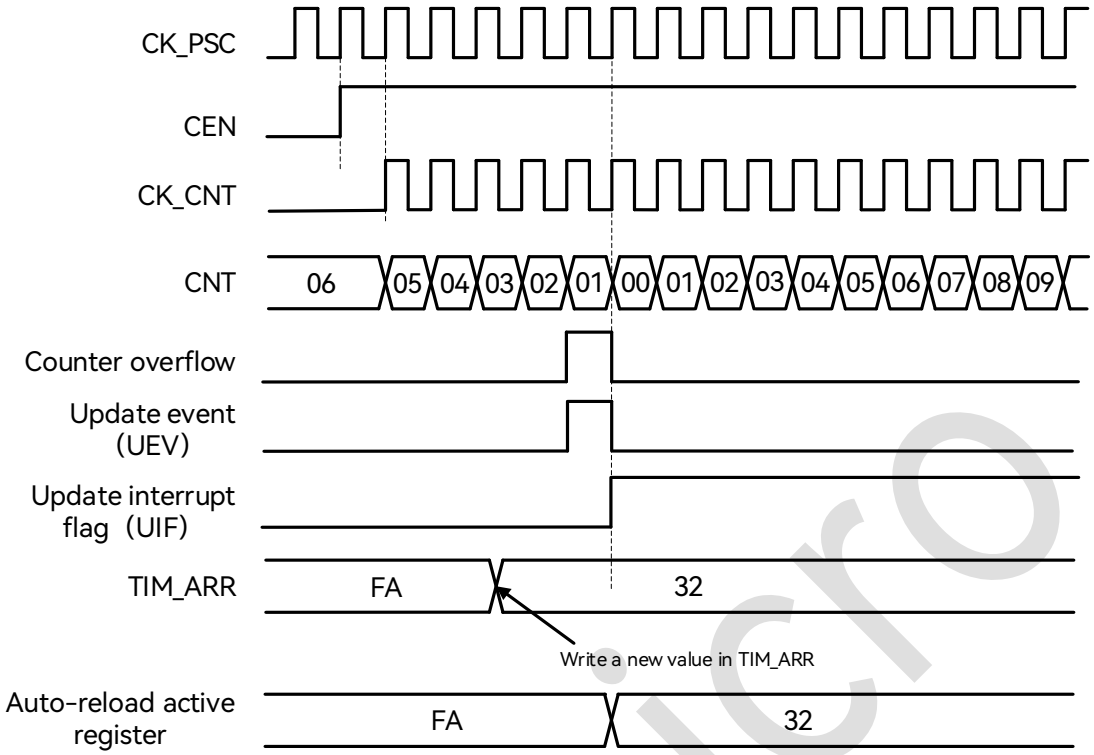

Figure 18-13: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Underflow)

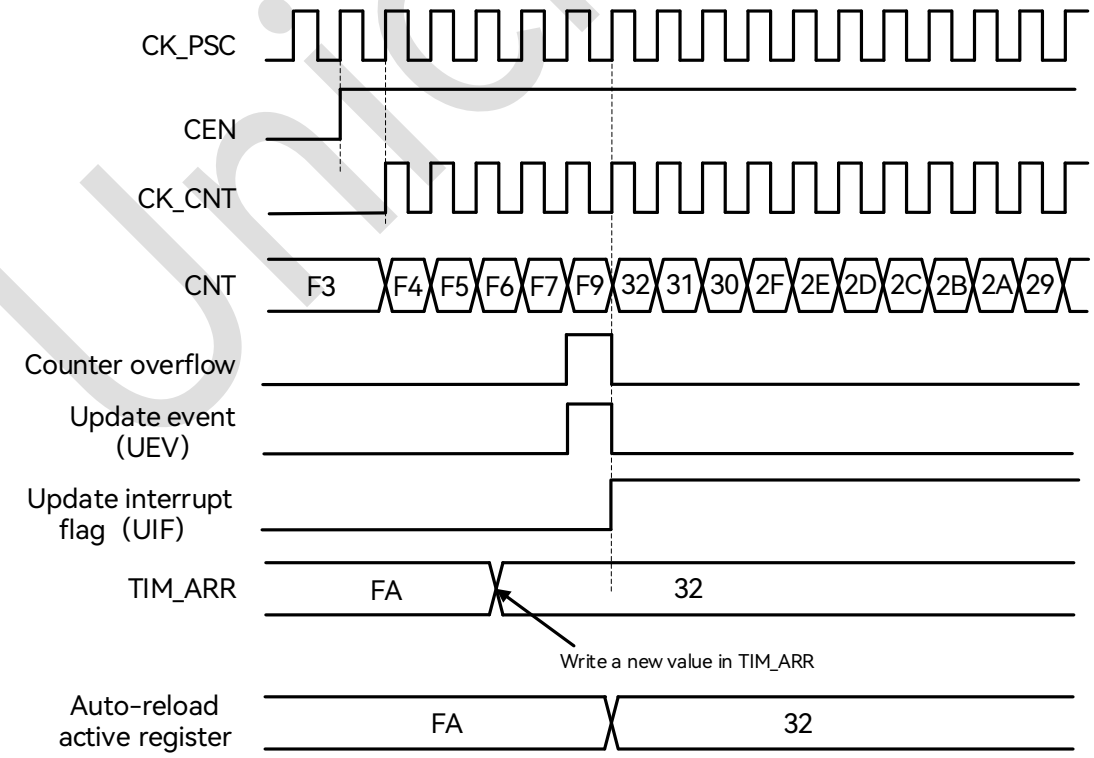

Figure 18-14: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Overflow)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.3 Repetition Counter

The update event is generated only when the repetition counter has reached zero at counter overflow or underflow. This means that the data are transferred from the preload registers of ARR, PSC and CCR (compare/capture registers in output compare mode) to the shadow registers every N + 1 counter overflows/underflows, where N is the value in the repetition counter register RCR.

The repetition counter is decremented:

- at each counter overflow in up-counting mode;
- at each counter underflow in down-counting mode;
- at each counter overflow / underflow in center-aligned mode.

Note: When the update event is generated by software or by hardware through the slave mode controller, it occurs immediately whatever the value of RCR is and the repetition counter is reloaded with the content of the TIM_RCR register.

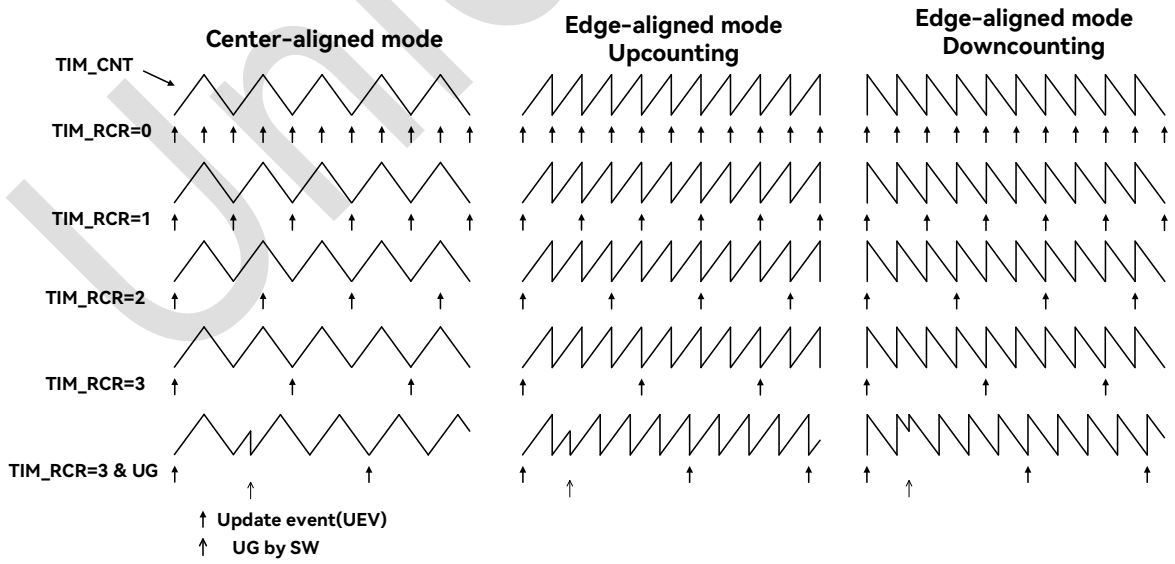

Figure 18-15: Examples of Updating Rate in Different Modes and TIM_RCR Register Setting

Unicmicro

18.6.4 Preload Register

The following functional registers support the preload function:

- Auto-reload register TIM_ARR
- Prescaler register TIM_PSC (preload function cannot be disabled)
- Channel control register TIM_CCR
- CCxE and CCxNE control register
- OCxM control register

The preload function can be enabled or disabled by software for all of the above registers except TIM_PSC.

Registers with preload function contain two sets of physical entities:

- Shadow register: the register being used by the actual timer
- Preload register: the register accessible to software

When the preload function is disabled, the register with preload function has the following characteristics:

- The preload register can be accessed and overwritten by software in real time.
- The shadow register is updated synchronously with the preload register.

If the preload function is enabled, then:

- All software operations access the preload register.
- At the occurrence of update event, the content of all preload registers will be synchronously transferred to the corresponding shadow registers.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.5 Counter Clock

The counter clock can be provided by the following clock sources:

- Internal clock: Timerx_clk
- External clock mode 1: External input pin TIx
- External clock mode 2: External trigger input ETR
- Internal trigger inputs (ITRx): using the trigger output (TRGO) of one timer as the counter clock

18.6.5.1 Internal Clock Source

If the slave mode controller is disabled (SMS = 000), then the CEN, DIR and UG bits are controlled by software. After the UG bit is set and the update signal is synchronized by CLK_PSC, the counter value is reinitialized.

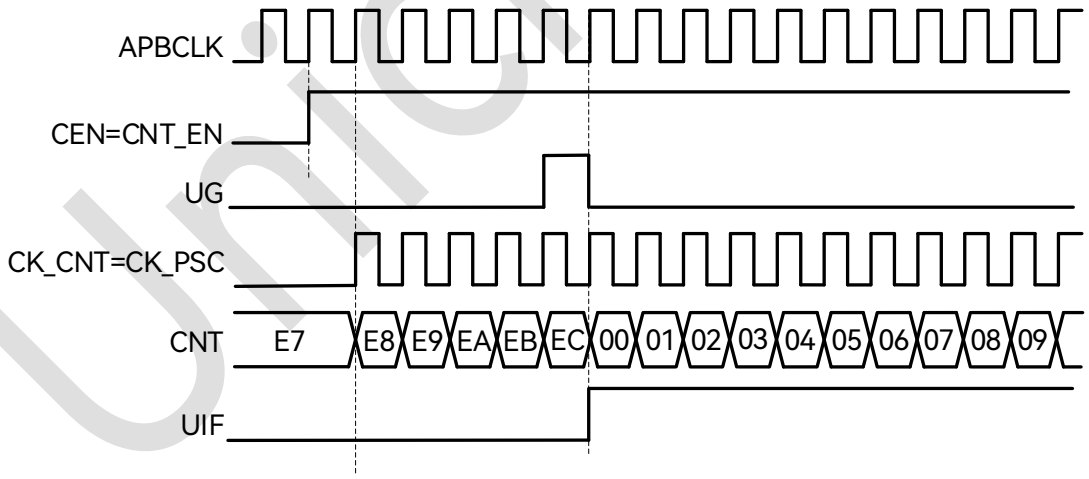

Figure 18-16: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.5.2 External Clock Source Mode 1

In this mode, the external pin input signal is directly used as the counter clock when SMS = 111, and the counter can count at either rising or falling edge on a selected input.

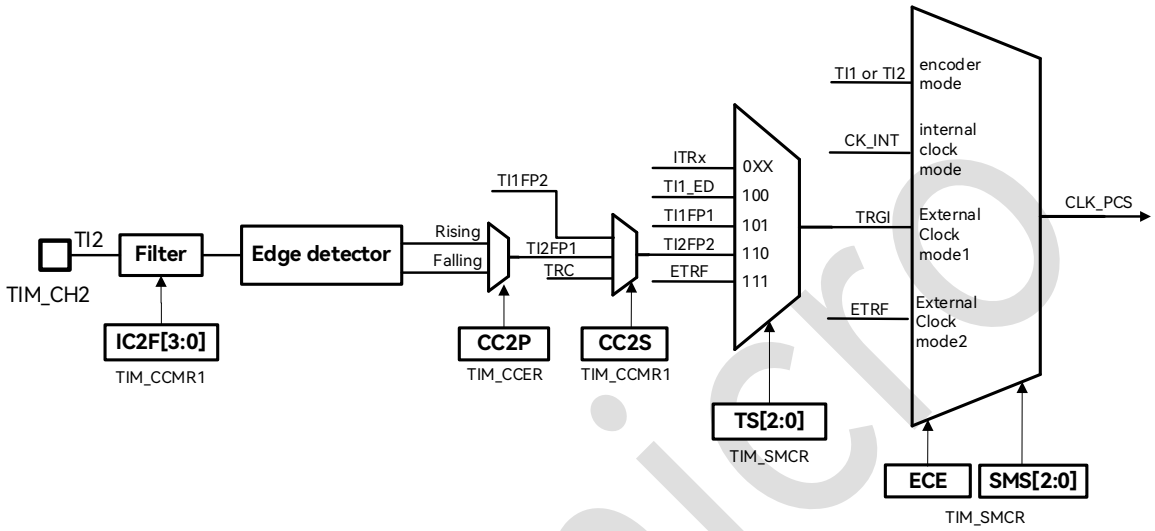

Figure 18-17: External Clock Connection Diagram

The external input signal will be synchronized with the internal clock before triggering the counter counting, and the TIF flag will be set by the valid edge on the input.

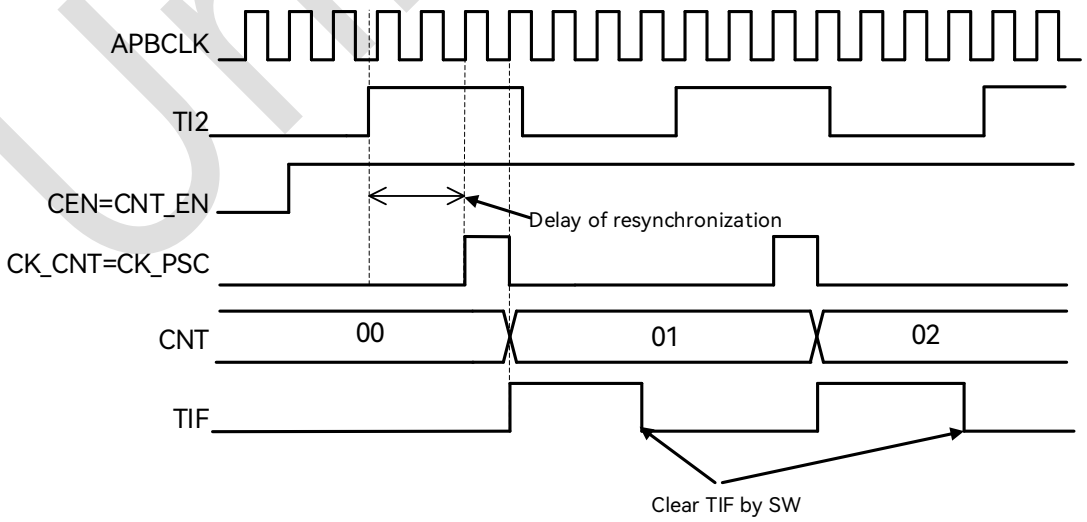

Figure 18-18: Timing Diagram in External Clock Source Mode 1

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When counting with an external clock, the internal clock (timerx_clk) shall still be enabled so that TIM can use timerx_clk to synchronize and filter the external input clock. In external clock mode 1, the external input clock is first subject to filtering and edge selection to obtain a valid counting edge, which is input to the prescaler module as the valid operating clock (CLK_PSC).

The external clock synchronization adopts a simple two-stage flip-flop structure, so in order to avoid metastability, the external input clock width is required to be at least 2 timerx_clk cycles.

In this mode, only the inputs of channels 1 and 2 can be used as clock inputs, and the required configuration is as follows:

- In GPIO module, configure the corresponding pin as TIM_CH2.
- Disable the channel by setting TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
- Select the input channel by setting TIM_CCMR1[9:8] = 01, with IC2 mapped on TI2.
- Select the active counting edge to be rising or falling edge by setting TIM_CCER[5] = 0.
- Configure the input filter duration by writing the IC2F[3:0] bits in the TIM_CCMR1 register (if no filter is required, keep IC2F = 0000).
- Enable the external clock source mode 1 by setting TIM_SMCR[2:0] = 111.
- Select TI2 as the trigger input source by setting TIM_SMCR[6:4] = 110.
- Enable the channel by setting TIM_CCER[4] = 1.
- Enable the counter by setting TIM_CR1[0] = 1.

The following diagram shows an example of typical external clock source mode 1:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

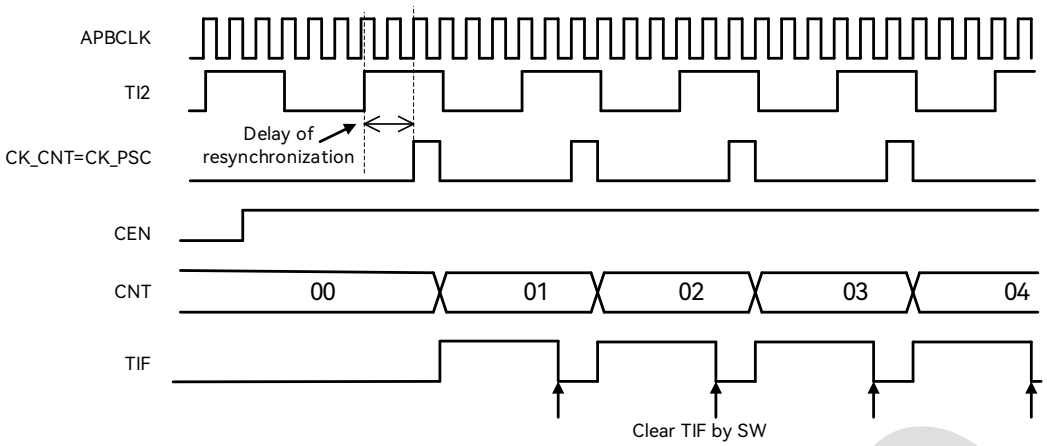

Figure 18-19: Timing Diagram in External Clock Source Mode 1

18.6.5.3 External Clock Source Mode 2

In this mode, the counter counts at either rising edge or falling edge (double-edge not supported) on the external trigger input TIM_ETR.

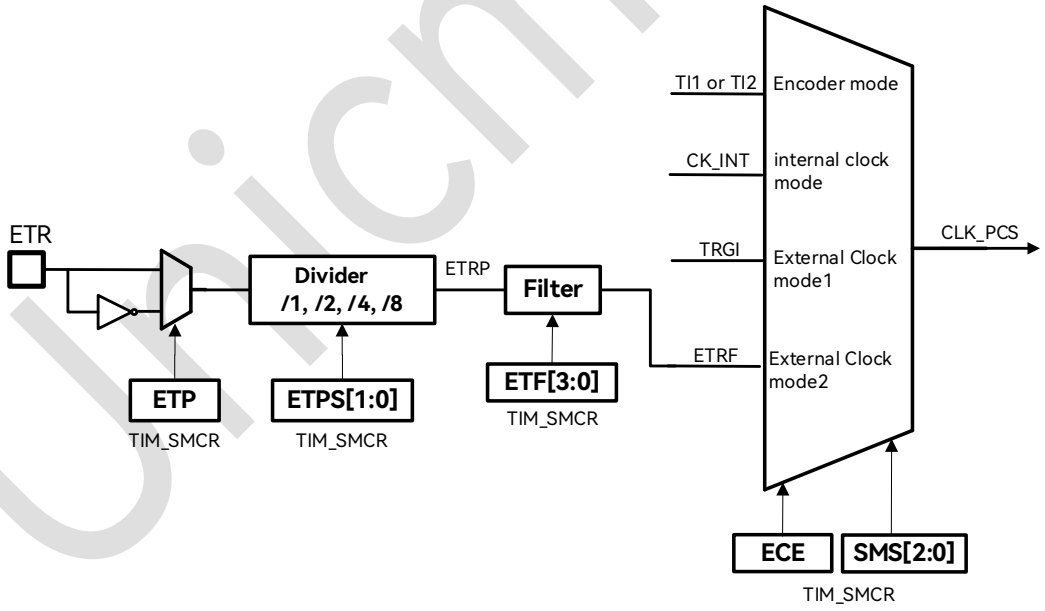

Figure 18-20: External Trigger Input Block Diagram

The following diagram shows the counter counting each 2 rising edges on ETR. The delay between the rising edge of ETR and the actual clock of the counter is due to the resynchronization of the internal clock.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

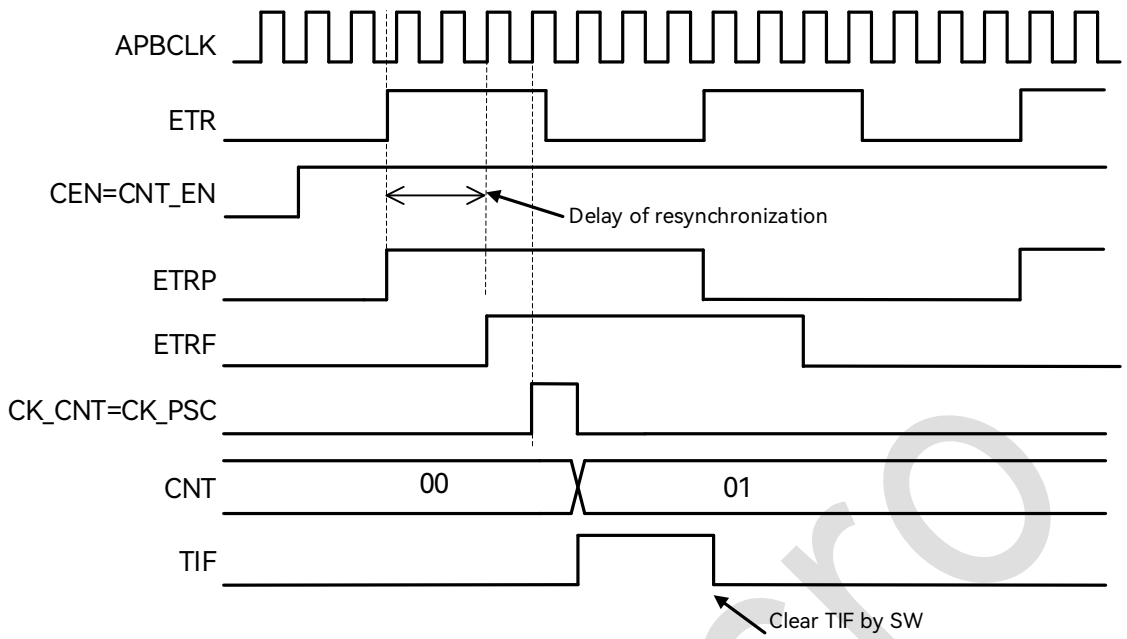

Figure 18-21: Timing 1 Diagram in External Clock Source Mode 2

The main difference from external clock source mode 1 is that the ETR input is directly divided and then filtered to generate CK_PSC clock, which mens that the application scenarios where the ETR input frequency is higher than timerx_clk can be supported, in which case the ETR input shall be pre-divided first before it is used to drive the counter.

The configuration required for this mode is as follows:

- In GPIO module, configure the corresponding pin as TIM_ETR.
- Select the ETP edge by setting TIM_SMCR[15] = 0.
- Set the ETR division ratio by writing TIM_SMCR.ETPS[1:0] = 01.
- Configure the input filter duration by setting TIM_SMCR.ETF[3:0] = 0000.
- Set the ECE register and enable the external clock source mode 2 by setting TIM_SMCR[14] = 1 and TIM_SMCR[2:0] = 000.
- Enable the counter by setting TIM_CR1[0] = 1.

The following diagram shows an example of typical external clock source mode 2:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

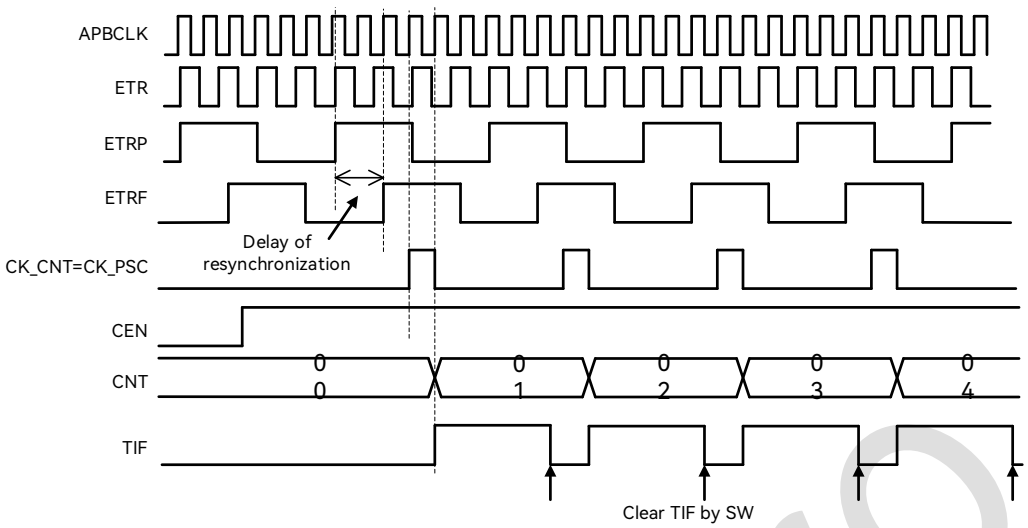

Figure 18-22: Timing 2 Diagram in External Clock Source Mode 2

In external clock source mode 2, TIM can still be configured as slave mode: For example, ETR input is used for counting while TRGO of another timer is used as the trigger signal, and the reset counter restarts counting at the arrival of trigger event.

18.6.6 Internal Trigger Signal (ITRx)

TIM supports four internal trigger inputs, which can be used for counting trigger or internal signal capture. For internal signal capture, it is required to configure TS to 000−011 for selecting ITR0−ITR3, and configure CCxS to 11 for selecting TRC (signal selected by four ITRx inputs) as the capture signal.

Each ITR input supports 4 internal signal extensions configured by the ITRxSEL register.

18.6.7 Capture/Compare Channels

TIM consists of 4 capture/compare channels, each of which is built around a capture/compare register CCR (including a shadow register), an input stage for capture and an output stage (with comparator and output control).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The input stage samples the corresponding TIx input to generate a filtered signal TIxF. Then, an edge detection with polarity selection generates a signal (TIxFPx), which can be used as trigger input for counting or as the capture command and is prescaled before being captured.

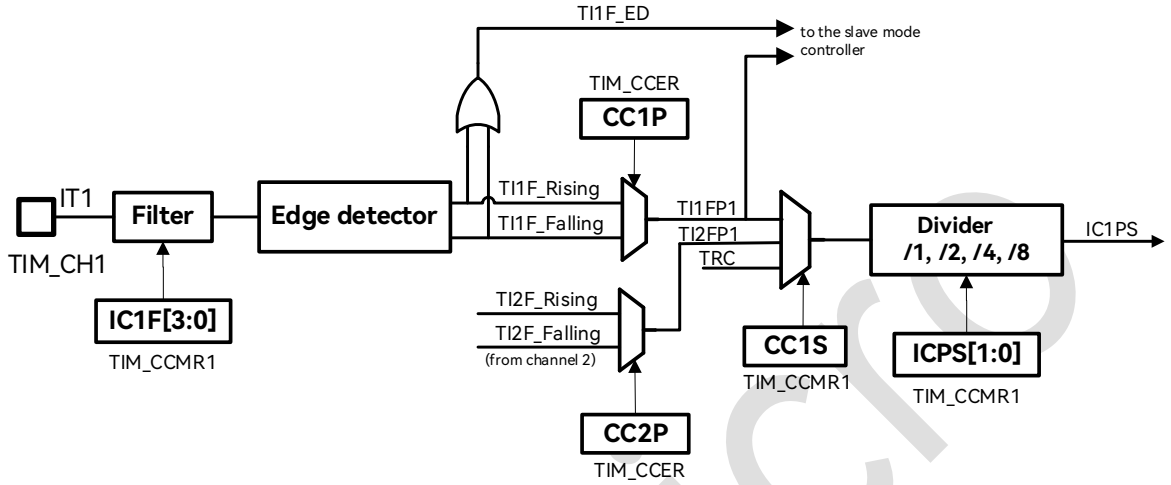

Figure 18-23: Capture/Compare Channel (Channel 1 Input Stage)

The output stage generates an output reference signal OCxREF, which is fixed to be active high and acts as the reference input to the final output circuit. Wherein, channels 1–3 support complementary output and dead-time insertion, while channel 4 is relatively simple and does not support complementary output.

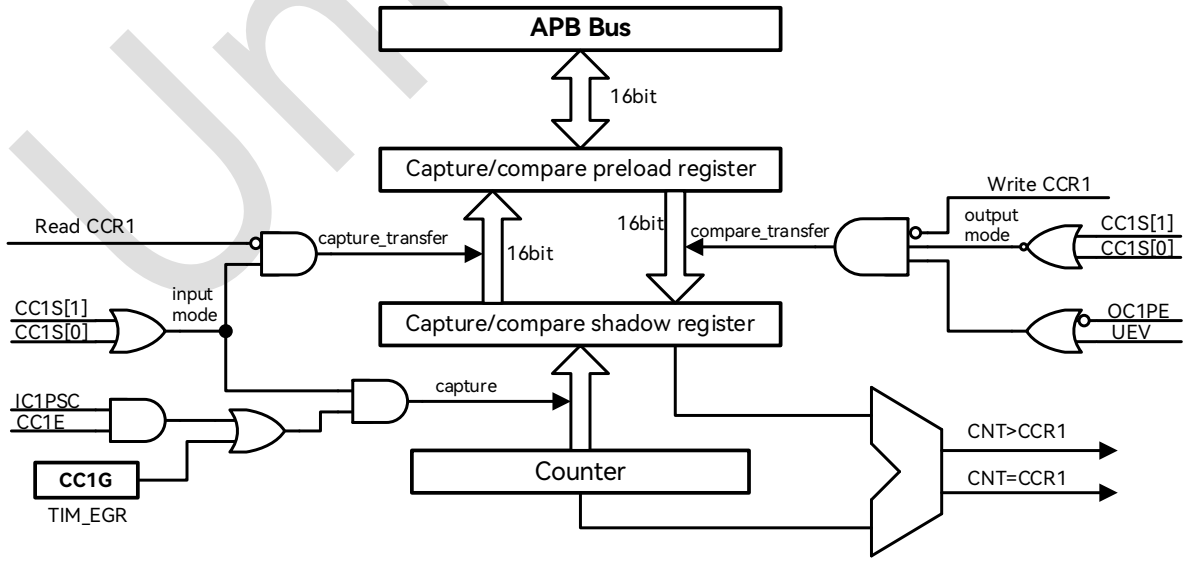

Figure 18-24: Capture/Compare Channel 1 Main Circuit

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

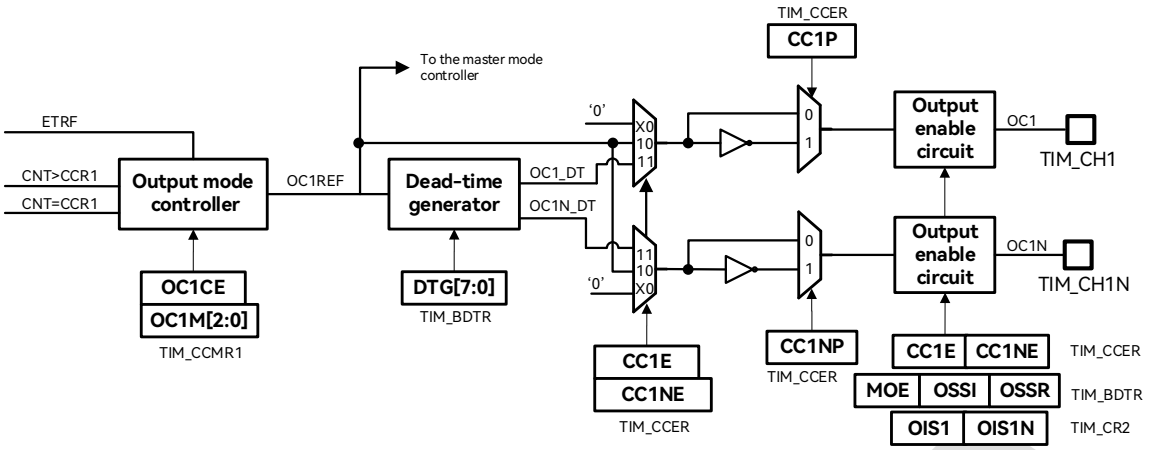

Figure 18-25: Output Stage of Capture/Compare Channel (Channels 1–3)

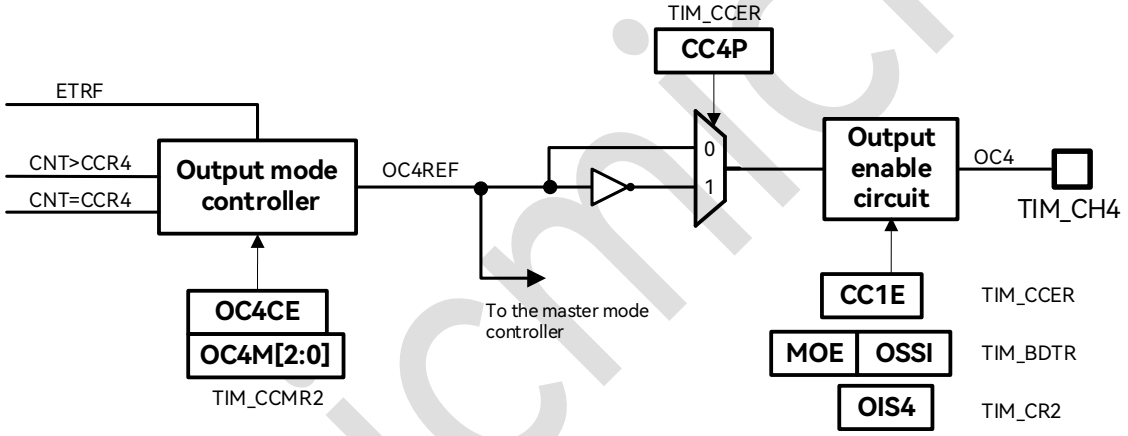

Figure 18-26: Output Stage of Capture/Compare Channel (Channel 4)

The capture/compare block is made of one preload register and one shadow register. Write and read always access the preload register. In capture mode, the capture value is saved in the shadow register and copied into the preload register. In compare mode, the content of the preload register is copied into the shadow register for comparison with the counter.

18.6.8 Input Capture Mode

When the expected level transition is detected by the ICx signal, a capture is triggered, and the current counter value is latched into CCR. At the same time, the CCxIF interrupt flag is set and a corresponding interrupt or a DMA request can be triggered. If a capture occurs while the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

CCxIF flag is already high, then the over-capture flag CCxOF is set (the last capture value in CCR is overwritten). CCxIF can be cleared by software or automatically cleared by reading the CCR register. CCxOF can be cleared by software writing it to 1.

The input capture of PWM signals can be realized through the cooperation of two or more channels. For example, to calculate the period and duty cycle of an input signal, input the signal from TI1 pin, and take the rising edge and falling edge of the filtered signal inside the chip to obtain TI1FP1 and TI1FP2 respectively. Input TI1FP1 into capture channel 1 and TI1FP2 into channel 2 to realize captures of input signals at rising edge by channel 1 and at falling edge by channel 2. After the capture interrupt occurs periodically, the software can calculate the period and duty cycle of the input signal through the values of CCR1 and CCR2 registers.

To capture the counter value to the TIM_CCR1 register on the rising edge of the TI1 input, the configuration steps are as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01, with IC1 mapped on TI1.
4. Select the active counting edge to be rising edge or falling edge by setting TIM_CCER[1].
5. Configure the input filter duration by setting the IC1F[3:0] bits in the TIM_CCMR1 register.
6. Configure the input prescaler by setting the IC1PS[1:0] bits in the TIM_CCMR1 register.
7. Enable the channel by setting TIM_CCER[0] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

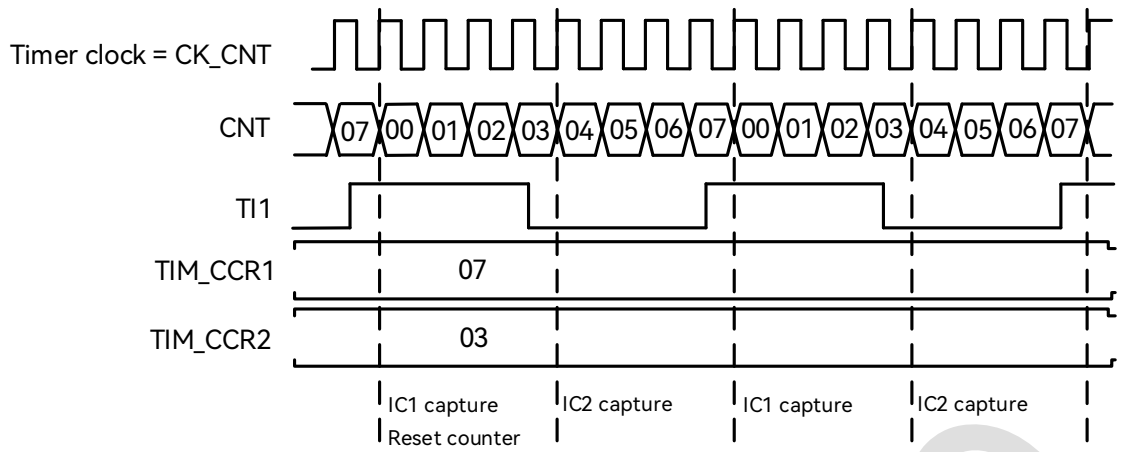

Figure 18-27: PWM Input Capture Mode Timing Diagram

The following settings are required for PWM input capture:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 and TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel, with the two signals IC1 and IC2 mapped on the same TI1 input, and configure TIM_CCMR1[1:0] = 01 and TIM_CCMR1[9:8] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0 and TIM_CCER[5] = 1, with the two signals IC1 and IC2 active on edges with opposite polarities.
5. Configure the input filter duration by setting the IC1F[3:0] and IC2F[3:0] bits in the TIM_CCMR1 register.
6. Configure the input prescaler by setting the IC1PS[1:0] and IC2PS[1:0] bits in the TIM_CCMR1 register.
7. Select the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
8. Configure the slave mode controller to reset mode by setting TIM_SMCR.SMS[2:0] = 100.
9. Enable the channel by setting TIM_CCER[0] = 1 and TIM_CCER[4] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.9 Forced Output Mode

In output compare mode, the OCxREF signal can be forced to active or inactive level directly by software, independently of any comparison between the CCR and the counter.

The OCxREF signal can be forced to be active (OCxREF is always active high) by writing OCxM = 101, and forced to be inactive (low level) by writing OCxM = 100. Anyway, the comparison between CCR and the counter is still performed.

18.6.10 Output Compare Mode

In output compare mode, when a match is found between the capture/compare register CCR and the counter, the OCxREF can be set to be active, inactive or to toggle on match. At the same time, the interrupt flag is also set and DMA requests can be sent.

The output compare can also be used to output a pulse signal of a specific width (in one-pulse mode).

Procedure:

1. Select the counter clock (internal, external, prescaler).
2. Write the desired data to the ARR and CCR registers.
3. Set the interrupt enable bit and DMA enable bit as required.
4. Select the output mode.
5. Enable the counter.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

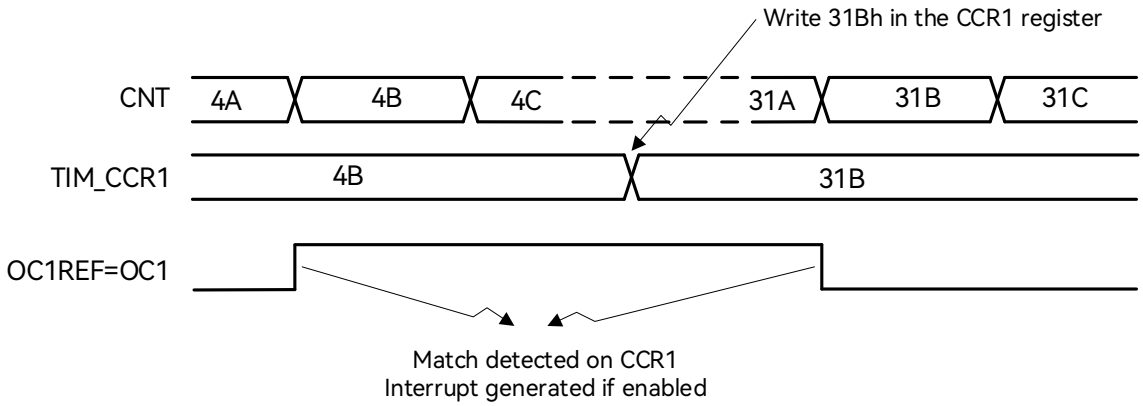

Figure 18-28: Output Compare Mode, Toggle on OC1

The CCR register can be updated at any time by software to control the output waveform, provided that the preload register is not enabled. Otherwise, the CCR shadow register is only updated with the content of the preload register at the next update event.

18.6.11 PWM Output

PWM mode allows you to generate a pulse width modulation signal with a frequency determined by the value of the ARR register and a duty cycle determined by the value of the CCR register.

The polarity of the output signal is software programmable using the CCxP bit in the register. In PWM mode, CNT and CCR registers are always compared. The timer is able to generate PWM in edge-aligned mode or center-aligned mode.

18.6.11.1 PWM Edge-aligned Mode

In up-counting mode, when it is configured in PWM mode 1, the OCxREF signal is high as long as CNT < CCR, otherwise it is low. And OCxREF will be held at 1 if CCR > ARR and held at 0 if CCR is 0.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

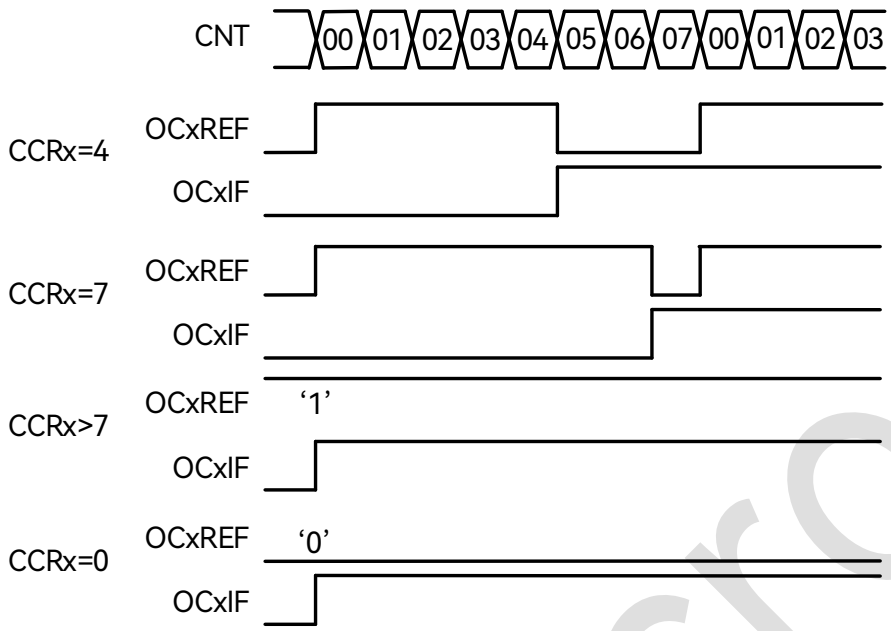

Figure 18-29: Edge-aligned PWM Waveform (ARR = 7)

In down-counting mode, the definition of OCxREF level is the same as that in up-counting mode.

18.6.11.2 PWM Center-aligned Mode

The definition of OCxREF level is the same as that in edge-aligned mode. The figure below is an example.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

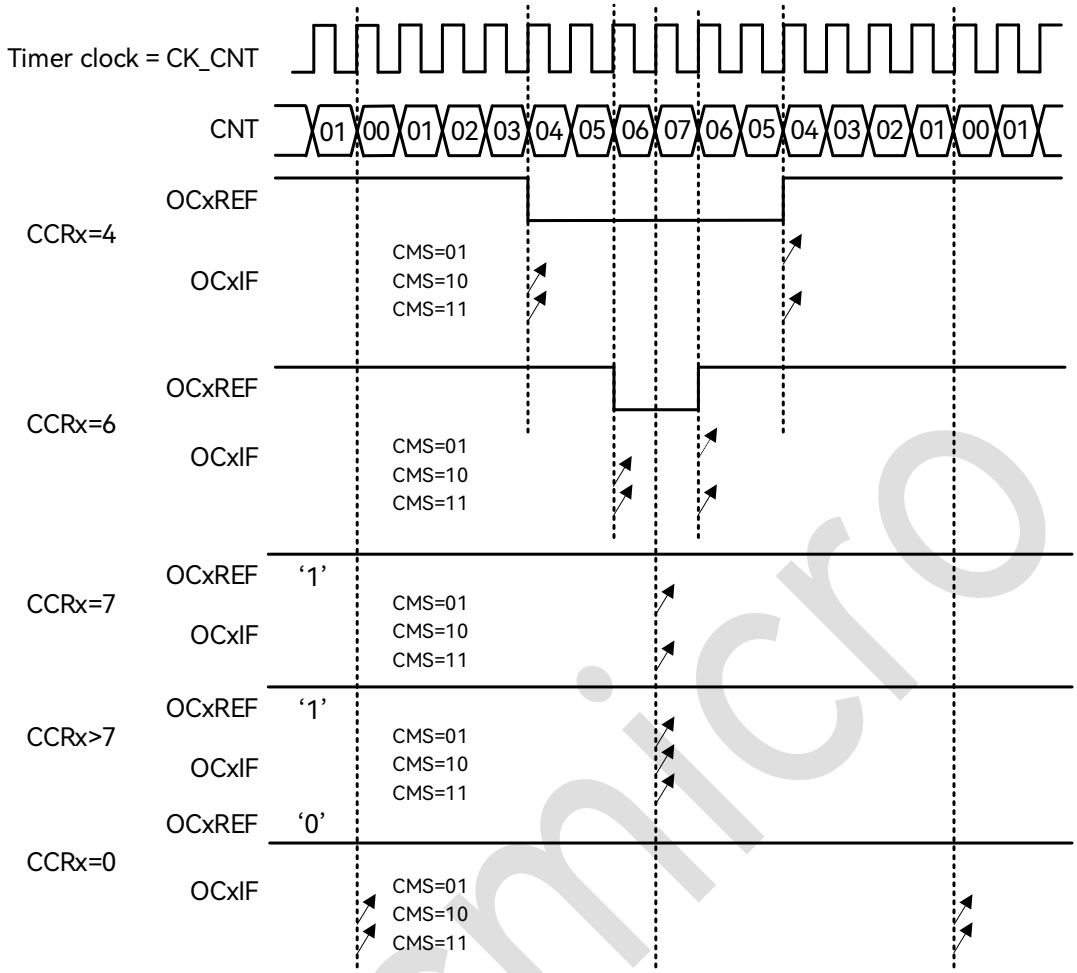

Figure 18-30: Center-aligned PWM Waveform (ARR = 7)

When start counting in center-aligned mode, the initial counting direction is determined by the DIR bit in the register, and in the subsequent process, the DIR bit is directly controlled by hardware. The safest way to use center-aligned mode is to generate an update by setting the UG bit in the register just before starting the counter and not to overwrite the counter while it is running.

18.6.12 Complementary Output and Dead-time Insertion

Channels 1–3 of TIM support complementary output and dead-time insertion. The DTG[7:0] bits in the register are used to set the dead-time delay (valid for all channels at the same time). The output signal OCx is in phase with the reference signal OCxREF, with the rising edge delayed

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

relative to the reference rising edge. The output signal OCxN is in the opposite phase with the reference signal OCxREF, with the rising edge delayed relative to the reference falling edge.

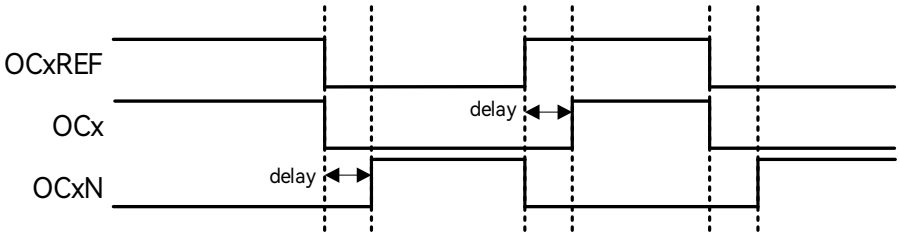

Figure 18-31: Waveform of Complementary Output with Dead-time Insertion

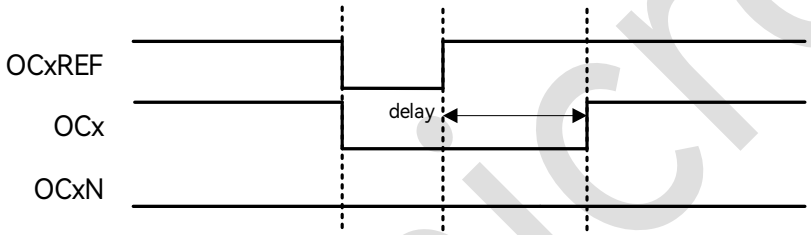

Figure 18-32: Dead-time Waveform with Delay Greater than Negative Pulse

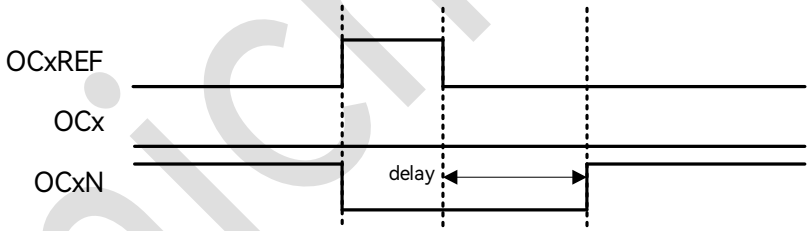

Figure 18-33: Dead-time Waveform with Delay Greater than Positive Pulse

18.6.13 Break Function

The break function can be activated using an external break signal or a clock fault signal.

When a break occurs:

- The output enable register is cleared asynchronously, and the output can be forced to inactive, idle or reset state (selected by the OSSI bit).
- Each output channel is driven with the level programmed in the OISx bit in the register.

Unicmicro

- When complementary outputs are enabled, the outputs are asynchronously set to the inactive state and reset state, and the dead-time insertion circuit starts to work, driving the output to the level defined by OISx and OIXN after the dead time.
- The break status flag is set. An interrupt or DMA request can be triggered according to the configuration.
- If the automatic output is enabled (AOE = 1), the MOE bit will be automatically set at the next update event; otherwise, MOE will remain 0 until it is reset by software.

Note: The BRK signal is acting on level, so MOE cannot be set while BRK remains active, and the break flag BIF cannot be cleared.

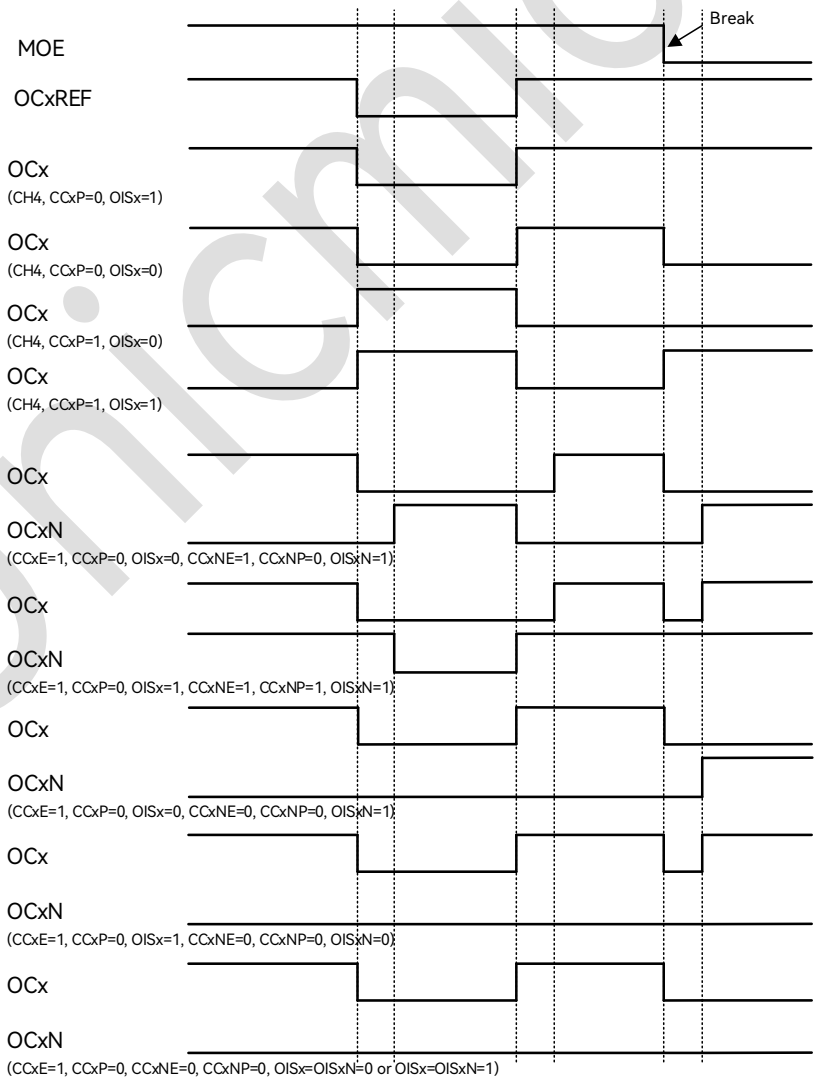

Figure 18-34: Various Output Behavior in Response to A Break Event

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.14 6-step PWM Generation

When complementary outputs are used on a channel, preload bits are available on the OCxM, CCxE and CCxNE bits. The preload bits are transferred to the shadow bits at the COM commutation event, thus one can program in advance the configuration for the next step and update the configuration of all the channels at the same time. COM can be generated by software by setting the COM bit in the TIM_EGR register or by hardware (on TRGI rising edge).

A flag is set when the COM event occurs, which can generate an interrupt or a DMA request.

The figure below describes the behavior of the OCx and OCxN outputs when a COM event occurs, in 3 different examples of programmed configurations.

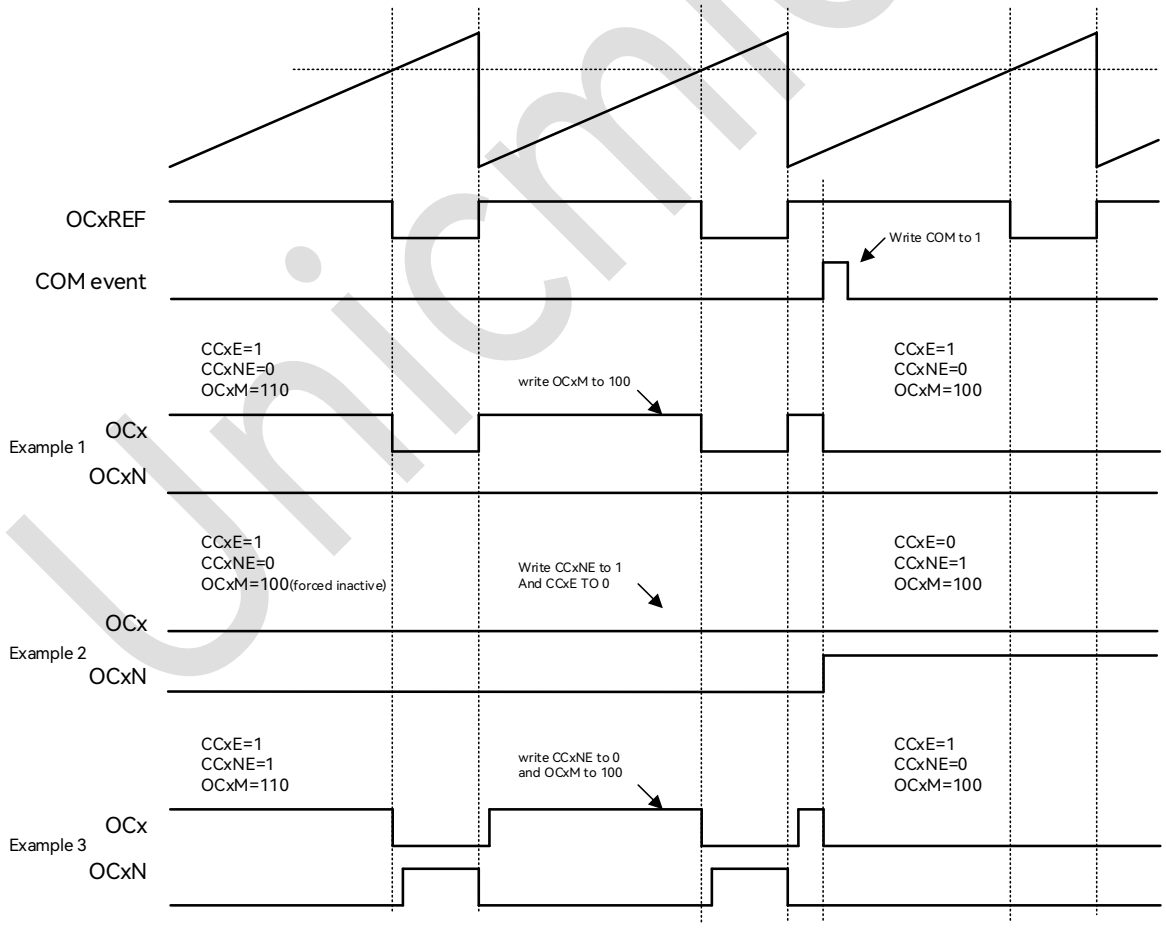

Figure 18-35: 6-step PWM Generation, COM Example (OSSR = 1)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.15 One-pulse Output Mode

One-pulse output mode is a particular case of the compare output mode, which allows the counter to generate a pulse with a programmable length after a programmable delay following the occurrence of an event.

Different from other output modes, the counter will stop automatically at the next update event. A pulse can be correctly generated only if the compare value is different from the counter initial value. In up-counting, it is required that CNT < CCR ≤ ARR; in down-counting, it is required that CNT > CCR.

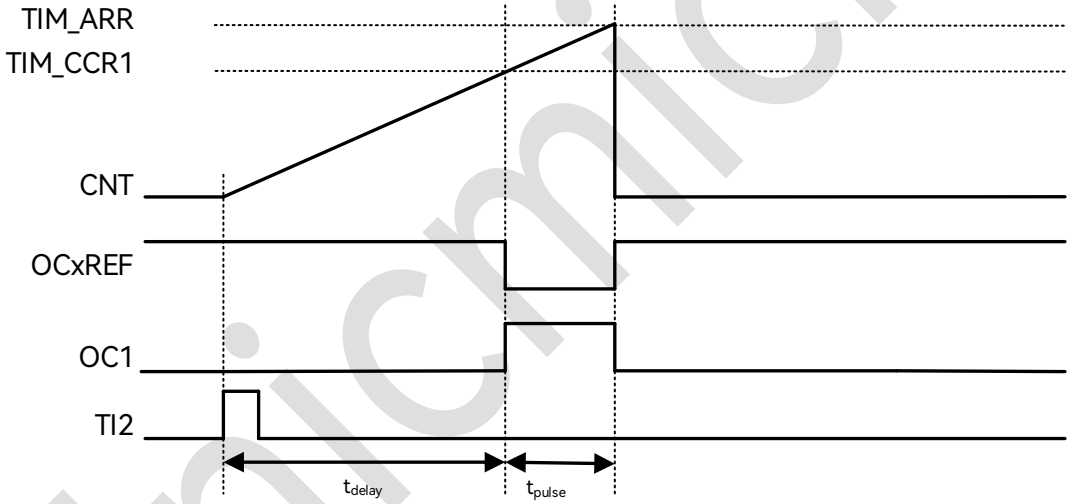

Figure 18-36: Timing Diagram ofOne-pulse Mode

In the above figure, TI2 input is used as the counter trigger signal. When the count value reaches CCR, the OCxREF outputs a low level. Once the counter counts up to ARR, the OCxREF signal returns to a high level, and the counter rolls back to 0, stopping the counting process.

The configuration for realizing the above function of TI2 as an input trigger is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH2.
2. Disable the channel by setting TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[9:8] = 01.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Select the active counting edge by setting TIM_CCER[5] = 0.
5. Select TI2FP2 as the trigger input source by setting TIM_SMCR.TS[2:0] = 110.
6. Set the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110, with TI2FP2 for activating the counter.
7. Enable the channel by setting TIM_CCER[4] = 1.

The configuration for realizing the above function of OC1 as an output is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the output channel by setting TIM_CCMR1[1:0] = 00.
4. Select the active counting edge by setting TIM_ CCMR1[6:4] = 111, in PWM mode 2.
5. Enable the channel by setting TIM_CCER[0] = 1.

Special settings for generating OPM waveform timing:

1. t_{delay} is determined by the value of TIM_CCR1.
2. t_{pulse} is determined by the difference between TIM_ARR and TIM_CCR1 (TIM_ARR − TIM_CCR1).
3. Configure to one-pulse mode by setting TIM_CR1[3] = 1.

18.6.16 Clearing OCxREF Signal on External Event

OCxREF is active at high level, and it can be pulled down directly until the next update event by applying a high level to the external ETR pin. This function can only be used in output compare and PWM modes, and does not work in forced mode. Enabling this function requires setting OCxCE to 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

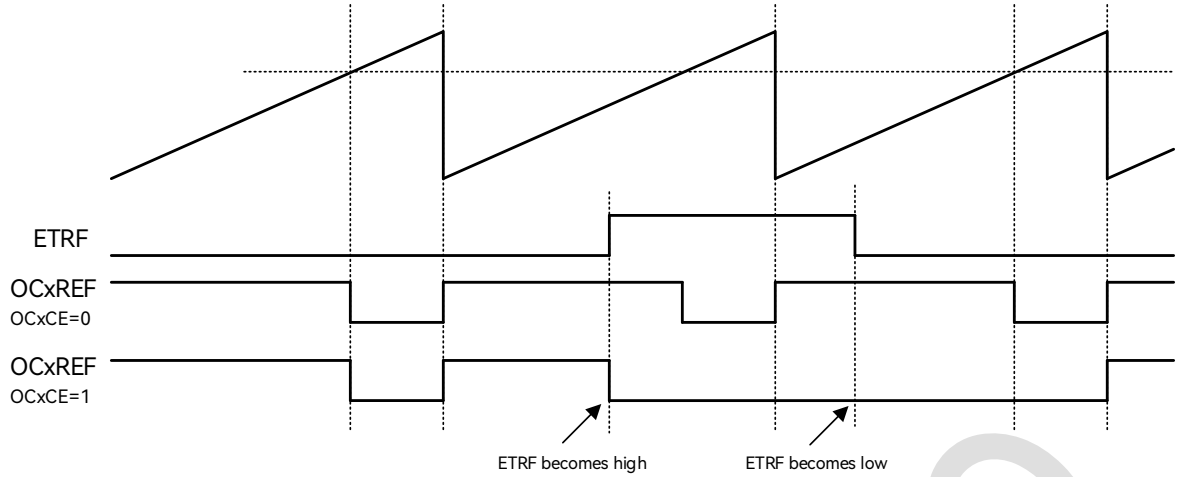

Figure 18-37: Timing Diagram of ETR Signal Clearing OCxREF of TIM

18.6.17 Encoder Interface Mode

The encoder interface mode involves two external input signals. The TIM determines whether to count up or down according to the edge of one signal relative to the level of the other signal. The following table shows the relationship between the counting mode and the two inputs:

Table 18-3: Counting Direction versus Encoder Signals

Active Edge	Level on Opposite Signal (TI1 for TI2, TI2 for TI1)	TI1 Signal		TI2 Signal	
		Rising	Falling	Rising	Falling
Counting on TI1 only	High	Down	Up	No count	No count
	Low	Up	Down	No count	No count
Counting on TI2 only	High	No count	No count	Up	Down
	Low	No count	No count	Down	Up
Counting on TI1 and TI2	High	Down	Up	Up	Down
	Low	Up	Down	Down	Up

For example, when the counter is counting on TI1, it will count down if TI2 is sampled as high level on the rising edge of TI1, and count up if TI2 is sampled as high level on the falling edge of TI1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

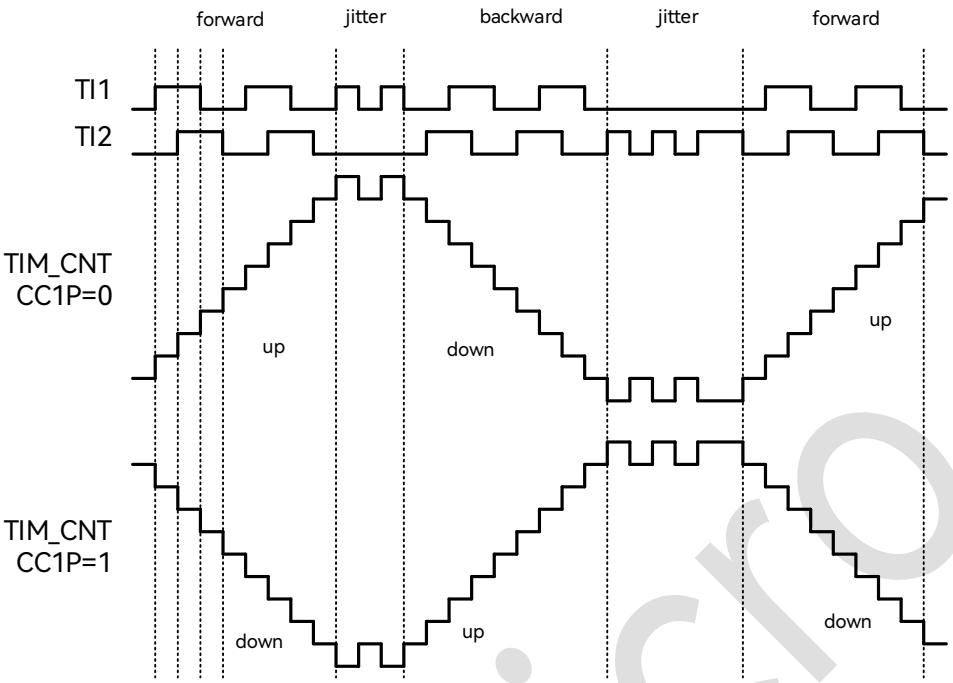

Figure 18-38: Example of Counter Operation in Encoder Interface Mode

The input channels in encoder interface mode shall be set as follows:

1. In GPIO module, configure the corresponding pins with TIM_CH1 and TIM_CH2 functions.
2. Disable the channel by setting TIM_CCER[0] = 0 and TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01 and TIM_CCMR1[9:8] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0 and TIM_CCER[5] = 0.
5. Set the slave mode controller to encoder mode 3 by setting TIM_SMCR.SMS[2:0] = 011.
6. Enable the channel by setting TIM_CCER[0] = 1 and TIM_CCER[4] = 1.

18.6.18 TIM slave mode:

When TIM is used as a slave (triggered by an external event), it can be configured to operate in three modes: reset mode, gated mode, and trigger mode.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.18.1 Reset Mode

In this mode, all the preload registers in TIM will be reinitialized in response to an event on a trigger input, and the counter will restart from 0. The following figure shows that the counter behaves normally until rising edge is detected on TI1 input, at which time the counter is cleared and restarts from 0.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to reset mode by setting TIM_SMCR.SMS[2:0] = 100.
7. Enable the channel by setting TIM_CCER[0] = 1.
8. Enable the counter by setting TIM_CR1[0] = 1.

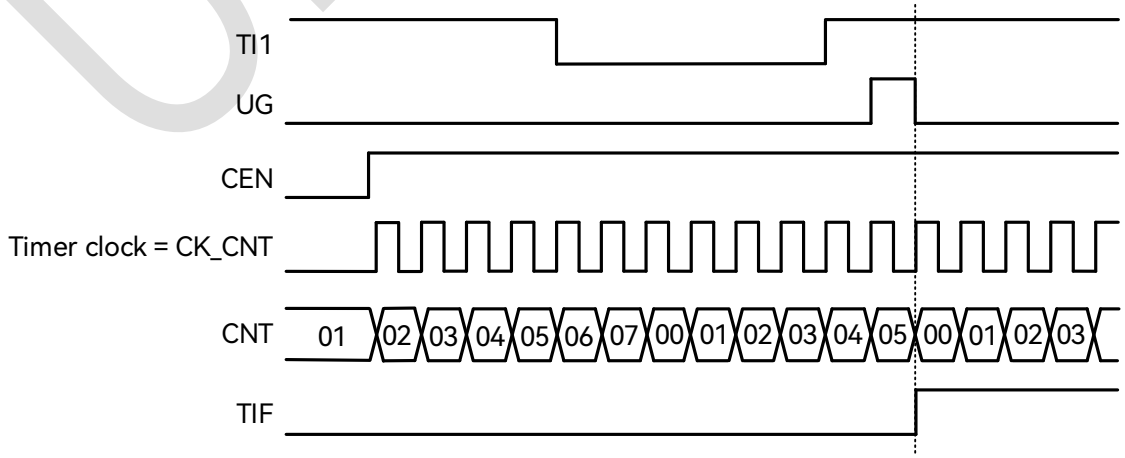

Figure 18-39: Timing Diagram in Reset Mode

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.18.2 Gated Mode

In this mode, the counter can be enabled depending on the level of a selected input. The interrupt flag is triggered whenever a level shift causes the counter to start or stop counting.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to gated mode by setting TIM_SMCR.SMS[2:0] = 101.
7. Enable the channel by setting TIM_CCER[0] = 1.
8. Enable the counter by setting TIM_CR1[0] = 1.

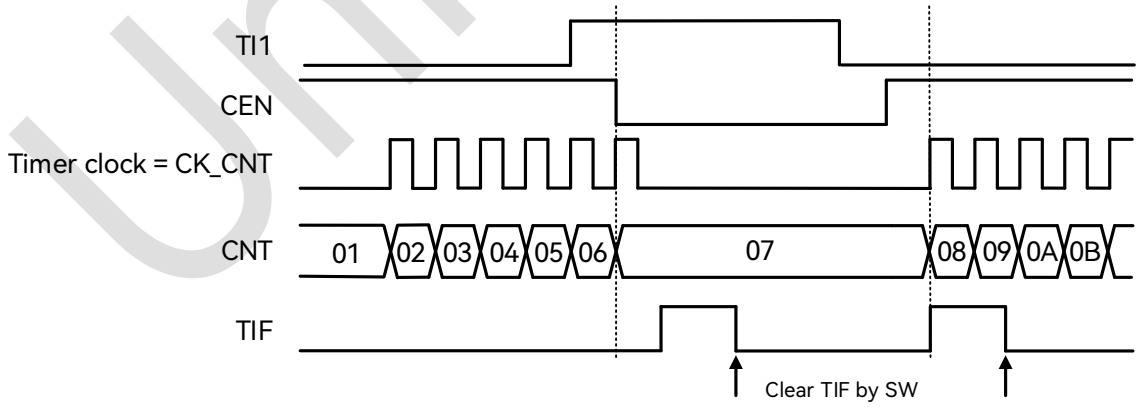

Figure 18-40: Timing Diagram in Gated Mode

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.6.18.3 Trigger Mode

The counter can start in response to an event on a selected input.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110.
7. Enable the channel by setting TIM_CCER[0] = 1.

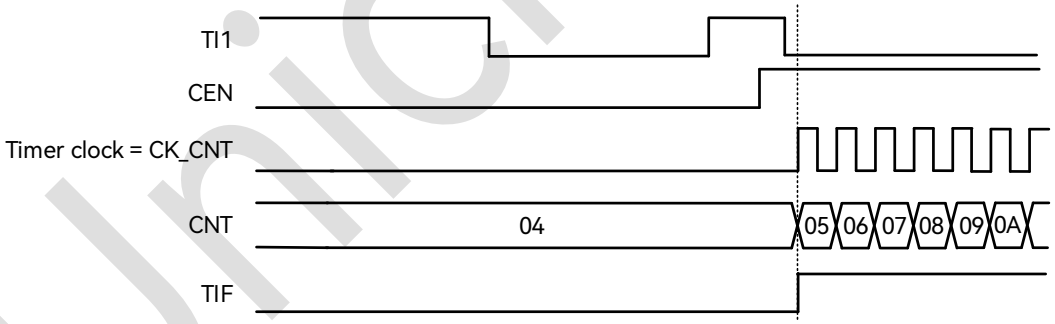

Figure 18-41: Timing Diagram in Trigger Mode

18.6.18.4 External Clock Mode 2 + Trigger Mode

In this mode, ETR can be set as the counting clock, while another external input is used as a trigger signal to start the counter. For instance, the counter begins counting on the rising edge of the ETR input after detecting the rising edge of TI1.

The configuration in the following figure example is as follows:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

1. In GPIO module, configure the corresponding pins as TIM_CH1 and TIM_ETR.
2. Select the ETP edge by setting TIM_SMCR[15] = 0.
3. Set the ETR division ratio by setting TIM_SMCR.ETPS[1:0] = 01.
4. Configure the input filter duration by setting TIM_SMCR.ETF[3:0] = 0000.
5. Set the ECE register and enable the external clock mode 2 by setting TIM_SMCR[14] = 1.
6. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
7. Select the input channel by setting TIM_CCMR1[1:0] = 01.
8. Select the active counting edge by setting TIM_CCER[1] = 0.
9. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
10. Configure the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110.
11. Enable the channel by setting TIM_CCER[0] = 1.

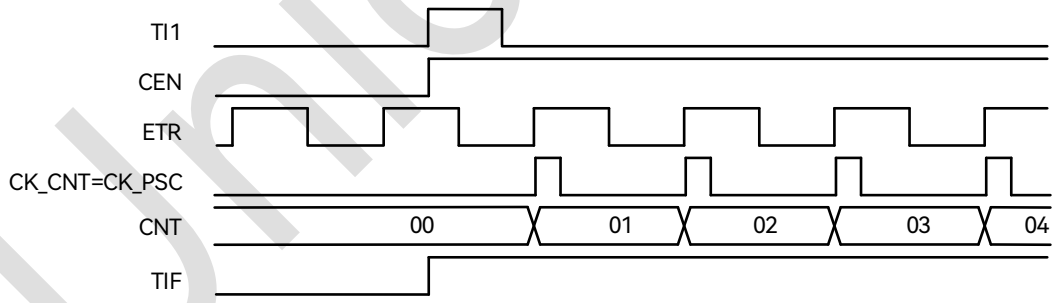

Figure 18-42: Timing Diagram in External Clock Mode 2 + Trigger Mode

18.6.19 Timer Synchronization

Timers can be cascaded together through trigger events to achieve synchronization or cascading operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

A timer can utilize four internal trigger inputs, allowing the trigger signal output from one timer to connect to the internal trigger input of other timers.

18.6.20 DMA Access

TIM supports seven types of DMA requests, including four CC channel requests, an external trigger request, an update event request, and a COM trigger request.

Each CC channel generates a DMA request, which is used to transfer the content of CCRx to RAM in capture mode, and to write the data in RAM to CCRx in compare mode. The DMA request can be configured as single-transfer or burst-transfer (CCxBURSTEN), wherein the former accesses only the CCRx register, while the latter accesses a specific set of registers based on the DCR register configuration.

In addition, DMA requests can also be generated from external trigger event, software trigger event and COM trigger event, and at the occurrence of these requests, DMA burst transfer will be started to write data to one or more registers within TIM or to read one or more register values from TIM.

Table 18-4: Seven DMA Requests Supported by TIM

DMA Request	CCxBURSTEN	DMA.CHxCTRL.DIR	DMA Access Object	Single-transfer Length
TIM_CH1	0	0	Read CCR1	1
		1	Write CCR1	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_CH2	0	0	Read CCR2	1
		1	Write CCR2	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_CH3	0	0	Read CCR3	1
		1	Write CCR3	
	1	0	Read DMAR	DBL

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

DMA Request	CCxBURSTEN	DMA.CHxCTRL.DIR	DMA Access Object	Single-transfer Length
		1	Write DMAR	
TIM_CH4	0	0	Read CCR4	1
		1	Write CCR4	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_TRIG	N/A	0	Read DMAR	DBL
		1	Write DMAR	
TIM_UEV	N/A	0	Read DMAR	DBL
		1	Write DMAR	
TIM_COM	N/A	0	Read DMAR	DBL
		1	Write DMAR	

18.6.21 MA Burst

TIM supports DMA and DMA-burst access. A DMA request can be generated at a specific event, so as to write the capture result in CCR to RAM or write the content of one or more registers in RAM to the preload register in TIM.

DMA-burst allows to generate multiple successive DMA requests upon a single event. The main purpose is to update the content of multiple registers in a row each time a given timer event is triggered, thus making it possible to dynamically modify the output waveform in real time.

The DMA controller destination is unique and must be directed to the virtual register TIM_DMAR. On a given timer event, the timer launches a sequence of DMA requests (burst). Each DMA write access to the TIM_DMAR register will be redirected to the actual function register by TIM.

The DBL bits in the register set the DMA burst length, and the DBA bits define the base address for DMA access to TIM (an offset starting from the address of the TIM_CR register).

In DMA-burst mode, all DMA access shall be directed to the virtual register DMAR, and TIM automatically accumulates the internal offset address according to the access. The DBA bits in

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the register are used to specify the destination address of the first DMA transfer within TIM, while the DBL bits are used to specify the burst length.

18.6.22 Input XOR Function

The XOR output of the input signals of channels 1–3 can be connected to the filter and edge circuit input of channel 1 for input capture or trigger.

The TI1S bit in the TIM_CR2 register is used to select whether the input to channel 1 comes from the XOR of the three channel inputs.

18.6.23 Debug Mode

When the CPU enters debug mode, the timer can either stop or continue working, and its behavior is defined by registers in the chip system.

When the timer is stopped during debugging, its output will be disabled (MOE is cleared). Depending on the register configuration, the output signal can be forced to be inactive or controlled by the GPIO module.

18.7 Register Description

TIM1 register base address: 0x4700_4000

TIM8 register base address: 0x4700_5000

The registers are listed below:

Table 18-5: List of Registers Regarding Advanced-control Timers TIM1 & TIM8

Offset Address	Name	Description
0x00	TIM_CR1	Control register 1
0x04	TIM_CR2	Control register 2
0x08	TIM_SMCR	Slave mode control register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x0C	TIM_DIER	DMA and interrupt enable register
0x10	TIM_SR	Status register
0x14	TIM_EGR	Event generation register
0x18	TIM_CCMR1	Capture/compare register 1
0x1C	TIM_CCMR2	Capture/compare register 2
0x20	TIM_CCER	Capture/compare enable register
0x24	TIM_CNT	Counter register
0x28	TIM_PSC	Prescaler register
0x2C	TIM_ARR	Auto-reload register
0x30	TIM_RCR	Repetition counter register
0x34	TIM_CCR1	Capture/compare register 1
0x38	TIM_CCR2	Capture/compare register 2
0x3C	TIM_CCR3	Capture/compare register 3
0x40	TIM_CCR4	Capture/compare register 4
0x44	TIM_BDTR	Break and dead-time control register
0x48	TIM_DCR	DMA control register
0x4C	TIM_DMAR	DMA access register

Registers are detailed in the following sections.

18.7.1 Control Register 1 (TIM_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:10	RSV	-	-	Reserved
9:8	CKD	R/W	0x0	Dead time and digital filter clock frequency division register (division ratio relative to CK_INT): 00: $t_{DTS} = t_{CK_INT}$ 01: $t_{DTS} = 2 * t_{CK_INT}$ 10: $t_{DTS} = 4 * t_{CK_INT}$ 11: Reserved, prohibited
7	APRE	R/W	0x0	Auto-reload preload enable: 0: ARR not preloaded

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: ARR preloaded
6:5	CMS	R/W	0x0	Counter alignment mode selection: 00: Edge-aligned mode 01: Center-aligned mode 1; output compare interrupt flags are set only when the counter is counting down. 10: Center-aligned mode 2; output compare interrupt flags are set only when the counter is counting up. 11: Center-aligned mode 3; output compare interrupt flags are set both when the counter is counting up or down.
4	DIR	R/W	0x0	Counting direction register: 0: Count up 1: Count down Note: This register is read-only when the timer is configured in center-aligned mode or encoder mode.
3	OPM	R/W	0x0	One-pulse output mode: 0: The counter does not stop at the occurrence of update event. 1: The counter stops at the occurrence of update event (CEN cleared automatically).
2	URS	R/W	0x0	Update request source: 0: An update interrupt or DMA request will be generated by any of the following events: ● Counter overflow/underflow ● Software setting the UG bit ● Update generated from the slave mode controller 1: An update interrupt or DMA request will be generated only at counter overflow or underflow.
1	UDIS	R/W	0x0	Update disable: 0: Update event enabled The update event can be generated by any of the following events:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				● Counter overflow/underflow ● Software setting the UG bit ● Update generated from the slave mode controller 1: Update event disabled, shadow register not updated; the counter and the prescaler will be reinitialized if the UG bit is set or if the slave mode controller receives a hardware reset.
0	CEN	R/W	0x0	Counter enable: 0: Disabled 1: Enabled Note: The external trigger mode can automatically set the CEN bit.

18.7.2 Control Register 2 (TIM_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:15	RSV	-	-	Reserved
14	OIS4	R/W	0x0	OC4 output idle state: 0: OC4 = 0 (after a dead-time if complementary output is enabled) when MOE = 0 1: OC4 = 1 (after a dead-time if complementary output is enabled) when MOE = 0
13	OIS3N	R/W	0x0	OC3N output idle state: 0: OC3N = 0 after a dead-time when MOE = 0 1: OC1N = 1 after a dead-time when MOE = 0 [For OC1−3N, (CCxP ‖ CCxNP) = 1 is also required for OCxN = 1.]

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	OIS3	R/W	0x0	OC3 output idle state: 0: OC3 = 0 (after a dead-time if complementary output is enabled) when MOE = 0 1: OC3 = 1 (after a dead-time if complementary output is enabled) when MOE = 0 [For OC1−3, (CCxP ‖ CCxNP) is also required for OCx = 1.]
11	OIS2N	R/W	0x0	OC2N output idle state: 0: OC2N = 0 after a dead-time when MOE = 0 1: OC2N = 1 after a dead-time when MOE = 0 [For OC1−3N, (CCxP‖CCxNP) = 1 is also required for OCxN = 1.]
10	OIS2	R/W	0x0	OC2 output idle state: 0: OC2 = 0 (after a dead-time if complementary output is enabled) when MOE = 0 1: OC2 = 1 (after a dead-time if complementary output is enabled) when MOE = 0 [For OC1−3, (CCxP ‖ CCxNP) is also required for OCx = 1.]
9	OIS1N	R/W	0x0	OC1N output idle state: 0: OC1N = 0 after a dead-time when MOE = 0 1: OC1N = 1 after a dead-time when MOE = 0 [For OC1−3N, (CCxP‖CCxNP) = 1 is also required for OCxN = 1.]

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
8	OIS1	R/W	0x0	OC1 output idle state: 0: OC1 = 0 (after a dead-time if complementary output is enabled) when MOE = 0 1: OC1 = 1 (after a dead-time if complementary output is enabled) when MOE = 0 [For OC1−3, (CCxP ‖ CCxNP) is also required for OCx = 1.]
7	TI1S	R/W	0x0	T1 input selection: 0: T1 input from CH1 pin 1: T1 input from XOR combination of CH1, CH2 and CH3 pins
6:4	MMS	R/W	0x0	Master mode selection, selecting the TRGO trigger mode: 000: Reset—TRGO is generated by the UG bit in the EGR register. 001: Enable—TRGO is generated by the counter enable signal, including the CEN control bit and external trigger. 010: Update—TRGO is generated by the update event. 011: Compare pulse—TRGO is generated when an input capture or compare event occurs that sets CC1F to 1. 100: Compare—TRGO is generated by OC1REF. 101: Compare—TRGO is generated by OC2REF. 110: Compare—TRGO is generated by OC3REF. 111: Compare—TRGO is generated by OC4REF.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	CCDS	R/W	0x0	Capture/compare DMA selection: 0: CCx DMA request sent when CCx event occurs 1: CCx DMA request sent when update event occurs
2	CCUS	R/W	0x0	Capture/compare control update selection, selecting the method by which the preloaded capture/compare control bits are updated: 0: By setting the COMG bit to 1 only 1: By setting the COMG bit to 1 or when the TRGI signal occurs
1	RSV	-	-	Reserved
0	CCPC	R/W	0x0	Capture/compare preload control: 0: The preload of CCxE, CCxNE, and OCxM registers is disabled. 1: The preload of CCxE, CCxNE, and OCxM registers is enabled. When enabled, these registers are updated only upon a commutation event (COM). Note: This bit acts only on channels that have a complementary output.

18.7.3 Slave Mode Control Register (TIM_SMCR)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	ETP	R/W	0x0	External trigger polarity: 0: Active at high level or rising edge 1: Active at low level or falling edge
14	ECE	R/W	0x0	Enable clock enable: 0: External clock mode 2 disabled 1: External clock mode 2 enabled; the counter is clocked by any active edge on the ETRF signal.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13:12	ETPS	R/W	0x0	External trigger prescaler: The frequency of external trigger signal ETRP must be at most 1/4 of TIM clock frequency. A prescaler can be enabled to reduce ETRP frequency when inputting fast external clocks. 00: Prescaler off 01: Frequency divided by 2 10: Frequency divided by 4 11: Frequency divided by 8
11:8	ETF	R/W	0x0	External trigger filter frequency and length selection: 0000: No filter 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS}$ / 2, N = 6 0101: $f_{SAMPLING} = f_{DTS}$ / 2, N = 8 0110: $f_{SAMPLING} = f_{DTS}$ / 4, N = 6 0111: $f_{SAMPLING} = f_{DTS}$ / 4, N = 8 1000: $f_{SAMPLING} = f_{DTS}$ / 8, N = 6 1001: $f_{SAMPLING} = f_{DTS}$ / 8, N = 8 1010: $f_{SAMPLING} = f_{DTS}$ / 16, N = 5 1011: $f_{SAMPLING} = f_{DTS}$ / 16, N = 6 1100: $f_{SAMPLING} = f_{DTS}$ / 16, N = 8 1101: $f_{SAMPLING} = f_{DTS}$ / 32, N = 5 1110: $f_{SAMPLING} = f_{DTS}$ / 32, N = 6 1111: $f_{SAMPLING} = f_{DTS}$ / 32, N = 8
7	MSM	R/W	0x0	Master/slave mode selection: 0: No action 1: The effect of an event on the trigger input (TRGI) is delayed to allow a perfect synchronization between the current timer and its slaves (through TRGO).
6:4	TS	R/W	0x0	Trigger selection for selecting the trigger source to be used to synchronize the counter: 000: Internal trigger 0 (ITR0) 001: Internal trigger 1 (ITR1)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				010: Internal trigger 2 (ITR2) 011: Internal trigger 3 (ITR3) 100: TI1 edge detector (TI1F_ED) 101: Filtered timer input 1 (TI1FP1) 110: Filtered timer input 2 (TI2FP2) 111: External trigger input (ETRF) Note: These bits can be changed only when the slave mode is disabled (i.e. SMS = 000).
3	RSV	-	-	Reserved
2:0	SMS	R/W	0x0	Slave mode selection: 000: Slave mode disabled—if CEN is enabled, then the prescaler is clocked directly by the internal clock. 001: Encoder mode 1—counter counts up/down on TI2FP1 edge depending on TI1FP2 level. 010: Encoder mode 2—counter counts up/down on TI1FP2 edge depending on TI2FP1 level. 011: Encoder mode 3—counter counts up/down on both TI1FP1 and TI2FP2 edges depending on the level of other inputs. 100: Reset mode—rising edge of the selected trigger input (TRGI) reinitializes the counter and generates an update of the registers. 101: Gated mode—the counter clock is enabled when TRGI is high, and stops as soon as it becomes low. 110: Trigger mode—the counter starts at the rising edge of TRGI (but it is not reset). 111: External clock mode 1—rising edges of TRGI directly clock the counter.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.4 DMA / Interrupt Enable Register (TIM_DIER)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	Reserved
19	CC4OF_DISABLE	R/W	0x0	CC4OF interrupt enable: 0: Enabled 1: Disabled
18	CC3OF_DISABLE	R/W	0x0	CC3OF interrupt enable: 0: Enabled 1: Disabled
17	CC2OF_DISABLE	R/W	0x0	CC2OF interrupt enable: 0: Enabled 1: Disabled
16	CC1OF_DISABLE	R/W	0x0	CC1OF interrupt enable: 0: Enabled 1: Disabled
15	RSV	-	-	Reserved
14	TDE	R/W	0x0	External trigger DMA request enable: 0: In slave mode, external trigger DMA request disabled 1: In slave mode, external trigger DMA request enabled (can be used to automatically update the preload register)
13	COMDE	R/W	0x0	COM DMA request enable: 0: COM DMA request disabled 1: COM DMA request enabled
12	CC4DE	R/W	0x0	Capture/compare channel 4 DMA request enable: 0: CC4 DMA request disabled 1: CC4 DMA request enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
11	CC3DE	R/W	0x0	Capture/compare channel 3 DMA request enable: 0: CC3 DMA request disabled 1: CC3 DMA request enabled
10	CC2DE	R/W	0x0	Capture/compare channel 2 DMA request enable: 0: CC2 DMA request disabled 1: CC2 DMA request enabled
9	CC1DE	R/W	0x0	Capture/compare channel 1 DMA request enable: 0: CC1 DMA request disabled 1: CC1 DMA request enabled
8	UDE	R/W	0x0	Update DMA request enable: 0: Update DMA request disabled 1: Update DMA request enabled
7	BIE	R/W	0x0	Break interrupt enable: 0: Break interrupt disabled 1: Break interrupt enabled
6	TIE	R/W	0x0	Trigger interrupt enable: 0: Trigger interrupt disabled 1: Trigger interrupt enabled
5	COMIE	R/W	0x0	COM interrupt enable: 0: COM interrupt disabled 1: COM interrupt enabled
4	CC4IE	R/W	0x0	Capture/compare channel 4 interrupt enable: 0: CC4 interrupt disabled 1: CC4 interrupt enabled
3	CC3IE	R/W	0x0	Capture/compare channel 3 interrupt enable: 0: CC3 interrupt disabled 1: CC3 interrupt enabled
2	CC2IE	R/W	0x0	Capture/compare channel 2 interrupt enable: 0: CC2 interrupt disabled 1: CC2 interrupt enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	CC1IE	R/W	0x0	Capture/compare channel 1 interrupt enable: 0: CC1 interrupt disabled 1: CC1 interrupt enabled
0	UIE	R/W	0x0	Update interrupt enable: 0: Update interrupt disabled 1: Update interrupt enabled

18.7.5 Status Register (TIM_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	CC4OF	R/W0C	0x0	Capture/compare channel 4 overcapture flag: This bit is only valid when the corresponding channel is configured in input capture mode. It is set by hardware and cleared by software writing it to 0. 0: No overcapture event 1: A new capture occurs while CC4IF flag is 1.
11	CC3OF	R/W0C	0x0	Capture/compare channel 3 overcapture flag: This bit is only valid when the corresponding channel is configured in input capture mode. It is set by hardware and cleared by software writing it to 0. 0: No overcapture event 1: A new capture occurs while CC3IF flag is 1.
10	CC2OF	R/W0C	0x0	Capture/compare channel 2 overcapture flag: This bit is only valid when the corresponding channel is configured in input capture mode. It is set by hardware and cleared by software writing it to 0.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: No overcapture event 1: A new capture occurs while CC2IF flag is 1.
9	CC1OF	R/W0C	0x0	Capture/compare channel 1 overcapture flag: This bit is only valid when the corresponding channel is configured in input capture mode. It is set by hardware and cleared by software writing it to 0. 0: No overcapture event 1: A new capture occurs while CC1IF flag is 1.
8	RSV	-	-	Reserved
7	BIF	R/W0C	0x0	Break interrupt flag is set by hardware and cleared by software via writing it to 0.
6	TIF	R/W0C	0x0	Trigger interrupt flag is set by hardware and cleared by software via writing it to 0.
5	COMIF	R/W0C	0x0	COM interrupt flag is set by hardware and cleared by software via writing it to 0.
4	CC4IF	R/W0C	0x0	Capture/compare channel 4 interrupt flag: If channel CC4 is configured as output: the CC4IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC4 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR4.
3	CC3IF	R/W0C	0x0	Capture/compare channel 3 interrupt flag: If channel CC3 is configured as output: the CC3IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC3 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR3.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	CC2IF	R/W0C	0x0	Capture/compare channel 2 interrupt flag: If channel CC2 is configured as output: the CC2IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC2 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR2.
1	CC1IF	R/W0C	0x0	Capture/compare channel 1 interrupt flag: If channel CC1 is configured as output: the CC1IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC1 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR1.
0	UIF	R/W0C	0x0	Update interrupt flag is set by hardware and cleared by software via writing it to 0. UIF is set and the shadow register is updated at the following events: ● Counter overflow occurs if repetition counter = 0 and UDIS = 0. ● The counter is reinitialized by software setting the UG bit if URS = 0 and UDIS = 0. ● The counter is reinitialized by a trigger event if URS = 0 and UDIS = 0.

18.7.6 Event Generation Register (TIM_EGR)

Offset address: 0x14

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	BG	W	0x0	Software break: This bit is set by software to generate a break event, and it is automatically cleared by hardware.
6	TG	W	0x0	Software trigger: This bit is set by software to generate a trigger event, and it is automatically cleared by hardware.
5	COMG	W	0x0	Software COM event: This bit is set by hardware and cleared by software writing it to 1.
4	CC4G	W	0x0	Capture/compare channel 4 software trigger: If channel CC4 is configured as output: CC4G is set, and if enabled, it can generate the corresponding interrupt and DMA request. If channel CC4 is configured as input: the current counter value is captured in the TIM_CCR4 register, the CC4G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
3	CC3G	W	0x0	Capture/compare channel 3 software trigger: If channel CC3 is configured as output: CC3G is set, and if enabled, it can generate the corresponding interrupt and DMA request. If channel CC3 is configured as input: the current counter value is captured in the TIM_CCR3 register, the CC3G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
2	CC2G	W	0x0	Capture/compare channel 2 software trigger: If channel CC2 is configured as output, CC2G is set, and if enabled, it can generate the corresponding interrupt and DMA request. If channel CC2 is configured as input: the current counter value is captured in the TIM_CCR2 register, the CC2G is set, and if enabled, it can generate the corresponding interrupt and DMA request.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	CC1G	W	0x0	Capture/compare channel 1 software trigger: If channel CC1 is configured as output: CC1G is set, and if enabled, it can generate the corresponding interrupt and DMA request. If channel CC1 is configured as input: the current counter value is captured in the TIM_CCR1 register, the CC1G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
0	UG	W	0x0	Update generation: This bit can be set by software to generate an update event, and is automatically cleared by hardware. When the software sets UG, the counter is reinitialized, the shadow register is updated, and the prescaler counter is cleared.

18.7.7 Capture/Compare Register 1 (TIM_CCMR1)

Offset address: 0x18

Reset value: 0x0000 0000

Note: This register is multiplexed for two different functions under output compare and input capture modes.

- Output compare mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	OC2CE	R/W	0x0	Output compare 2 clear enable: 0: OC2REF is not affected by ETRF input. 1: OC2REF is automatically cleared once a high level is detected on ETRF input.
14:12	OC2M	R/W	0x0	Output compare 2 mode configuration: These bits define the behavior of the output reference signal OC2REF.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				000: The comparison between the output compare register CCR2 and the counter CNT has no effect on the outputs. 001: Set OC2REF high when CCR2 = CNT (falling edge) 010: Set OC2REF low when CCR2 = CNT (falling edge) 011: Toggle OC2REF when CCR2 = CNT (falling edge) 100: Force OC2REF low (inactive) 101: Force OC2REF high (active) 110: PWM mode 1 In up-counting, OC2REF is set high when CNT < CCR2, otherwise it is set low. In down-counting, OC2REF is set low when CNT ≥ CCR2, otherwise it is set high. 111: PWM mode 2 In up-counting, OC2REF is set low when CNT < CCR2, otherwise it is set high. In down-counting, OC2REF is set high when CNT ≥ CCR2, otherwise it is set low.
11	OC2PE	R/W	0x0	Output compare 2 preload enable: 0: Preload register on CCR2 disabled; CCR2 can be written directly. 1: Preload register on CCR2 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
10	OC2FE	R/W	0x0	Output compare 2 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enabled, the trigger input will immediately change OC2REF to the output

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
9:8	CC2S	R/W	0x0	Capture/compare channel 2 selection: 00: CC2 channel is configured as output. 01: CC2 channel is configured as input, IC2 is mapped on TI2. 10: CC2 channel is configured as input, IC2 is mapped on TI1. 11: CC2 channel is configured as input, IC2 is mapped on TRC. Note: CC2S bits are writable only when the channel is OFF (CC2E = 0).
7	OC1CE	R/W	0x0	Output compare 1 clear enable: 0: OC1REF is not affected by ETRF input. 1: OC1REF is automatically cleared once a high level is detected on ETRF input.
6:4	OC1M	R/W	0x0	Output compare 1 mode: these bits define the behavior of the output reference signal OC1REF. 000: The comparison between the output compare register CCR1 and the counter CNT has no effect on the outputs. 001: Set OC1REF high when CCR1 = CNT (falling edge) 010: Set OC1REF low when CCR1 = CNT (falling edge) 011: Toggle OC1REF when CCR1 = CNT (falling edge) 100: Force OC1REF low (inactive) 101: Force OC1REF high (active) 110: PWM mode 1 In up-counting, OC1REF is set high when CNT < CCR1, otherwise it is set low.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				In down-counting, OC1REF is set low when CNT ≥ CCR1, otherwise it is set high. 111: PWM mode 2 In up-counting, OC1REF is set low when CNT < CCR1, otherwise it is set high. In down-counting, OC1REF is set high when CNT ≥ CCR1, otherwise it is set low.
3	OC1PE	R/W	0x0	Output compare 1 preload enable: 0: Preload register disabled; CCR1 can be written directly. 1: Preload register on CCR1 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
2	OC1FE	R/W	0x0	Output compare 1 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enabled, the trigger input will immediately change OC1REF to the output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
1:0	CC1S	R/W	0x0	Capture/compare channel 1 selection: 00: CC1 channel is configured as output. 01: CC1 channel is configured as input, IC1 is mapped on TI1. 10: CC1 channel is configured as input, IC1 is mapped on TI2. 11: CC1 channel is configured as input, IC1 is mapped on TRC. Note: CC1S bits are writable only when the channel is OFF (CC1E = 0).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Input capture mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	IC2F	R/W	0x0	Input capture 2 filter
11:10	IC2PSC	R/W	0x0	Input capture 2 prescaler
9:8	CC2S	R/W	0x0	Capture/compare channel 2 selection: 00: CC2 channel is configured as output. 01: CC2 channel is configured as input, IC2 is mapped on TI2. 10: CC2 channel is configured as input, IC2 is mapped on TI1. 11: CC2 channel is configured as input, IC2 is mapped on TRC. Note: CC2S bits are writable only when the channel is OFF (CC2E = 0).
7:4	IC1F	R/W	0x0	Input capture 1 filter: This bit-field defines the frequency used to sample TI1 input and the length of the digital filter applied to TI1. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8 1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5 1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
3:2	IC1PSC	R/W	0x0	Input capture 1 prescaler: 00: No prescaler

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC1PSC register is reset when CC1E = 0.
1:0	CC1S	R/W	0x0	Capture/compare channel 1 selection: 00: CC1 channel is configured as output. 01: CC1 channel is configured as input, IC1 is mapped on TI1. 10: CC1 channel is configured as input, IC1 is mapped on TI2. 11: CC1 channel is configured as input, IC1 is mapped on TRC. Note: CC1S bits are writable only when the channel is OFF (CC1E = 0).

18.7.8 Capture/Compare Register 2 (TIM_CCMR2)

Offset address: 0x1C

Reset value: 0x0000 0000

Note: This register is multiplexed for two different functions under output compare and input capture modes.

- Output compare mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	OC4CE	R/W	0x0	Output compare 4 clear enable: 0: OC4REF is not affected by ETRF input. 1: OC4REF is automatically cleared once a high level is detected on ETRF input.
14:12	OC4M	R/W	0x0	Output compare 4 mode configuration: These bits define the behavior of the output reference signal OC4REF. 000: The comparison between the output compare register CCR4 and the counter CNT has no effect on the outputs.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				001: Set OC4REF high when CCR4 = CNT. 010: Set OC4REF low when CCR4 = CNT. 011: Toggle OC4REF when CCR4 = CNT. 100: Force OC4REF low (inactive) 101: Force OC4REF high (active) 110: PWM mode 1 In up-counting, OC4REF is set high when CNT < CCR4, otherwise it is set low. In down-counting, OC4REF is set low when CNT > CCR4, otherwise it is set high. 111: PWM mode 2 In up-counting, OC4REF is set low when CNT < CCR4, otherwise it is set high. In down-counting, OC4REF is set high when CNT > CCR4, otherwise it is set low.
11	OC4PE	R/W	0x0	Output compare 4 preload enable: 0: Preload register disabled; CCR4 can be written directly. 1: Preload register on CCR4 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
10	OC4FE	R/W	0x0	Output compare 4 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enable is on, the trigger input will immediately change OC4REF to the output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
9:8	CC4S	R/W	0x0	Capture/compare channel 4 selection: 00: CC4 channel is configured as output.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				01: CC4 channel is configured as input, IC4 is mapped on TI4. 10: CC4 channel is configured as input, IC4 is mapped on TI3. 11: CC4 channel is configured as input, IC4 is mapped on TRC. Note: CC4S bits are writable only when the channel is OFF (CC4E = 0).
7	OC3CE	R/W	0x0	Output compare 3 clear enable: 0: OC3REF is not affected by ETRF input. 1: OC3REF is automatically cleared once a high level is detected on ETRF input.
6:4	OC3M	R/W	0x0	Output compare 3 mode: these bits define the behavior of the output reference signal OC3REF. 000: The comparison between the output compare register CCR3 and the counter CNT has no effect on the outputs. 001: Set OC3REF high when CCR3 = CNT. 010: Set OC3REF low when CCR3 = CNT. 011: Toggle OC3REF when CCR3 = CNT. 100: Force OC3REF low (inactive) 101: Force OC3REF high (active) 110: PWM mode 1 In up-counting, OC3REF is set high when CNT < CCR3, otherwise it is set low. In down-counting, OC3REF is set low when CNT > CCR3, otherwise it is set high. 111: PWM mode 2 In up-counting, OC3REF is set low when CNT < CCR3, otherwise it is set high. In down-counting, OC3REF is set high when CNT > CCR3, otherwise it is set low.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	OC3PE	R/W	0x0	Output compare 3 preload enable: 0: Preload register disabled; CCR3 can be written directly. 1: Preload register on CCR3 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
2	OC3FE	R/W	0x0	Output compare 3 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enabled, the trigger input will immediately change OC3REF to the output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
1:0	CC3S	R/W	0x0	Capture/compare channel 3 selection: 00: CC3 channel is configured as output. 01: CC3 channel is configured as input, IC3 is mapped on TI3. 10: CC3 channel is configured as input, IC3 is mapped on TI4. 11: CC3 channel is configured as input, IC3 is mapped on TRC. Note: CC3S bits are writable only when the channel is OFF (CC3E = 0).

- Input capture mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:12	IC4F	R/W	0x0	Input capture 4 filter: This bit-field defines the frequency used to sample TI4 input and the length of the digital filter applied to TI4. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8 1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5 1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
11:10	IC4PSC	R/W	0x0	Input capture 4 prescaler: 00: No prescaler 01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC4PSC register is reset when CC4E = 0.
9:8	CC4S	R/W	0x0	Capture/compare channel 4 selection: 00: CC4 channel is configured as output. 01: CC4 channel is configured as input, IC4 is mapped on TI4. 10: CC4 channel is configured as input, IC4 is mapped on TI3. 11: CC4 channel is configured as input, IC4 is mapped on TRC. Note: CC4S bits are writable only when the channel is OFF (CC4E = 0).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:4	IC3F	R/W	0x0	Input capture 3 filter: This bit-field defines the frequency used to sample TI3 input and the length of the digital filter applied to TI3. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8 1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5 1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
3:2	IC3PSC	R/W	0x0	Input capture 3 prescaler: 00: No prescaler 01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC3PSC register is reset when CC3E = 0.
1:0	CC3S	R/W	0x0	Capture/compare channel 3 selection: 00: CC3 channel is configured as output. 01: CC3 channel is configured as input, IC3 is mapped on TI3. 10: CC3 channel is configured as input, IC3 is mapped on TI4. 11: CC3 channel is configured as input, IC3 is mapped on TRC. Note: CC1S bits are writable only when the channel is OFF (CC1E = 0).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.9 Capture/Compare Enable Register (TIM_CCER)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	-	Reserved
13	CC4P	R/W	0x0	Capture/compare 4 output polarity: CC4 channel configured as output: 0: OC4 is OC4REF. 1: OC4 is the inverse OC4REF. CC4 channel configured as input: 0: Non-inverted: Capture is done on the rising edge of IC4. 1: Inverted: Capture is done on the falling edge of IC4.
12	CC4E	R/W	0x0	Capture/compare 4 output enable: CC4 channel configured as output: 0: OC4 not active 1: OC4 active CC4 channel configured as input: 0: Capture disabled 1: Capture enabled
11	CC3NP	R/W	0x0	Capture/compare 3 complementary output polarity: **CC3 channel configured as output:** 0: OC3N is the inverse OC3REF. 1: OC3N is OC3REF. **CC3 channel configured as input:** Used in conjunction with CC3P
10	CC3NE	R/W	0x0	Capture/compare 3 complementary output enable: 0: Disabled—OC3N is not active. 1: Enabled—OC3N signal is output.
9	CC3P	R/W	0x0	Capture/compare 3 output polarity: **CC3 channel configured as output:** 0: OC3 is OC3REF.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: OC3 is the inverse OC3REF. **CC3 channel configured as input:** In conjunction with CC3NP, {CC3NP,CC3P} bits select the polarity of IC3: 00: Non-inverted: Capture is done on the rising edge of IC3. 01: Inverted: Capture is done on the falling edge of IC3. 10: Reserved 11: Non-inverted mode: Capture is done on both the rising edge and falling edge of IC3. This configuration must not be used for encoder mode.
8	CC3E	R/W	0x0	Capture/compare 3 output enable: refer to CC1E
7	CC2NP	R/W	0x0	Capture/compare 2 complementary output polarity: CC2 channel configured as output: 0: OC2N is the inverse OC2REF. 1: OC2N is OC2REF. CC2 channel configured as input: Used in conjunction with CC2P
6	CC2NE	R/W	0x0	Capture/compare 2 complementary output enable: 0: Disabled—OC2N is not active. 1: Enabled—OC2N signal is output.
5	CC2P	R/W	0x0	Capture/compare 2 output polarity: CC2 channel configured as output: 0: OC2 is OC2REF. 1: OC2 is the inverse OC2REF. CC2 channel configured as input: In conjunction with CC2NP, {CC2NP,CC2P} bits select the polarity of IC2: 00: Non-inverted: Capture is done on the rising edge of IC2. 01: Inverted: Capture is done on the falling edge of IC2.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				10: Reserved 11: Non-inverted mode: Capture is done on both the rising edge and falling edge of IC2. This configuration must not be used for encoder mode.
4	CC2E	R/W	0x0	Capture/compare 2 output enable: CC2 channel configured as output: 0: OC2 not active 1: OC2 active CC2 channel configured as input: 0: Capture disabled 1: Capture enabled
3	CC1NP	R/W	0x0	Capture/compare 1 complementary output polarity: CC1 channel configured as output: 0: OC1N is the inverse OC1REF. 1: OC1N is OC1REF. CC1 channel configured as input: In conjunction with CC1P, select the polarity of IC2:
2	CC1NE	R/W	0x0	Capture/compare 1 complementary output enable: 0: Disabled—OC1N is not active. 1: Enabled—OC1N signal is output.
1	CC1P	R/W	0x0	Capture/compare 1 output polarity: CC1 channel configured as output: 0: OC1 is OC1REF. 1: OC1 is the inverse OC1REF. CC1 channel configured as input: In conjunction with CC1NP, {CC1NP,CC1P} bits select the polarity of IC1: 00: Non-inverted: Capture is done on the rising edge of IC1. 01: Inverted: Capture is done on the falling edge of IC1. 10: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				11: Non-inverted mode: Capture is done on both the rising edge and falling edge of IC1. This configuration must not be used for encoder mode.
0	CC1E	R/W	0x0	Capture/compare 1 output enable: CC1 channel configured as output: 0: OC1 not active 1: OC1 active CC1 channel configured as input: 0: Capture disabled 1: Capture enabled

As shown in the table below, MOE is the total output enable bit of the timer, OSSI is the off_state selection bit in IDLE state (MOE = 0), and OSSR is the off_state selection bit in RUN state (MOE = 1).

Table 18-6: State Correspondence between Control Register and Complementary Output Channel

Control Register					Output State	
MOE	OSSI	OSSR	CCxE	CCxNE	OCx Output State	OCxN Output State
1	X	0	0	0	Output disabled (not driven by TIM) OCx = 0, OCx_EN = 0	Output disabled (not driven by TIM) OCxN = 0, OCxN_EN = 0
		0	0	1	Output disabled (not driven by TIM) OCx = 0, OCx_EN = 0	OCxREF + Polarity OCxN = OCxREF xor CCxNP, OCxN_EN = 1
		0	1	0	OCxREF + Polarity OCx = OCxREF xor CCxP, OCx_EN = 1	Output disabled (not driven by the timer) OCxN = 0, OCxN_EN = 0
		0	1	1	OCREF + Polarity + dead-time OCx_EN = 1	Complementary to OCREF (not OCREF) + Polarity + dead-time OCxN_EN = 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

<table>
<tr><th colspan="5">Control Register</th><th colspan="2">Output State</th></tr>
<tr><th>MOE</th><th>OSSI</th><th>OSSR</th><th>CCxE</th><th>CCxNE</th><th>OCx Output State</th><th>OCxN Output State</th></tr>
<tr><td rowspan="4"></td><td rowspan="4"></td><td>1</td><td>0</td><td>0</td><td>Output disabled (not driven by the timer)
OCx = CCxP, OCx_EN = 0</td><td>Output disabled (not driven by the timer)
OCxN = CCxNP, OCxN_EN = 0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Off-state (output enabled with inactive state)
OCx = CCxP
OCx_EN = 1</td><td>OCxREF + Polarity
OCxN = OCxREF xor CCxNP
OCxN_EN = 1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>OCxREF + Polarity
OCx = OCxREF xor CCxP
OCx_EN = 1</td><td>Off-state (output enabled with inactive state)
OCxN = CCxNP, OCxN_EN = 1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>OCREF + Polarity + dead-time
OCx_EN = 1</td><td>Complementary to OCREF (not OCREF) + Polarity + dead-time
OCxN_EN = 1</td></tr>
<tr><td rowspan="6">0</td><td>0</td><td rowspan="6">X</td><td>0</td><td>0</td><td>Output disabled (not driven by TIM)
OCx = CCxP, OCx_EN = 0</td><td>Output disabled (not driven by TIM)
OCxN = CCxNP, OCxN_EN = 0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td rowspan="3">Output disabled (not driven by TIM)
If no clock is provided: OCx = CCxP, OCx_EN = 0, OCxN = CCxNP, OCxN_EN = 0
If the clock is provided: OCx = OISx, OCxN = OISxN after a dead-time.</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Output disabled (not driven by TIM)
OCx = CCxP, OCx_EN = 0</td><td>Output disabled (not driven by TIM)
OCxN = CCxNP, OCxN_EN = 0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td colspan="2">Off-state (output enabled with inactive state)
If no clock is provided: OCx = CCxP, OCx_EN = 1, OCxN = CCxNP, OCxN_EN = 1
If the clock is provided: OCx = OISx, OCxN = OISxN after a dead-time.</td></tr>
</table>

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.10 Counter Register (TIM_CNT)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CNT	R/W	0x0	Counter value

18.7.11 Prescaler Register (TIM_PSC)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	PSC	R/W	0x0	Counter clock (CK_CNT) prescaler value: $f_{CK_CNT} = f_{CK_PSC} / (PSC[15:0] + 1)$ This is a preload register whose content are transferred into the shadow register at each update event.

18.7.12 Auto-reload Register (TIM_ARR)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ARR	R/W	0x0	Auto-reload value in the case of counter overflow This is a preload register whose content are transferred into the shadow register at each update event.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.13 Repetition Counter Register (TIM_RCR)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	REP	R/W	0x0	Repetition counter value: If REP is not 0, the repetition counter counts down at each update condition, and an update event is triggered once REP = 0.

18.7.14 Capture/Compare Register 1 (TIM_CCR1)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR1	R/W	0x0	Capture/compare channel 1 register: **If channel CC1 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC1 output. **If channel CC1 is configured as input:** CCR1 stores the counter value at the time of the most recent input capture event. In this case, CCR1 is read-only.

18.7.15 Capture/Compare Register 2 (TIM_CCR2)

Offset address: 0x38

Reset value: 0x0000 0000

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CCR2	R/W	0x0	Capture/compare channel 2 register: **If channel CC2 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC2 output. **If channel CC2 is configured as input:** CCR2 stores the counter value at the time of the most recent input capture event. In this case, CCR2 is read-only.

18.7.16 Capture/Compare Register 3 (TIM_CCR3)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR3	R/W	0x0	Capture/compare channel 3 register: **If channel CC3 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC3 output. **If channel CC3 is configured as input:** CCR3 stores the counter value at the time of the most recent input capture event. In this case, CCR3 is read-only.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.7.17 Capture/Compare Register 4 (TIM_CCR4)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR4	R/W	0x0	Capture/compare channel 4 register: **If channel CC4 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC4 output. **If channel CC4 is configured as input:** CCR4 stores the counter value at the time of the most recent input capture event. In this case, CCR4 is read-only.

18.7.18 Break and Dead-time Register (TIM_BDTR)

Offset address: 0x44

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	MOE	R/W	0x0	Main output enable: This register controls the output enable of all channels, and the independent channel is enabled if the respective enable bits (i.e. CCxE and CCxNE) are set. MOE is set by software or automatically set by hardware if AOE = 1. It is cleared asynchronously by hardware as soon as the break input is active. 0: OC and OCN outputs disabled (with specific IO state determined by OSSI) 1: OC and OCN outputs enabled (determined by the states of respective CCxE and CCxNE bits)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
14	AOE	R/W	0x0	Automatic output enable: 0: MOE can be set only by software. 1: MOE can be set by software or automatically set at the next update event.
13	BKP	R/W	0x0	Break polarity: 0: Break input active low 1: Break input active high
12	BKE	R/W	0x0	Break enable: 0: Break input disabled 1: Break input enabled
11	OSSR	R/W	0x0	Output disabled state (of channel at CCx(N)E = 0) in run mode (MOE = 1): This is only effective for channels with complementary output enabled when MOE = 1. 0: When the output channel is disabled, OC and OCN do not drive IO, and OCxN always drives IO. 1: When the output channel is disabled, OC and OCN drive GPIO to be “inactive” (referring to OIS in CR2). *(If cfg_timx_break_ossi0_disout = 1 / TIMCFGR[0] / [1] = 0, it will have no effect and will directly cause OC4 not to drive.)”
10	OSSI	R/W	0x0	Output disabled state (of channel at CCx(N)E = 0) in idle mode (MOE = 0): This is only effective for output channels when MOE=0. 0: When the output channel is disabled, OC and OCN do not drive GPIO. ● If cfg_timx_break_ossi0_disout = 0 / TIMCFGR[0] / [1][12] = 1, it will keep driving. ● OCxN always drives IO. 1: When the output channel is disabled, OC and OCN are forced with idle level first, and then with “inactive level” (referring to OIS in CR2) after a dead-time.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				*(When MOE=0, OCx(N) = (|CCx(N)P) &&OISx(N), OC4 = OIS4)
9:8	LOCK	R/W	0x0	Register write protection configuration: 00: No write protection 01: Protection level 1— DTG, OISx, OISxN, BKE, BKP and AOE cannot be rewritten. 10: Protection level 2— In addition to level 1, CCxP, CCxNP, OSSR and OSSI cannot be rewritten. 11: Protection level 3— In addition to level 2, OCxM and OCxPE cannot be rewritten when the corresponding channel is configured as output. Note: The LOCK register cannot be rewritten after being written with any value, and the write-protected register can only be rewritten after the TIM module is reset.
7:0	DTG	R/W	0x0	This bit-filed defines the duration of the dead-time inserted between the complementary outputs. DT corresponds to this duration. DTG[7:5] = 0xx: DT = DTG[7:0] * t_{DTS} DTG[7:5] = 10x: DT = (64 + DTG[5:0]) * 2 * t_{DTS} DTG[7:5] = 110: DT= (32 + DTG[4:0]) * 8 * t_{DTS} DTG[7:5] = 111: DT= (32 + DTG[4:0]) * 16 * t_{DTS}

18.7.19 DMA Control Register (TIM_DCR)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12:8	DBL	R/W	0x0	DMA burst length: A read or write access to the TIM_DMAR register will trigger DMA transfer with a burst length of 1−18. 00000: Burst length = 1

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				00001: Burst length = 2 …… 10001: Burst length = 18 Others: Invalid value, write prohibited
7:5	RSV	-	-	Reserved
4:0	DBA	R/W	0x0	DMA base address, defined as the offset address directed to the register: 00000: TIM_CR1 00001: TIM_CR2 00010: TIM_SMCR …… Note: When DBA + DBL exceeds the TIM register address range, the actual burst transfer stops automatically when it reaches the highest TIM register address, i.e., the burst length is shortened.

18.7.20 DMA Access Register (TIM_DMAR)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DMAR	R/W	0x0	DMA register for burst access When using DMA burst transfer, set the DMA channel peripheral address to TIM_DMAR. Accesses to this register will point to the register specified in TIM_DCR, and TIM will generate multiple DMA requests based on the DBL value.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.8 Operation Procedure

18.8.1 Counting Mode

1. Enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_RCR[7:0] to 0 to disable the repetition counting.
7. Set TIM_CR1[2] to 1 so that an update interrupt or DMA request will be generated only at counter overflow or underflow.
8. Set TIM_CR1[1] to 0 to enable the update event.
9. Set TIM_EGR[0] to 1 so that when the software sets UG, the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
10. Set TIM_CR1[0] to 1 to enable the counter.
11. Set TIM_DIER[0] to 1 to enable the update event interrupt.

18.8.2 PWM Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_RCR[7:0] to 0 to disable the repetition counting.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. According to the output channel, set TIM_CCMRx[1:0/9:8] to 0 to configure channel x as output.
8. Configure the OCxM[6:4/14:12] bits in TIM_CCMRx register to select PWM mode 1/2.
9. Configure TIM_CCER[1/3/5/7/9/11/13] to set the output polarity.
10. Set TIM_CCER[0/4/8/12] to 1 to enable channel x output.
11. Configure the main output enable bit TIM_BDTR[15] to 1 to enable OC and OCN outputs.
12. Set TIM_CR1[2] to 1 so that an update interrupt or DMA request will be generated only at counter overflow or underflow.
13. Set TIM_CR1[1] to 0 to enable the update event.
14. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
15. Set TIM_CR1[0] to 1 to enable the counter.
16. Set TIM_DIER[0] to 1 to enable the update event interrupt.
17. Configure TIM_CCRx[31:0] to set the compare value of channel x.

18.8.3 Input Capture Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_RCR[7:0] to 0 to disable the repetition counting.
7. Configure TIM_CCMRx[1:0/9:8] to set channel CCx as input, and perform mapping as required.
8. Configure TIM_CCER[1/3/5/7/9/11/13] to set the capture polarity.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9. Configure TIM_CCMRx[7:4/15:12] to set the sampling frequency and filter length, generally set to 0.
10. Configure TIM_CCMRx[3:2/11:10] to set the prescaler value of input capture.
11. Set TIM_CCER[0/4/8/12] to 1 to enable the capture function.
12. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
13. Set TIM_CR1[0] to 1 to enable the counter.
14. Set TIM_DIER[1/2/3/4] to 1 to enable the capture interrupt of channel x.

18.8.4 Complementary Output and Dead-time Insertion

Take 168 MHz as an example to calculate the cycle time: t_{DTS} = 5.95 ns

- DT = (0–127) * 5.95 = 0–755.65 ns, DTG[7:5] = 0xx: DT = DTG[7:0] * t_{DTS}
- DT = (64 + (0–63)) * 2 * 5.95 = 761.6–1511.3 ns, DTG[7:5] = 10x: DT = (64 + DTG[5:0]) * 2 * t_{DTS}
- DT = (32 + (0–31)) * 8 * 5.95 = 1523.2–2998.8 ns, DTG[7:5] = 110: DT = (32 + DTG[4:0]) * 8 * t_{DTS}
- DT = (32 + (0–31)) * 16 * 5.95 = 3046.4–5997.6 ns, DTG[7:5] = 111: DT = (32 + DTG[4:0]) * 16 * t_{DTS}

Before initializing the PWM mode, add the following configurations:

1. Configure TIM_BDTR[7:0] to set the duration of the dead-time inserted between the complementary outputs.
2. Set TIM_CCER[2/6/10] to 1 to enable the complementary output.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

18.8.5 Break Function

Before initializing the PWM mode, add the following configurations:

1. Configure TIM_BDTR[11] to set the off-state for run mode.
2. Configure TIM_BDTR[10] to set the off-state for idle mode.
3. Configure TIM_BDTR[13] to set the break signal polarity.
4. Configure TIM_CCER[1/5/9/13] to set the output polarity of OCx.
5. Configure TIM_CCER[3/7/11] to set the output polarity of OCxN.
6. Configure TIM_CR2[8/10/12/14] to set the OCx output state in idle mode.
7. Configure TIM_CR2[9/11/13] to set the OCxN output state in idle mode.
8. Configure TIM_BDTR[14] to set the setting mode of TIM_BDTR[15].
9. Set TIM_BDTR[12] to 1 to enable the break input.

18.8.6 Encoder Interface Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_RCR[7:0] to 0 to disable the repetition counting.
7. Set TIM_CCMR1[1:0] to 1 to configure channel CC1 as input with IC1 mapped on TI1.
8. Set TIM_CCMR1[9:8] to 1 to configure channel CC2 as input with IC2 mapped on TI2.
9. Configure TIM_CCER[1] and TIM_CCER[3] to set the capture polarity.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10. Configure TIM_CCER[1] and TIM_CCER[7] to set the capture polarity.
11. Configure TIM_CCMR1[7:4] to set the sampling frequency and filter length, generally set to 0.
12. Configure TIM_CCMR1[15:12] to set the sampling frequency and filter length, generally set to 0.
13. Configure TIM_SMCR[2:0] to set encoder mode 1/2/3.
14. Set ATIM_CCER[0] to 1 to enable the capture function of channel 1.
15. Set ATIM_CCER[4] to 1 to enable the capture function of channel 2.
16. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
17. Set TIM_CR1[0] to 1 to enable the counter.
18. Set TIM_DIER[1] to 1 to enable capture interrupt of channel 1.

18.8.7 DMA Mode

- **In input capture mode, the channel capture value of TIMx is transferred to SRAM via DMA:**
    1. In input capture mode, before setting TIM_EGR[0] bit by software and enabling the counter, add the following configurations:
    2. Configure TIM_DCR[12:8] to set the DMA burst length.
    3. Configure TIM_DCR[4:0] to set the DMA base address. Generally, the base address here selects the capture/compare register corresponding to the capture channel.
    4. Set TIM_DIER[9/10/11/12] to 1 to enable the CCx DMA request.
    5. Set TIM_CR2[3] to 0 to enable the CCxDMA request generation at CCx event.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. For details on DMA controller configuration, please refer to chapter “11 DMA Controller”.
7. After initiating DMA transfer, when a capture event occurs on the channel, DMA will transfer the value stored in base address to SRAM.

- **In output compare mode, the value in SRAM is transferred via DMA to the compare register of TIMx.**

1. In PWM mode, before setting TIM_EGR[0] bit by software and enabling the counter, add the following configurations:
2. Configure TIM_DCR[12:8] to set the DMA burst length.
3. Configure TIM_DCR[4:0] to set the DMA base address. Generally, the base address here selects the capture/compare register corresponding to the compare channel.
4. Set TIM_DIER[9/10/11/12] to 1 to enable the CCx DMA request.
5. Set TIM_CR2[3] to 0 to enable the CCxDMA request generation at CCx event.
6. For details on DMA controller configuration, please refer to chapter “11 DMA Controller”.
7. After the DMA transfer is started, when the counter value matches the compare value, DMA will transfer the value in SRAM to the base address.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19 General-purpose Timers (TIM2 ~ TIM5 & TIM9 ~ TIM14)

19.1 Overview

The general-purpose timer consists of a 32-bit auto-reload counter driven by a programmable prescaler. It may be used for a variety of purposes, including input capture, output compare and PWM.

19.2 Main Features

- 32-bit up, down, up/down auto-reload counter
- 16-bit programmable prescaler allowing real-time adjustment of the counter clock division
- 4 independent channels for input capture, output compare, PWM generation, and one-pulse output
- Support cascading with other timers
- Interrupt or DMA event can be generated in the following cases:
  - Counter overflow/underflow, counter initialization (triggered by software or hardware)
  - Trigger event (counter start, stop, initialization, or count by internal/external trigger)
  - Input capture
  - Output compare
- Support incremental quadrature encoder and Hall sensor
- Trigger input for external clock

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.3 System Block Diagram

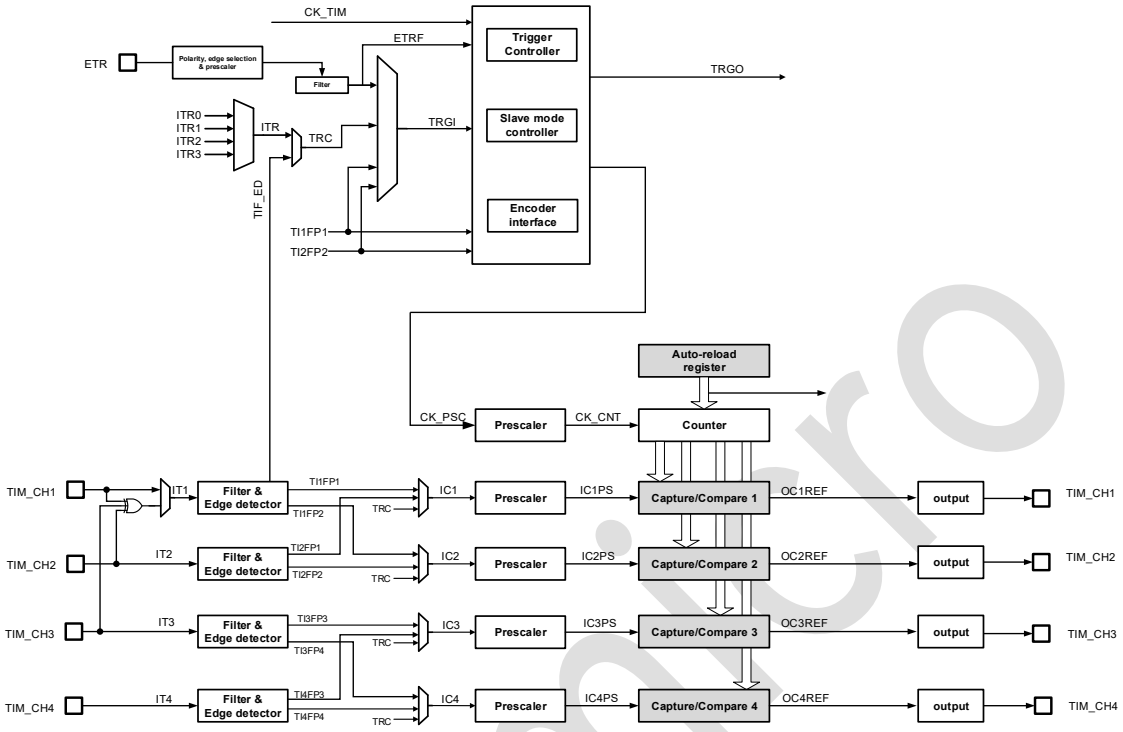

Figure 19-1: System Block Diagram of TIM2 ~ TIM5 & TIM9 ~ TIM14

19.4 Pin Description

Table 19-1: Pin Descriptions on TIM2 ~ TIM5 & TIM9 ~ TIM14

Function Pin	Alternate Function Pin	Direction	Functional Description
TIM2_CH1_ETR	PA0, PA5, PA15	Input/output	Used as an external trigger input for TIM2_ETR or as channel 1 input capture or PWM output for TIM2_CH1. TIM2_ETR and TIM2_CH1 cannot be used simultaneously.
TIM2_CH2	PA1, PB3	Input/output	Channel input capture / PWM output signal
TIM2_CH3	PA2, PB10	Input/output	Channel input capture / PWM output signal

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
TIM2_CH4	PA3, PB11	Input/output	Channel input capture / PWM output signal
TIM3_ETR	PD2	Input	External trigger input
TIM3_CH1	PA6, PB4, PC6	Input/output	Channel input capture / PWM output signal
TIM3_CH2	PA7, PB5, PC7	Input/output	Channel input capture / PWM output signal
TIM3_CH3	PB0, PC8	Input/output	Channel input capture / PWM output signal
TIM3_CH4	PB1, PC9	Input/output	Channel input capture / PWM output signal
TIM4_ETR	PE0	Input	External trigger input
TIM4_CH1	PB6, PD12	Input/output	Channel input capture / PWM output signal
TIM4_CH2	PB7, PD13	Input/output	Channel input capture / PWM output signal
TIM4_CH3	PB8, PD14	Input/output	Channel input capture / PWM output signal
TIM4_CH4	PB9, PD15	Input/output	Channel input capture / PWM output signal
TIM5_CH1	PA0, PC0	Input/output	Channel input capture / PWM output signal
TIM5_CH2	PA1, PC1	Input/output	Channel input capture / PWM output signal
TIM5_CH3	PA2, PC2	Input/output	Channel input capture / PWM output signal
TIM5_CH4	PA3, PC3	Input/output	Channel input capture / PWM output signal
TIM9_CH1	PA2, PE5	Input/output	Channel input capture / PWM output signal
TIM9_CH2	PA3, PE6	Input/output	Channel input capture / PWM output signal
TIM9_CH3	PE3	Input/output	Channel input capture / PWM output signal
TIM9_CH4	PE4	Input/output	Channel input capture / PWM output signal

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
TIM10_CH1	PB8, PC4	Input/output	Channel input capture / PWM output signal
TIM10_CH2	PA15, PC5	Input/output	Channel input capture / PWM output signal
TIM10_CH3	PB2, PC11	Input/output	Channel input capture / PWM output signal
TIM10_CH4	PC10	Input/output	Channel input capture / PWM output signal
TIM11_CH1	PB9, PD5	Input/output	Channel input capture / PWM output signal
TIM11_CH2	PD2	Input/output	Channel input capture / PWM output signal
TIM11_CH3	PD3, PD6	Input/output	Channel input capture / PWM output signal
TIM11_CH4	PD4	Input/output	Channel input capture / PWM output signal
TIM12_CH1	PB14	Input/output	Channel input capture / PWM output signal
TIM12_CH2	PB15	Input/output	Channel input capture / PWM output signal
TIM12_CH3	PB12	Input/output	Channel input capture / PWM output signal
TIM12_CH4	PB13	Input/output	Channel input capture / PWM output signal
TIM13_CH1	PA6, PD7, PB2	Input/output	Channel input capture / PWM output signal
TIM13_CH2	PA5, PD8, PB3	Input/output	Channel input capture / PWM output signal
TIM13_CH3	PB0, PD9	Input/output	Channel input capture / PWM output signal
TIM13_CH4	PB1, PD10	Input/output	Channel input capture / PWM output signal
TIM14_CH1	PA7, PE12	Input/output	Channel input capture / PWM output signal
TIM14_CH2	PE7	Input/output	Channel input capture / PWM output signal

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
TIM14_CH3	PE8	Input/output	Channel input capture / PWM output signal
TIM14_CH4	PE10	Input/output	Channel input capture / PWM output signal

19.5 Functional Description

19.5.1 Time-base Unit

The main block of the time-base unit is a 32-bit counter with its related auto-reload register. The counter can count up, down, or both up and down. The counter clock can be divided by a 16-bit prescaler.

The counter, the auto-reload register and the prescaler register can be rewritten or read by software, which is true even when the counter is running.

The time-base unit includes:

- Counter register (TIM_CNT)
- Prescaler register (TIM_PSC)
- Auto-reload register (TIM_ARR)

The auto-reload register is preloaded, which is controlled by the auto-reload preload enable (ARPE) bit in the register. When ARPE = 0, write to the ARR register, and the written data is directly transferred to the shadow register. When ARPE = 1, the data written to the ARR register is transferred to the shadow register when an update event (TIM_CNT overflow or underflow) occurs. The update event of ARR can also be actively triggered by software via register operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The counter TIM_CNT is clocked by the prescaler output TIM_PSC, which is enabled only when the counter enable bit (CEN) in the register is set. When CNT = ARR, this round of counting is over and the update event is sent.

TIM_PSC is a synchronous prescaler that can divide the counter clock frequency by any factor between 1 and 65536. The PSC register is also buffered, and overwriting PSC does not actually overwrite the shadow register unless a new update event occurs. Thus the PSC register can be changed in real time on the fly, and the new prescaler ratio is taken into account at the next update event.

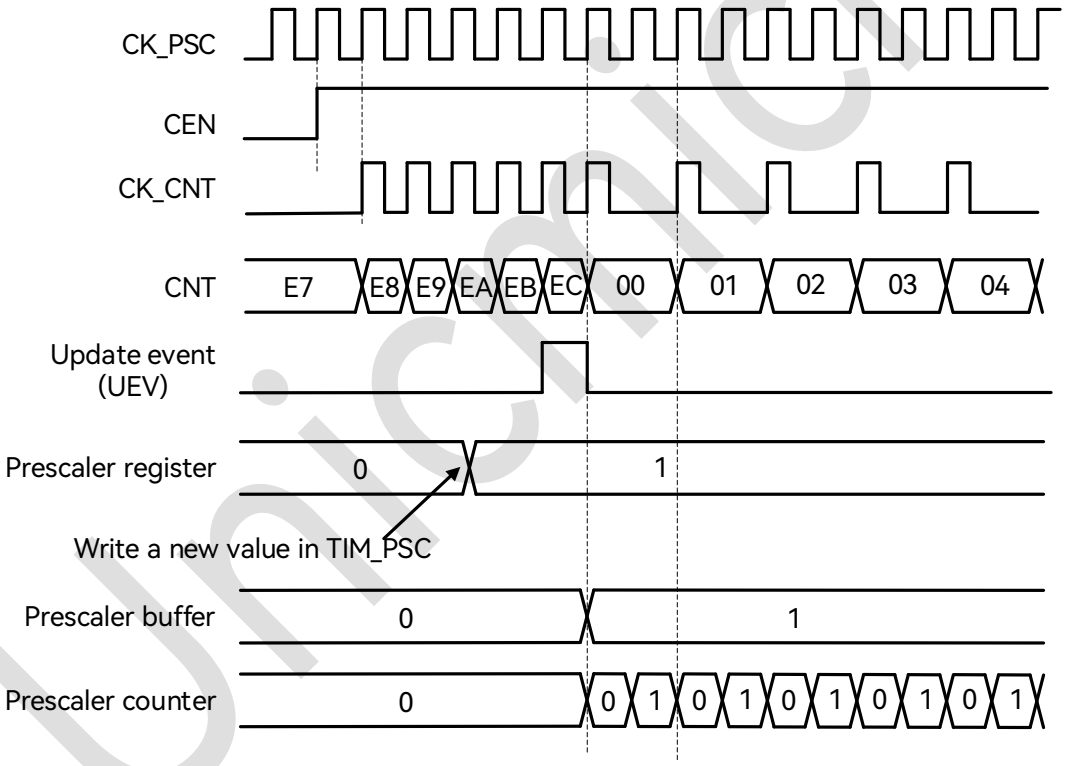

Figure 19-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 2

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

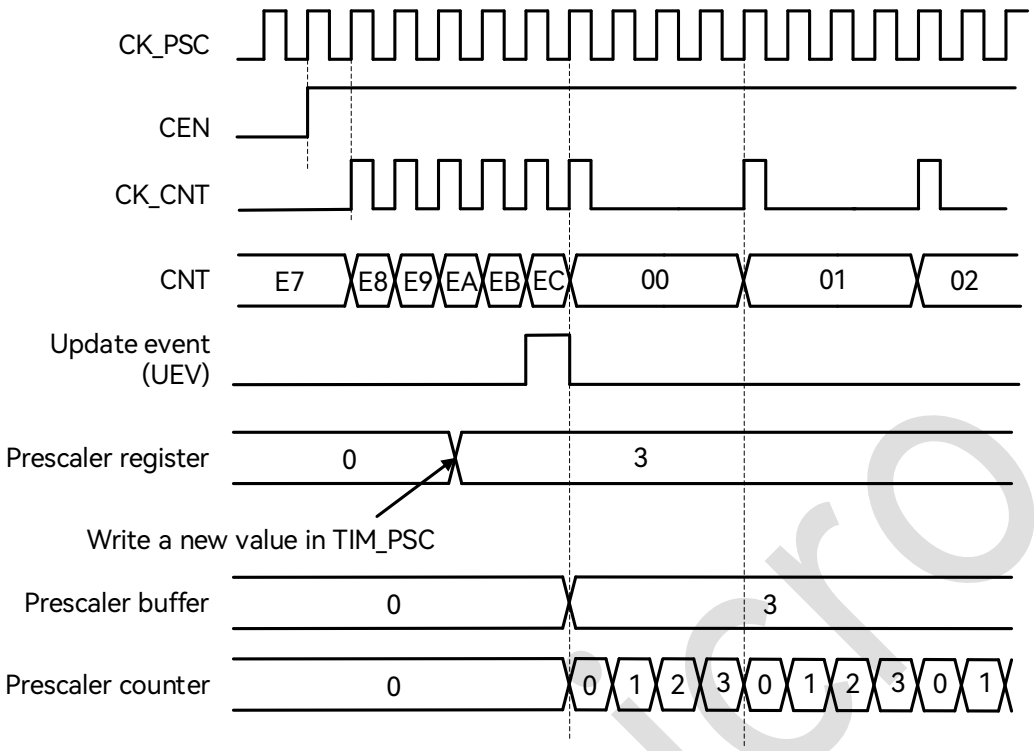

Figure 19-3: Counter Timing Diagram with Prescaler Division Changing from 1 to 4

19.5.2 Counter Operation Mode

The counter supports up-counting mode, down-counting mode and center-aligned mode.

19.5.2.1 Up-counting Mode

In up-counting mode, the counter counts from 0 to the auto-reload value, i.e. CNT = ARR, generating an overflow event, and then restarts counting from 0.

If the repetition counter is enabled, the counter repeats the above process a number of times (RCR + 1) as defined in RCR before generating an underflow event.

The software can directly trigger an update event by setting the UG bit in the register, at which time the CNT and the prescaler registers are automatically cleared. Whether setting the UG register triggers UIF (update interrupt flag) is determined by the setting of the URS register.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The update event can be disabled by setting the UDIS bit in the register to avoid updating the shadow register while writing new values in the preload registers.

When an update event occurs, the following registers are updated and the UIF bit is set:

- The auto-reload shadow register ARR is reloaded with the content of TIM_ARR register.
- The prescaler shadow register PSC is reloaded with the content of TIM_PSC register.

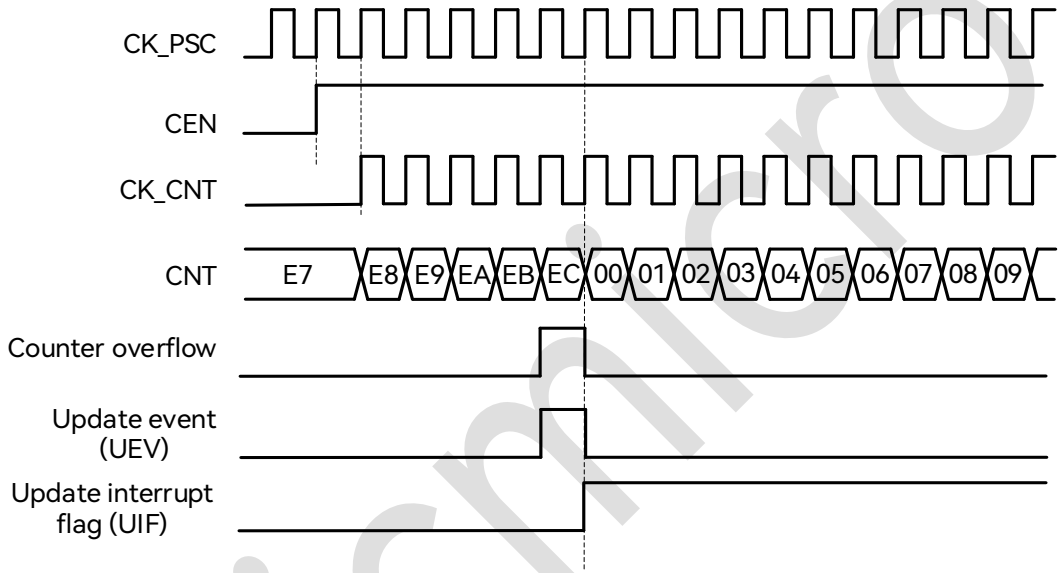

Figure 19-4: Up-counting Waveform Diagram, Internal Clock not Divided

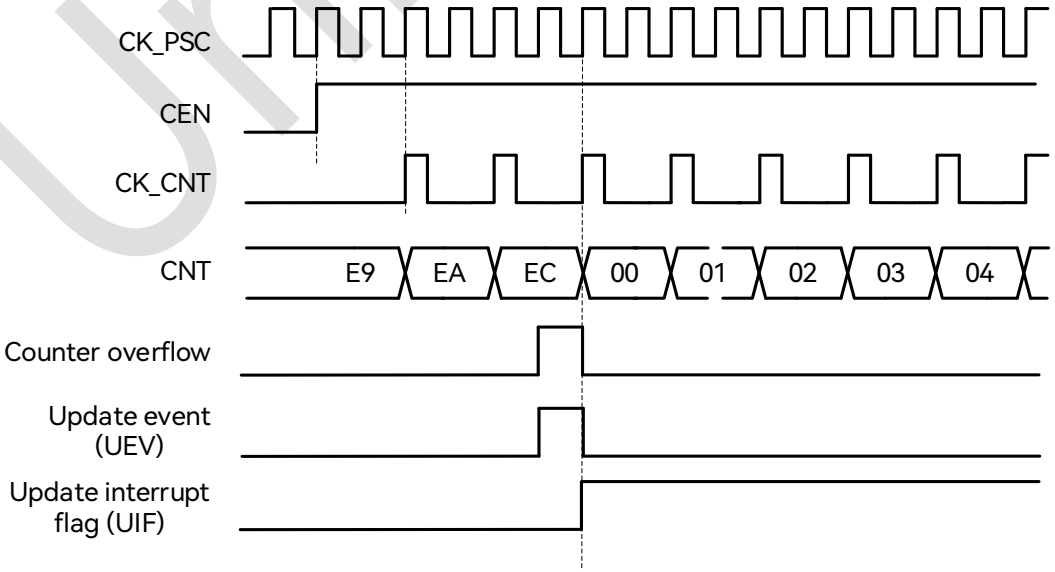

Figure 19-5: Up-counting Waveform Diagram, Internal Clock Divided by 2

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

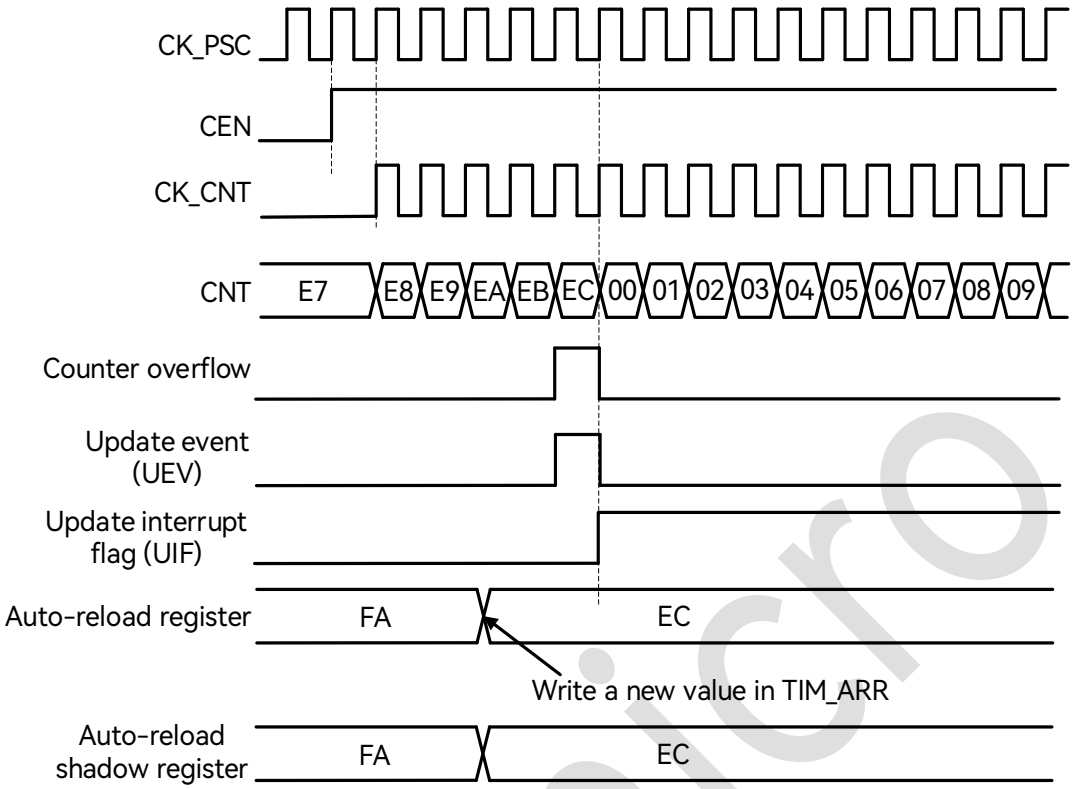

Figure 19-6: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded)

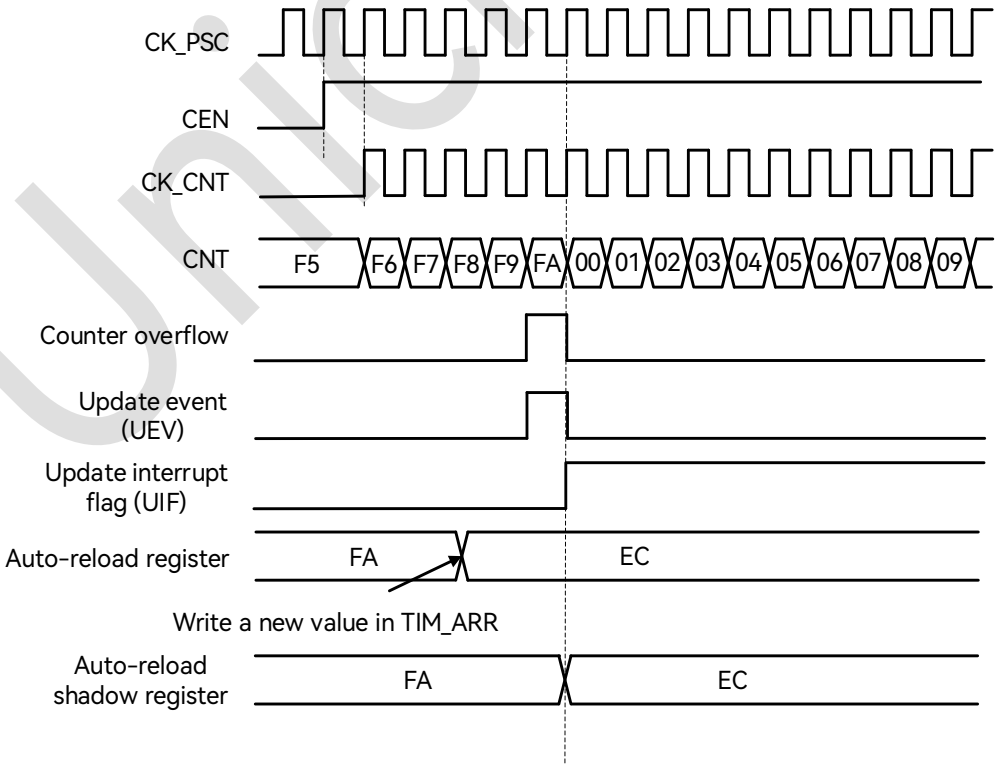

Figure 19-7: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.2.2 Down-counting Mode

In down-counting mode, the counter counts from the auto-reload value down to 0, generating an underflow event, and then restarts counting from the auto-reload value.

If the repetition counter is enabled, the counter repeats the above process a number of times (RCR + 1) as defined in RCR before generating an underflow event.

The software can directly trigger an update event by setting the UG bit in the register, at which time the CNT and the prescaler registers are automatically cleared. Whether setting the UG register triggers UIF (update interrupt flag) is determined by the setting of the URS register.

The update event can be disabled by setting the UDIS bit in the register to avoid updating the shadow register while writing new values in the preload registers.

When an update event occurs, the following registers are updated and the UIF bit is set:

- The auto-reload shadow register ARR is reloaded with the content of TIM_ARR register.
- The prescaler shadow register PSC is reloaded with the content of TIM_PSC register.

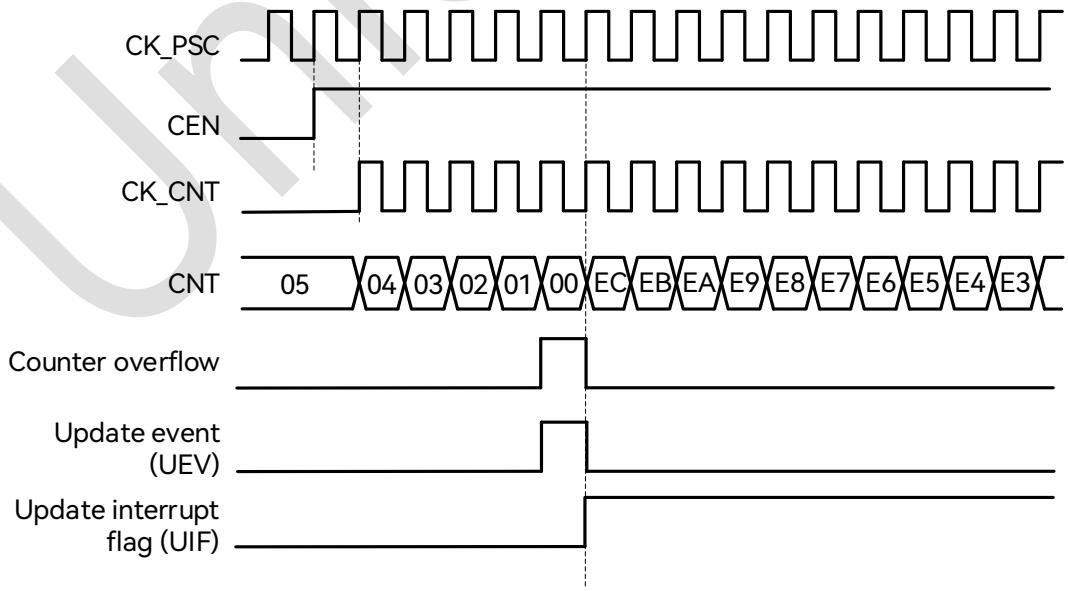

Figure 19-8: Down-counting Waveform Diagram, Internal Clock not Divided

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

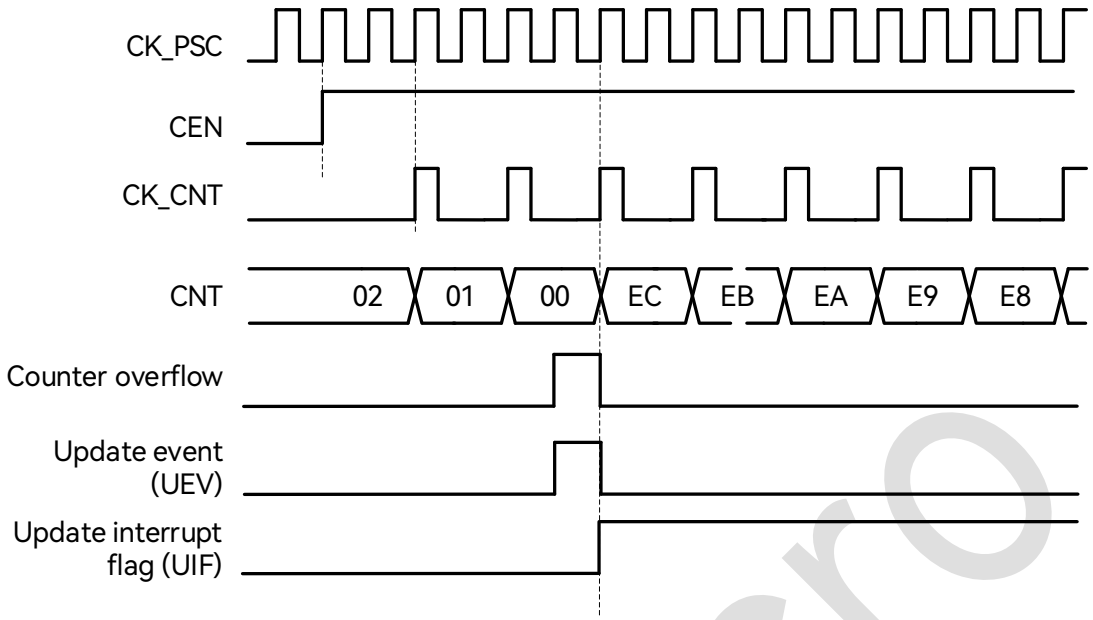

Figure 19-9: Down-counting Waveform Diagram, Internal Clock Divided by 2

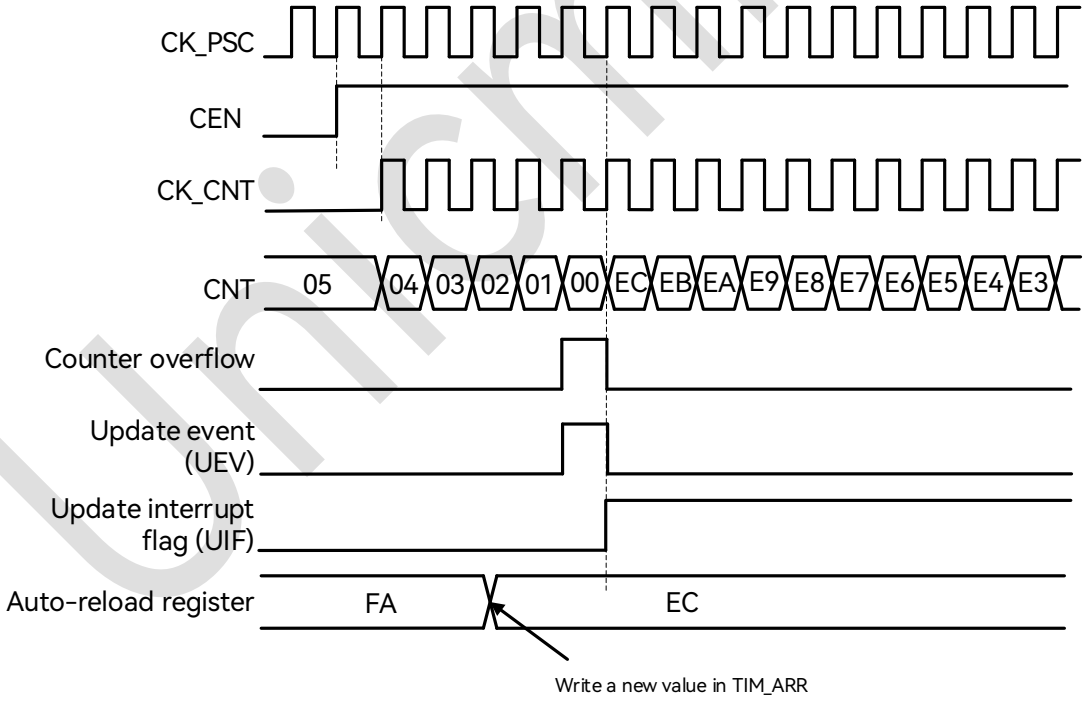

Figure 19-10: Down-counting Waveform Diagram, Internal Clock Divided by 2

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

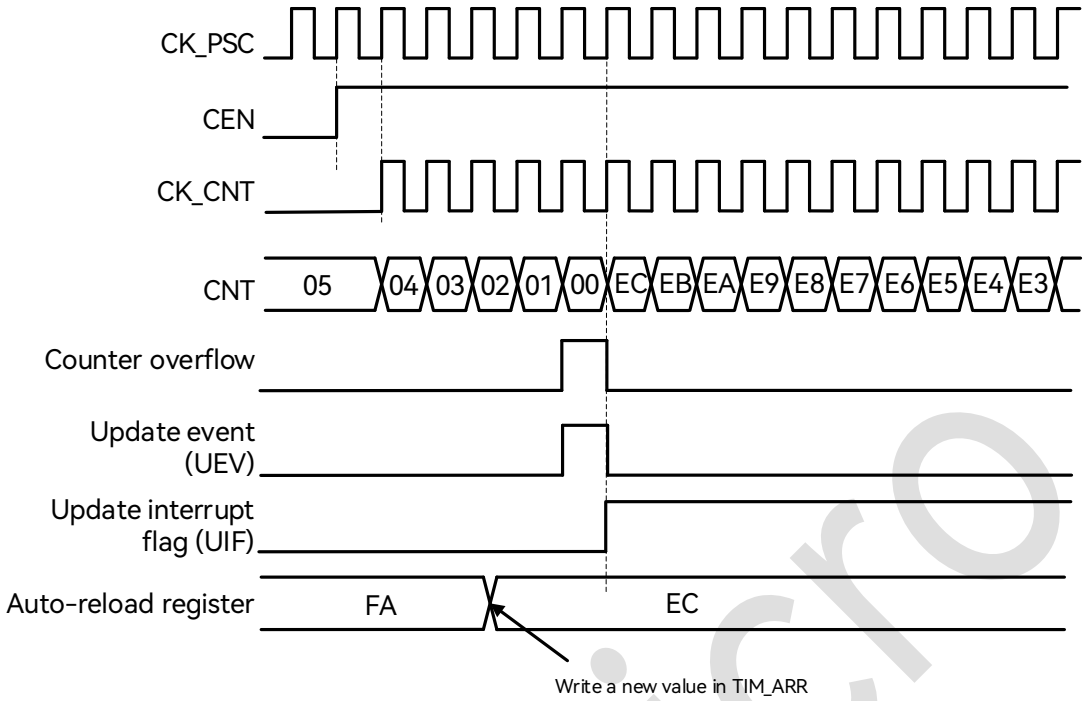

Figure 19-11: Down-counting Waveform Diagram, Update Event when Repetition Counter is not Used

19.5.2.3 Center-aligned Counting Mode

In center-aligned mode, the counter counts from 0 to the auto-reload value - 1 and generates a counter overflow event, then counts from the auto-reload value down to 1 and generates a counter underflow event, and then restarts counting from 0.

The CMS[1:0] bits in the register are used for enabling the center-aligned mode and selecting the output compare mode herein. The center-aligned mode is active when CMS! = 00. The output compare interrupt flag of channels configured in output is set when: the counter counts down (CMS = 01), the counter counts up (CMS = 10), the counter counts up and down (CMS = 11).

In this mode, the DIR direction bit in the register cannot be written by software. It is updated by hardware and gives the current direction of the counter.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The counter updates the shadow registers of ARR and PSC at each counter overflow and underflow.

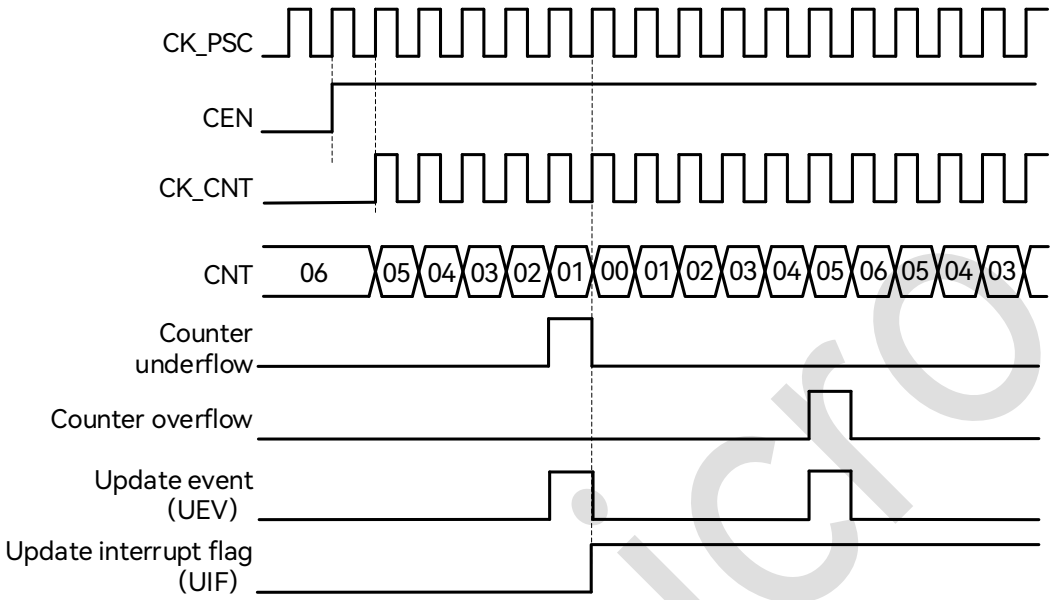

Figure 19-12: Center-aligned Counter Timing Diagram, TIM_PCS = 0, TIM_ARR = 0x6

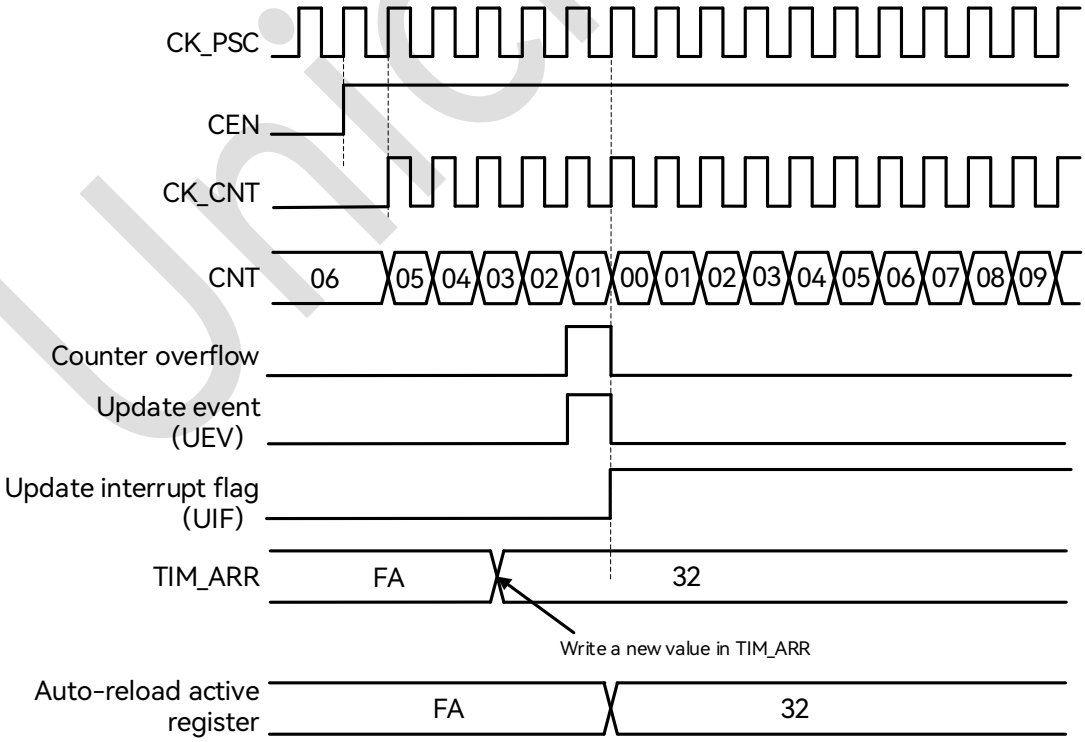

Figure 19-13: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Underflow)

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

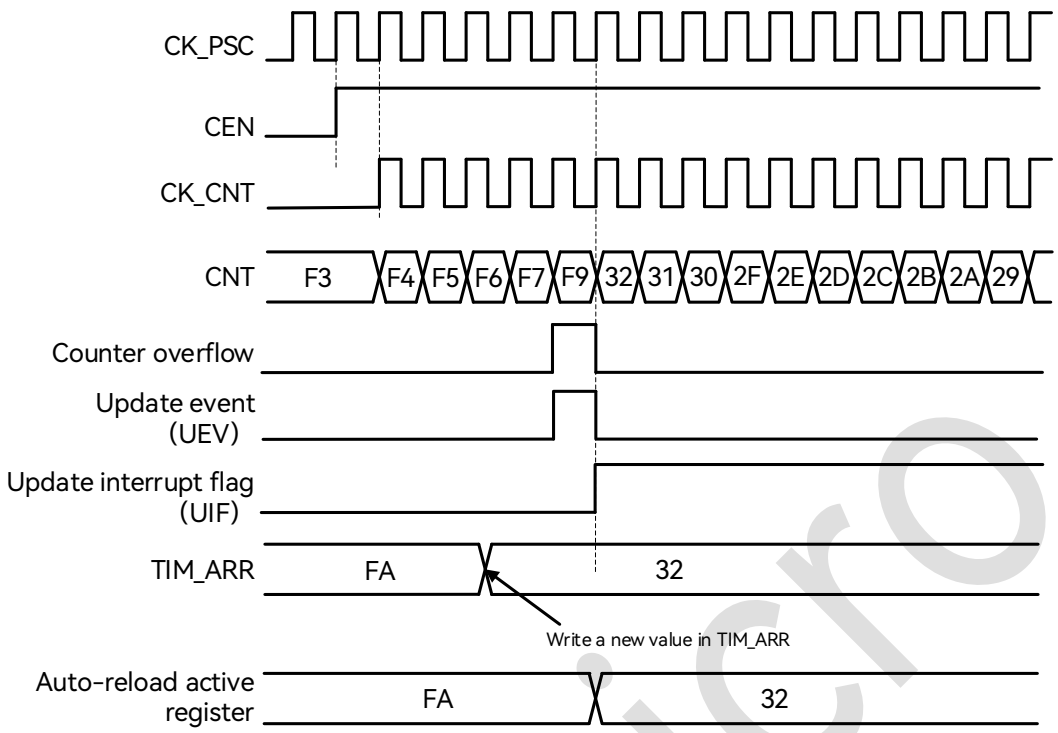

Figure 19-14: Counter Timing Diagram, Update Event with ARPE = 1 (Counter Overflow)

19.5.3 Preload Register

- The following functional registers support the preload function:
  - Auto-reload register TIM_ARR
  - Prescaler register TIM_PSC (preload function cannot be disabled)
  - Channel control register TIM_CCR
  - CCxE and CCxNE control register
  - OCxM control register

The preload function can be enabled or disabled by software for all of the above registers except TIM_PSC.

- Registers with preload function contain two sets of physical entities:
  - Shadow register: the register being used by the actual timer
  - Preload register: the register accessible to software

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- When the preload function is disabled, the register with preload function has the following characteristics:
  - The preload register can be accessed and overwritten by software in real time.
  - The shadow register is updated synchronously with the preload register.
- If the preload function is enabled, then:
  - All software operations access the preload register.
  - At the occurrence of update event, the content of all preload registers will be synchronously transferred to the corresponding shadow registers.

19.5.4 Counter Clock

The counter clock can be provided by the following clock sources:

- Internal clock: Timerx_clk
- External clock mode 1: External input pin TIx
- External clock mode 2: External trigger input ETR
- Internal trigger inputs (ITRx): using the trigger output (TRGO) of one timer as the counter clock

19.5.4.1 Internal Clock Source

If the slave mode controller is disabled (SMS = 000), then the CEN, DIR and UG bits are controlled by software.

After the UG bit is set and the update signal is synchronized by CLK_PSC, the counter value is reinitialized.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

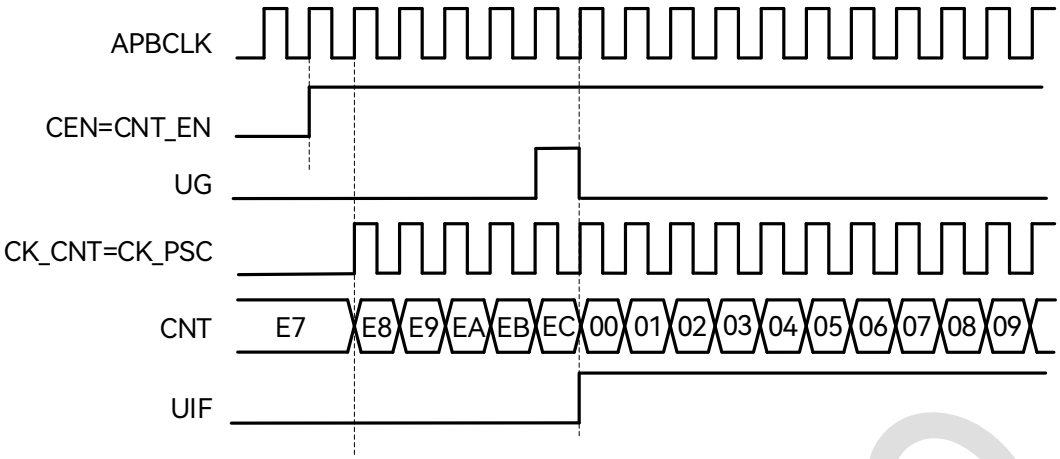

Figure 19-15: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1

19.5.4.2 External Clock Source Mode 1

In this mode, the external pin input signal is directly used as the counter clock when SMS = 111, and the counter can count at either rising or falling edge on a selected input.

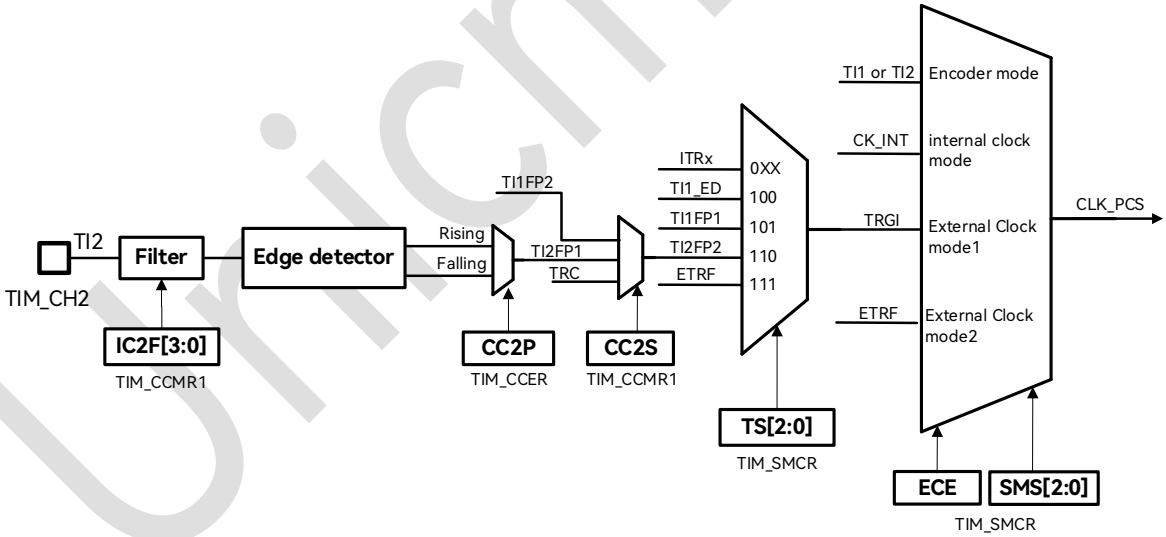

Figure 19-16: Example of External Clock Connection

The external input signal will be synchronized with the internal clock before triggering the counter counting, and the TIF flag will be set by the valid edge on the input.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

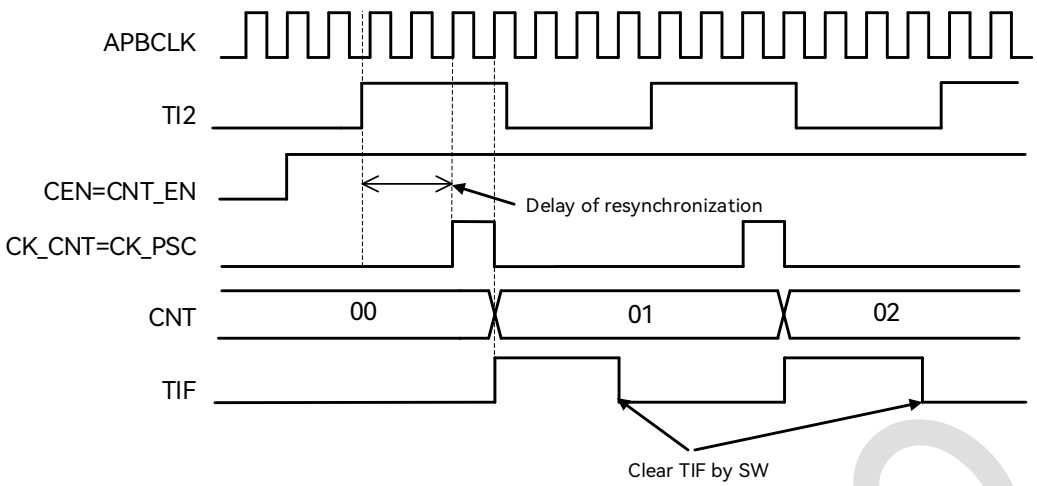

Figure 19-17: Timing Diagram in External Clock Source Mode 1

When counting with an external clock, the internal clock (timerx_clk) shall still be enabled so that TIM can use timerx_clk to synchronize and filter the external input clock. In external clock mode 1, the external input clock is first subject to filtering and edge selection to obtain a valid counting edge, which is input to the prescaler module as the valid operating clock (CLK_PSC).

The external clock synchronization adopts a simple two-stage flip-flop structure, so in order to avoid metastability, the external input clock width is required to be at least 2 timerx_clk cycles.

In this mode, only the inputs of channels 1 and 2 can be used as clock inputs, and the required configuration is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH2.
2. Disable the channel by setting TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[9:8] = 01, with IC2 mapped on TI2.
4. Select the active counting edge to be rising or falling edge by setting TIM_CCER[5] = 0.
5. Configure the input filter duration by writing the IC2F[3:0] bits in the TIM_CCMR1 register (if no filter is required, keep IC2F = 0000).
6. Enable the external clock source mode 1 by setting TIM_SMCR[2:0] = 111.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Select TI2 as the trigger input source by setting TIM_SMCR[6:4] = 110.
8. Enable the channel by setting TIM_CCER[4] = 1.
9. Enable the counter by setting TIM_CR1[0] = 1.

The following diagram shows an example of typical external clock source mode 1:

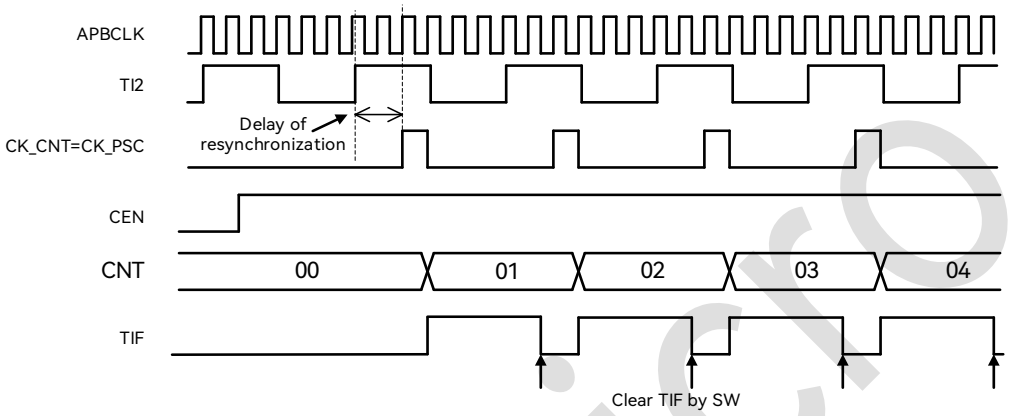

Figure 19-18: Timing Diagram in External Clock Source Mode 1

19.5.4.3 External Clock Source Mode 2

In this mode, the counter counts at either rising edge or falling edge (double-edge not supported) on the external trigger input TIM_ETR.

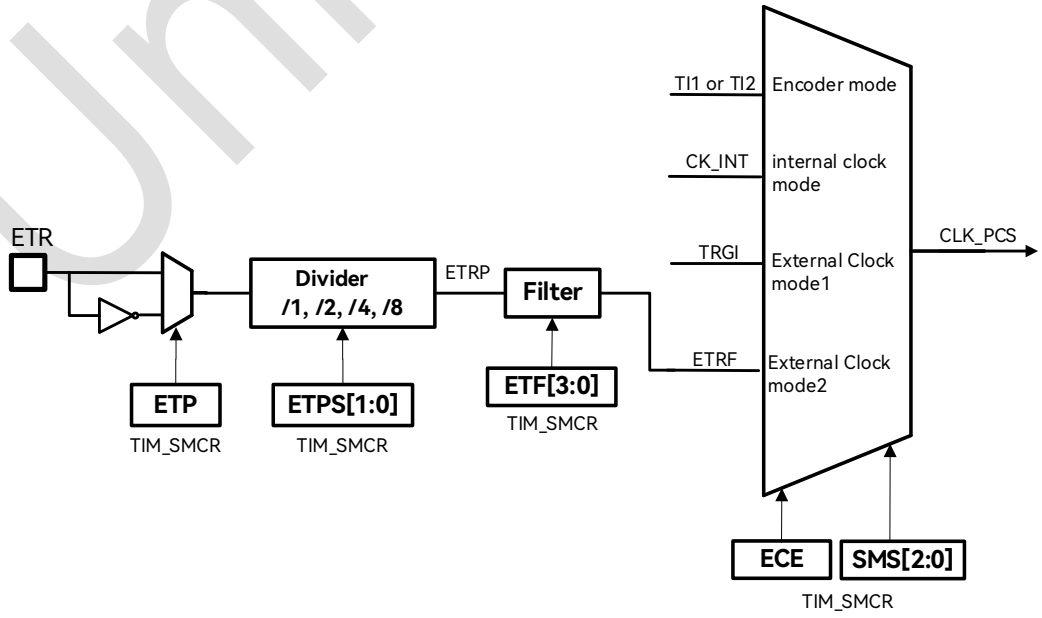

Figure 19-19: External Trigger Input Block Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The following diagram shows the counter counting each 2 rising edges on ETR. The delay between the rising edge of ETR and the actual clock of the counter is due to the resynchronization of the internal clock.

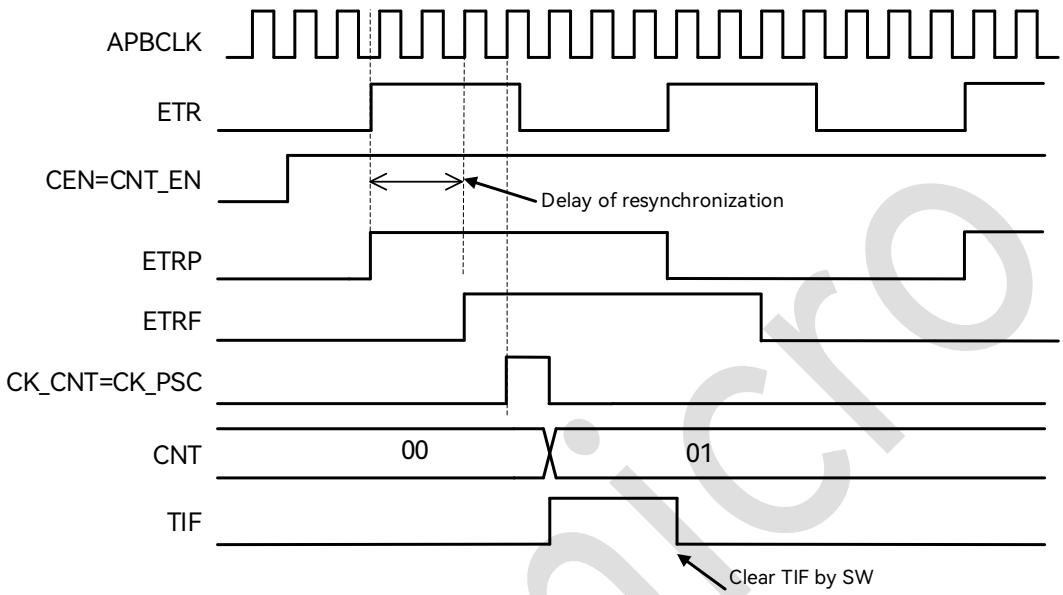

Figure 19-20: Timing 1 Diagram in External Clock Source Mode 2

The main difference from external clock source mode 1 is that the ETR input is directly divided and then filtered to generate CK_PSC clock, which means that the application scenarios where the ETR input frequency is higher than timerx_clk can be supported, in which case the ETR input shall be pre-divided first before it is used to drive the counter.

The configuration required for this mode is as follows:

1. In GPIO module, configure the corresponding pin as TIM_ETR.
2. Select the ETP edge by setting TIM_SMCR[15] = 0.
3. Set the ETR division ratio by setting TIM_SMCR.ETPS[1:0] = 01.
4. Configure the input filter duration by setting TIM_SMCR.ETF[3:0] = 0000.
5. Set the ECE register and enable the external clock source mode 2 by setting TIM_SMCR[14] = 1 and TIM_SMCR[2:0] = 000.
6. Enable the counter by setting TIM_CR1[0] = 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The following diagram shows an example of typical external clock source mode 2:

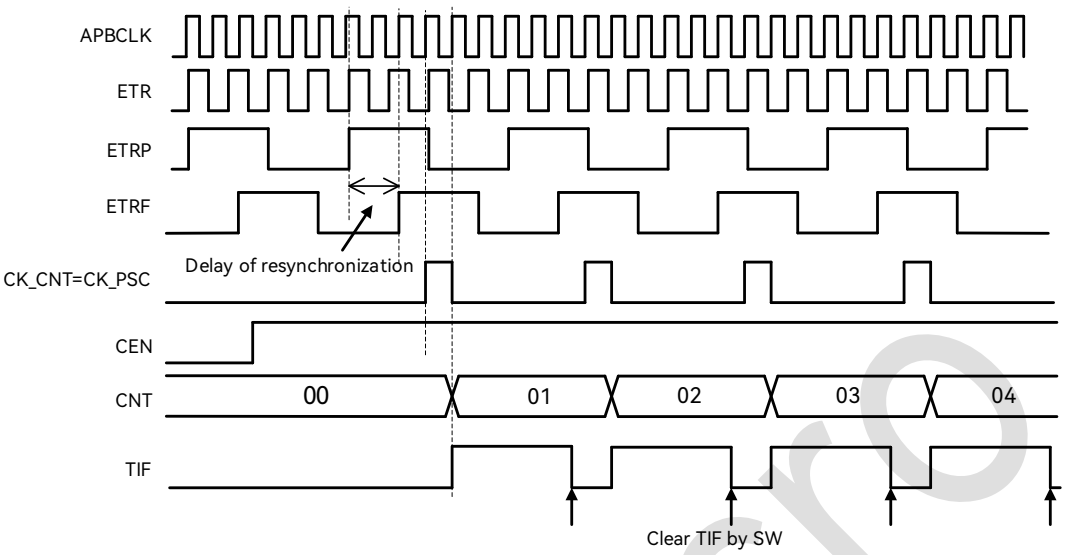

Figure 19-21: Timing 2 Diagram in External Clock Source Mode 2

In external clock source mode 2, TIM can still be configured as slave mode: For example, ETR input is used for counting while TRGO of another timer is used as the trigger signal, and the reset counter restarts counting at the arrival of trigger event.

19.5.5 Internal Trigger Signal (ITRx)

TIM supports four internal trigger inputs, which can be used for counting trigger or internal signal capture. For internal signal capture, it is required to configure TS to 000−011 for selecting ITR0−ITR3, and configure CCxS to 11 for selecting TRC as the capture signal.

Each ITR input supports 4 internal signal extensions configured by the ITRxSEL register.

19.5.6 Capture/Compare Channels

TIM consists of 4 capture/compare channels, each of which is built around a capture/compare register CCR (including a shadow register), an input stage for capture and an output stage for compare.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The input stage samples the corresponding TIx input to generate a filtered signal TIxF. Then, an edge detection with polarity selection generates a signal (TIxFPx), which can be used as trigger input for counting or as the capture command and is prescaled before being captured.

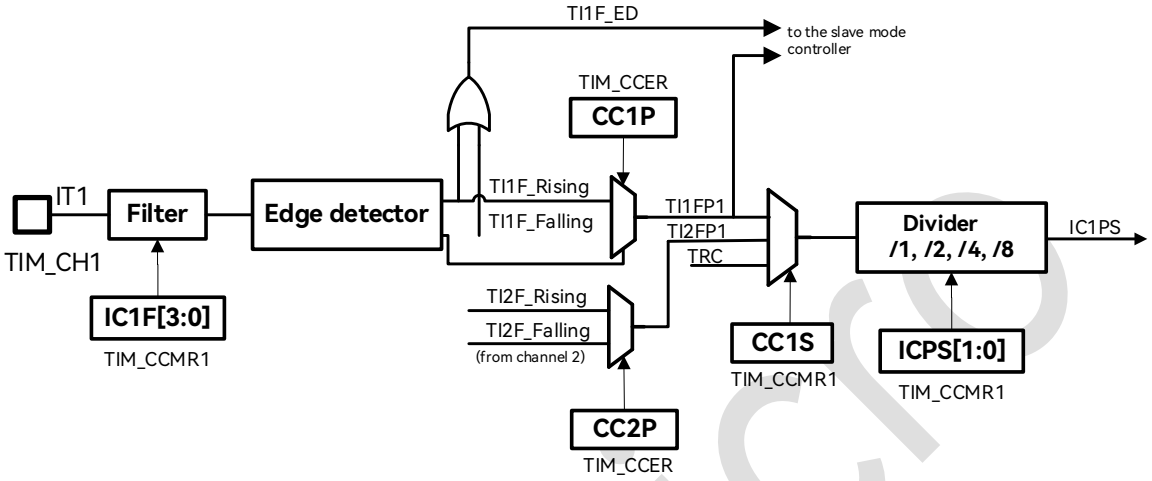

Figure 19-22: Capture/Compare Channel (Channel 1 Input Stage)

The output stage generates an output reference signal OCxREF, which is fixed to be active high and acts as the reference input to the final output circuit. Wherein, channels 1–3 support complementary output and dead-time insertion, while channel 4 is relatively simple and does not support complementary output.

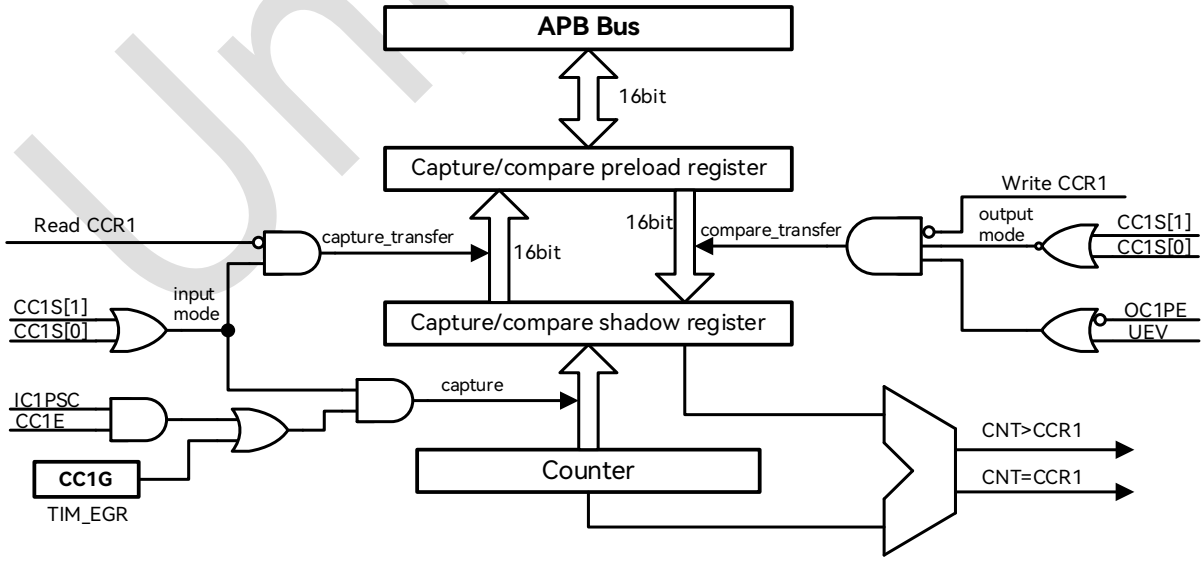

Figure 19-23: Capture/Compare Channel 1 Main Circuit

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

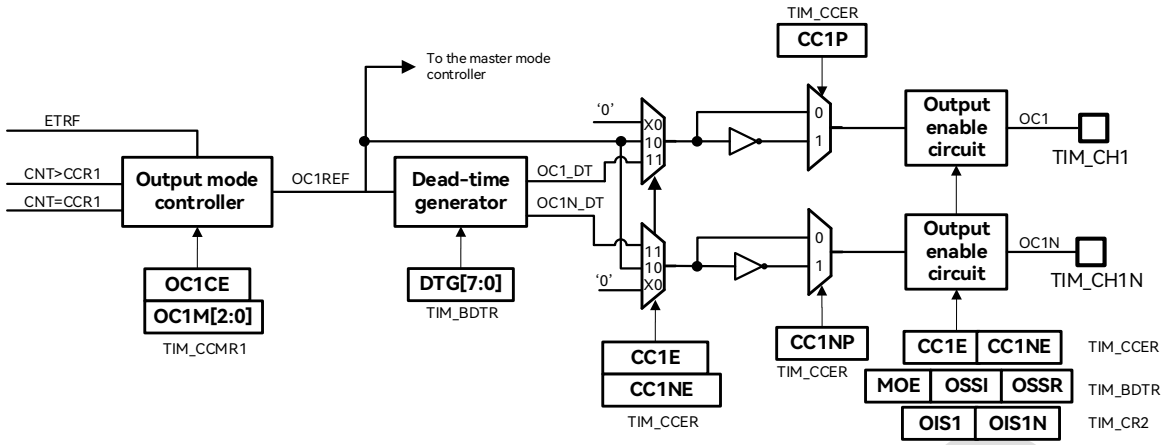

Figure 19-24: Output Stage of Capture/Compare Channel (Channels 1–3)

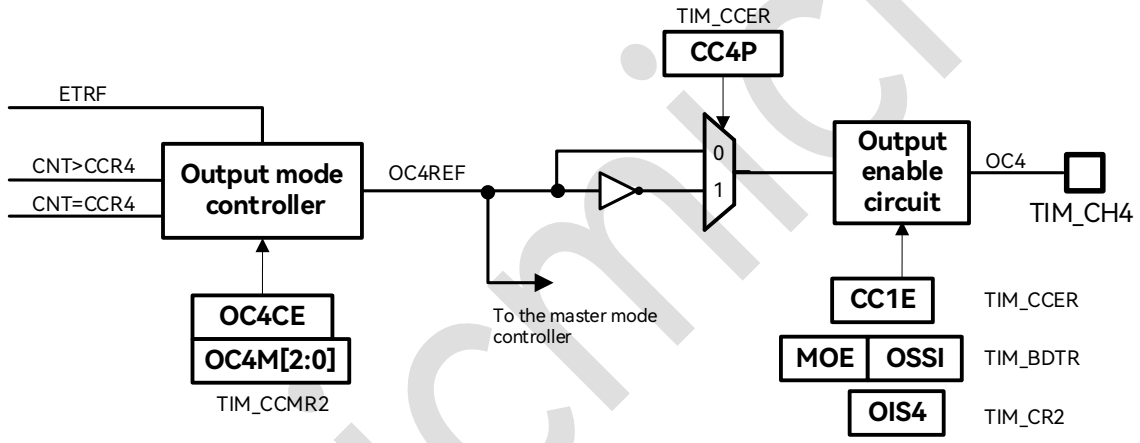

Figure 19-25: Output Stage of Capture/Compare Channel (Channel 4)

The capture/compare block is made of one preload register and one shadow register. Write and read always access the preload register。 In capture mode, the capture value is saved in the shadow register and copied into the preload register. In compare mode, the content of the preload register is copied into the shadow register for comparison with the counter.

19.5.7 Input Capture Mode

When the expected level transition is detected by the ICx signal, a capture is triggered, and the current counter value is latched into CCR. At the same time, the CCxIF interrupt flag is set and a corresponding interrupt or a DMA request can be triggered. If a capture occurs while the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

CCxIF flag is already high, then the over-capture flag CCxOF is set (the last capture value in CCR is overwritten). CCxIF can be cleared by software or automatically cleared by reading the CCR register. CCxOF can be cleared by software writing it to 1.

The input capture of PWM signals can be realized through the cooperation of two or more channels. For example, to calculate the period and duty cycle of an input signal, input the signal from TI1 pin, and take the rising edge and falling edge of the filtered signal inside the chip to obtain TI1FP1 and TI1FP2 respectively. Input TI1FP1 into capture channel 1 and TI1FP2 into channel 2 to realize captures of input signals at rising edge by channel 1 and at falling edge by channel 2. After the capture interrupt occurs periodically, the software can calculate the period and duty cycle of the input signal through the values of CCR1 and CCR2 registers.

To capture the counter value to the TIM_CCR1 register on the rising edge of the TI1 input, the configuration steps are as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01, with IC1 mapped on TI1.
4. Select the active counting edge to be rising edge or falling edge by setting TIM_CCER[1].
5. Configure the input filter duration by setting the IC1F[3:0] bits in the TIM_CCMR1 register.
6. Configure the input prescaler by setting the IC1PS[1:0] bits in the TIM_CCMR1 register.
7. Enable the channel by setting TIM_CCER[0] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

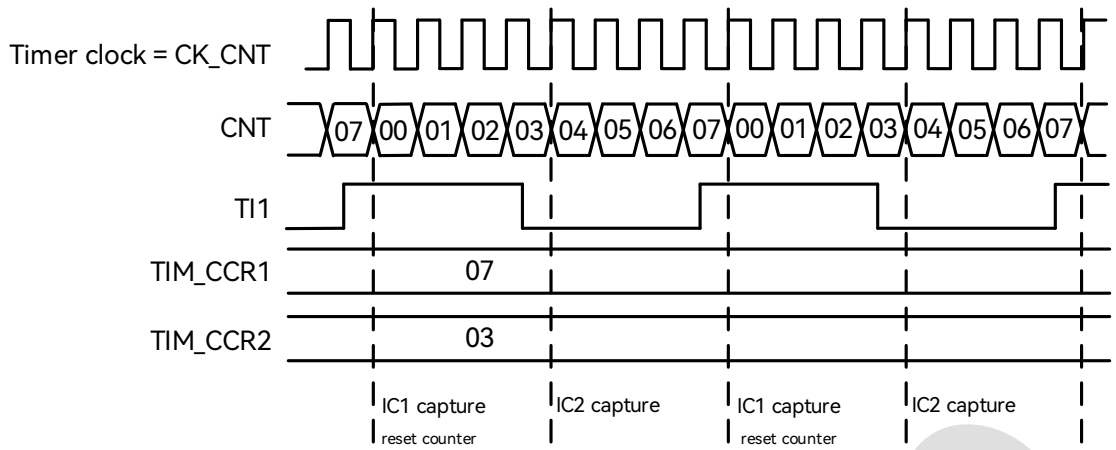

Figure 19-26: PWM Input Capture Mode Timing Diagram

The following settings are required for PWM input capture:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 and TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01 and TIM_CCMR1[9:8] = 10, with the two signals IC1 and IC2 mapped on the same TI1 input.
4. Select the active counting edge by setting TIM_CCER[1] = 0 and TIM_CCER[5] = 1, with the two signals IC1 and IC2 active on edges with opposite polarities.
5. Configure the input filter duration by setting the IC1F[3:0] and IC2F[3:0] bits in the TIM_CCMR1 register.
6. Configure the input prescaler by setting the IC1PS[1:0] and IC2PS[1:0] bits in the TIM_CCMR1 register.
7. Select the trigger input source by setting TIM_SMCR[6:4][2:0] = 101.
8. Configure the slave mode controller to reset mode by setting TIM_SMCR.SMS[2:0] = 100.
9. Enable the channel by setting TIM_CCER[0] = 1 and TIM_CCER[4] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.8 Forced Output Mode

In output compare mode, the OCxREF signal can be forced to active or inactive level directly by software, independently of any comparison between the CCR and the counter.

The OCxREF signal can be forced to be active (OCxREF is always active high) by writing OCxM = 101, and forced to be inactive (low level) by writing OCxM = 100. Anyway, the comparison between CCR and the counter is still performed.

19.5.9 Output Compare Mode

In output compare mode, when a match is found between the capture/compare register CCR and the counter, the OCxREF can be set to be active, inactive or to toggle on match. At the same time, the interrupt flag is also set and DMA requests can be sent (overwriting the configuration register).

The output compare can also be used to output a pulse signal of a specific width (in one-pulse mode).

Procedure:

1. Select the counter clock (internal, external, prescaler).
2. Write the desired data to the ARR and CCR registers.
3. Set the interrupt enable bit and DMA enable bit as required.
4. Select the output mode.
5. Enable the counter.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

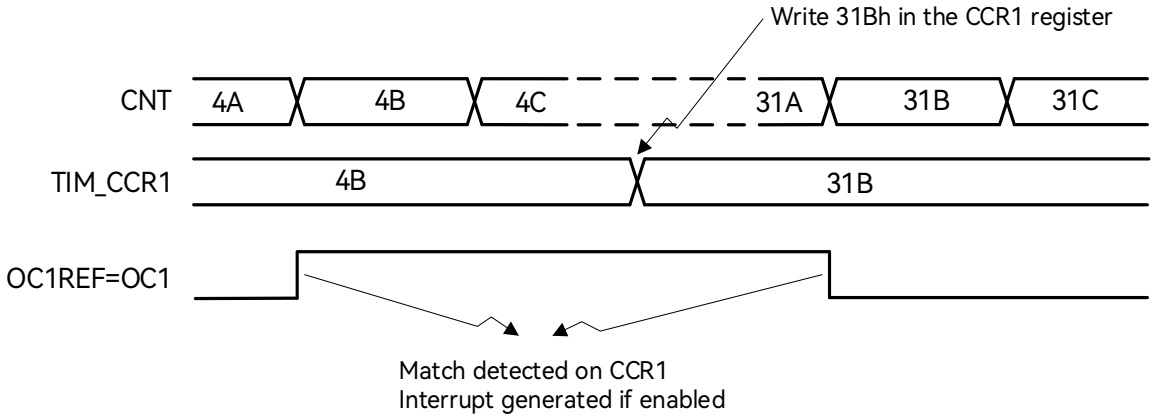

Figure 19-27: Output Compare Mode, Toggle on OC1

The CCR register can be updated at any time by software to control the output waveform, provided that the preload register is not enabled. Otherwise, the CCR shadow register is only updated with the content of the preload register at the next update event.

19.5.10 PWM Output

PWM mode allows you to generate a pulse width modulation signal with a frequency determined by the value of the ARR register and a duty cycle determined by the value of the CCR register.

The polarity of the output signal is software programmable using the CCxP bit in the register. In PWM mode, CNT and CCR registers are always compared. The timer is able to generate PWM in edge-aligned mode or center-aligned mode.

19.5.10.1 PWM Edge-aligned Mode

In up-counting mode, when it is configured in PWM mode 1, the OCxREF signal is high as long as CNT < CCR, otherwise it is low. And OCxREF will be held at 1 if CCR > ARR and held at 0 if CCR is 0.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

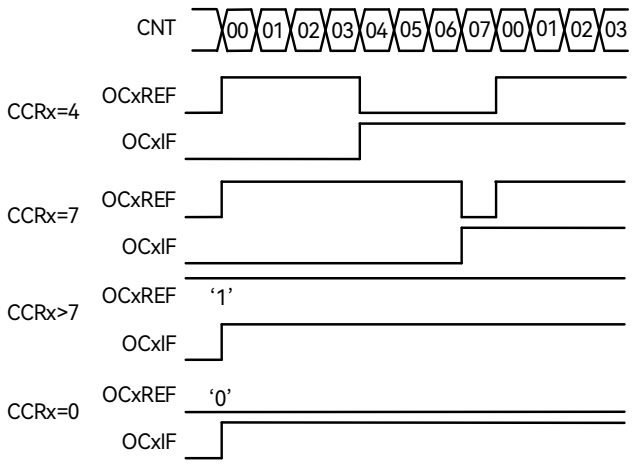

Figure 19-28: Edge-aligned PWM Waveform (ARR = 7)

In down-counting mode, the definition of OCxREF level is the same as that in up-counting mode.

19.5.10.2 PWM Center-aligned Mode

The definition of OCxREF level is the same as that in edge-aligned mode. The figure below is an example.

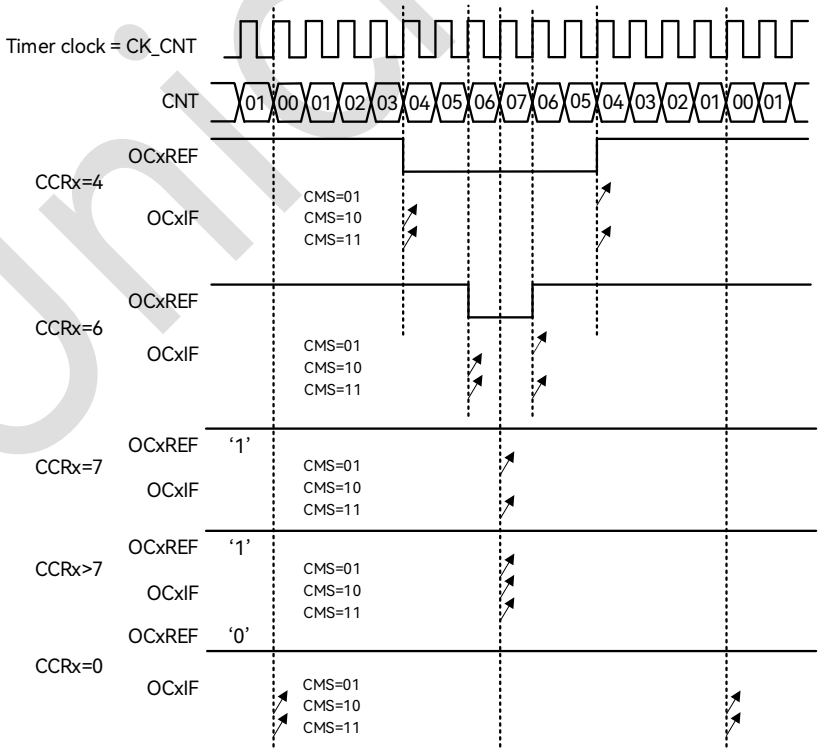

Figure 19-29: Center-aligned PWM Waveform (ARR = 7)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When start counting in center-aligned mode, the initial counting direction is determined by the DIR bit in the register, and in the subsequent process, the DIR bit is directly controlled by hardware. The safest way to use center-aligned mode is to generate an update by setting the UG bit in the register just before starting the counter and not to overwrite the counter while it is running.

19.5.11 One-pulse Output Mode

One-pulse output mode is a particular case of the compare output mode, which allows the counter to generate a pulse with a programmable length after a programmable delay following the occurrence of an event.

Different from other output modes, the counter will stop automatically at the next update event. A pulse can be correctly generated only if the compare value is different from the counter initial value. In up-counting, it is required that CNT < CCR ≤ ARR; in down-counting, it is required that CNT > CCR.

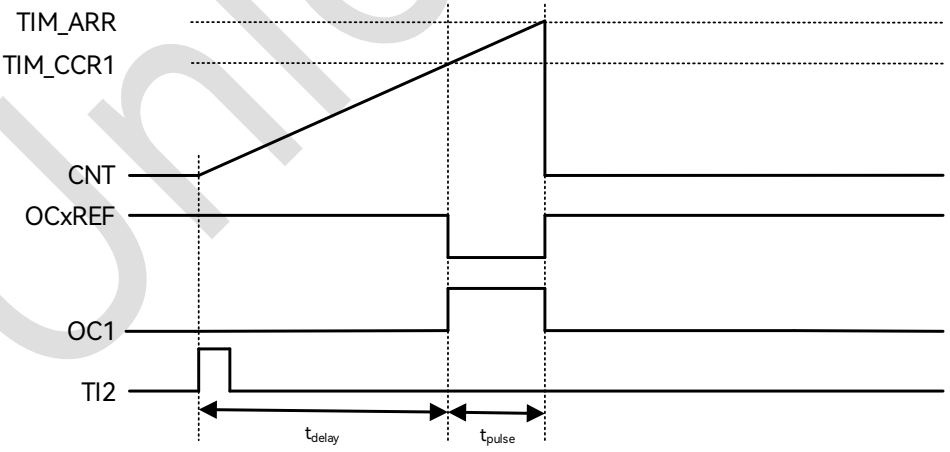

Figure 19-30: Timing Diagram of One-pulse Mode

In the above figure, TI2 input is used as the counter trigger signal. When the count value reaches CCR, the OCxREF outputs a low level. Once the counter counts up to ARR, the OCxREF signal returns to a high level, and the counter rolls back to 0, stopping the counting process.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- The configuration for realizing the above function of TI2 as an input trigger is as follows:

  1. In GPIO module, configure the corresponding pin as TIM_CH2.
  2. Disable the channel by setting TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
  3. Select the input channel by setting TIM_CCMR1[9:8] = 01.
  4. Select the active counting edge by setting TIM_CCER[5] = 0.
  5. Select TI2FP2 as the trigger input source by setting TIM_SMCR.TS[2:0] = 110.
  6. Set the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110, with TI2FP2 for activating the counter.
  7. Enable the channel by setting TIM_CCER[4] = 1.

- The configuration for realizing the above function of OC1 as an output is as follows:

  1. In GPIO module, configure the corresponding pin as TIM_CH1.
  2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
  3. Select the output channel by setting TIM_CCMR1[1:0] = 00.
  4. Select the active counting edge by setting TIM_ CCMR1.OC1M = 111, in PWM mode 2.
  5. Enable the channel by setting TIM_CCER[0] = 1.

- Special settings for generating OPM waveform timing:

  1. t_{delay} is determined by the value of TIM_CCR1.
  2. t_{pulse} is determined by the difference between TIM_ARR and TIM_CCR1 (TIM_ARR−TIM_CCR1).
  3. Configure to one-pulse mode by setting TIM_CR1.OPM = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.12 Clearing OCxREF Signal on External Event

OCxREF is active at high level, and it can be pulled down directly until the next update event by applying a high level to the external ETR pin. This function can only be used in output compare and PWM modes, and does not work in forced mode. Enabling this function requires setting OCxCE to 1.

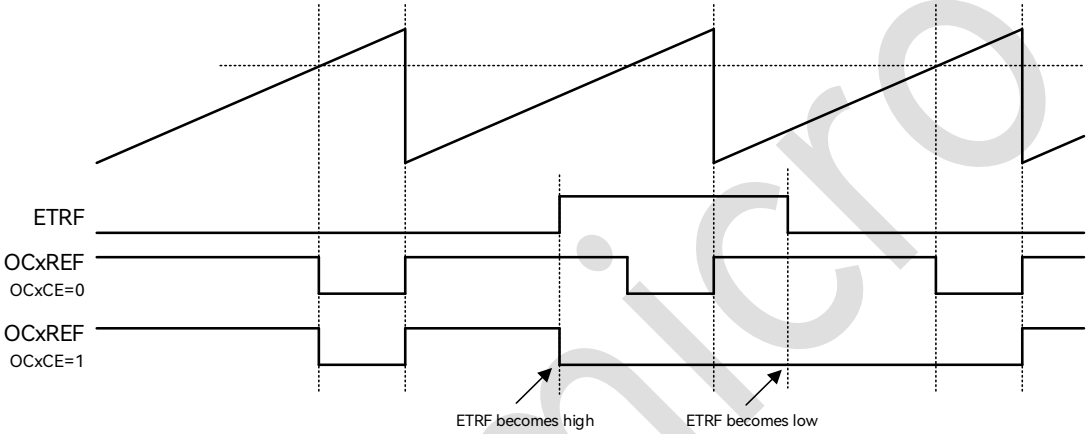

Figure 19-31: ETR Signal Clearing OCxREF of TIM

19.5.13 Encoder Interface Mode

The encoder interface mode involves two external input signals. The TIM determines whether to count up or down according to the edge of one signal relative to the level of the other signal. The following table shows the relationship between the counting mode and the two inputs:

Table 19-2: Counting Direction versus Encoder Signals

Active Edge	Level on Opposite Signal (TI1 for TI2, TI2 for TI1)	TI1 Signal		TI2 Signal	
		Rising	Falling	Rising	Falling
Counting on TI1 only	High	Down	Up	No count	No count
	Low	Up	Down	No count	No count
Counting on TI2 only	High	No count	No count	Up	Down
	Low	No count	No count	Down	Up
Counting on TI1 and TI2	High	Down	Up	Up	Down
	Low	Up	Down	Down	Up

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

For example, when the counter is counting on TI1, it will count down if TI2 is sampled as high level on the rising edge of TI1, and count up if TI2 is sampled as high level on the falling edge of TI1.

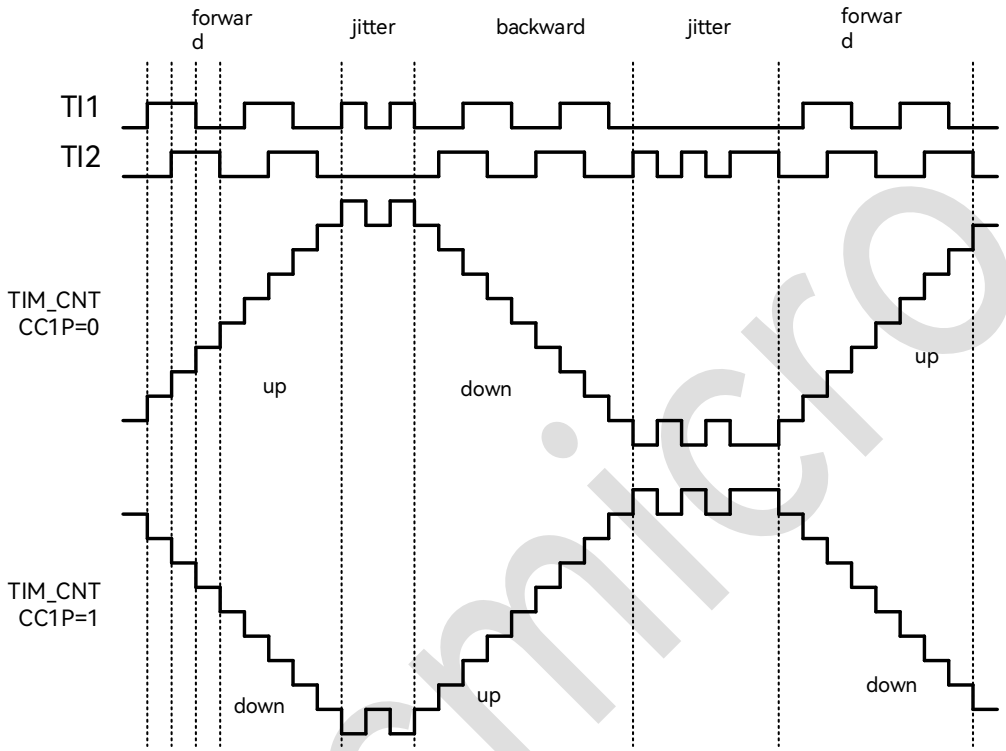

Figure 19-32: Example of Counter Operation in Encoder Interface Mode

The input channels in encoder interface mode shall be set as follows:

1. In GPIO module, configure the corresponding pins with TIM_CH1 and TIM_CH2 functions.
2. Disable the channel by setting TIM_CCER[0] = 0 and TIM_CCER[4] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01 and TIM_CCMR1[9:8] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0 and TIM_CCER[5] = 0.
5. Set the slave mode controller to encoder mode 3 by setting TIM_SMCR.SMS[2:0] = 011.
6. Enable the channel by setting TIM_CCER[0] = 1 and TIM_CCER[4] = 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.14 TIM slave mode:

When TIM is used as a slave (triggered by an external event), it can be configured to operate in three modes: reset mode, gated mode, and trigger mode.

19.5.14.1 Reset Mode

In this mode, all the preload registers in TIM will be reinitialized in response to an event on a trigger input, and the counter will restart from 0. The following figure shows that the counter behaves normally until rising edge is detected on TI1 input, at which time the counter is cleared and restarts from 0.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to reset mode by setting TIM_SMCR.SMS[2:0] = 100.
7. Enable the channel by setting TIM_CCER[0] = 1.
8. Enable the counter by setting TIM_CR1[0] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

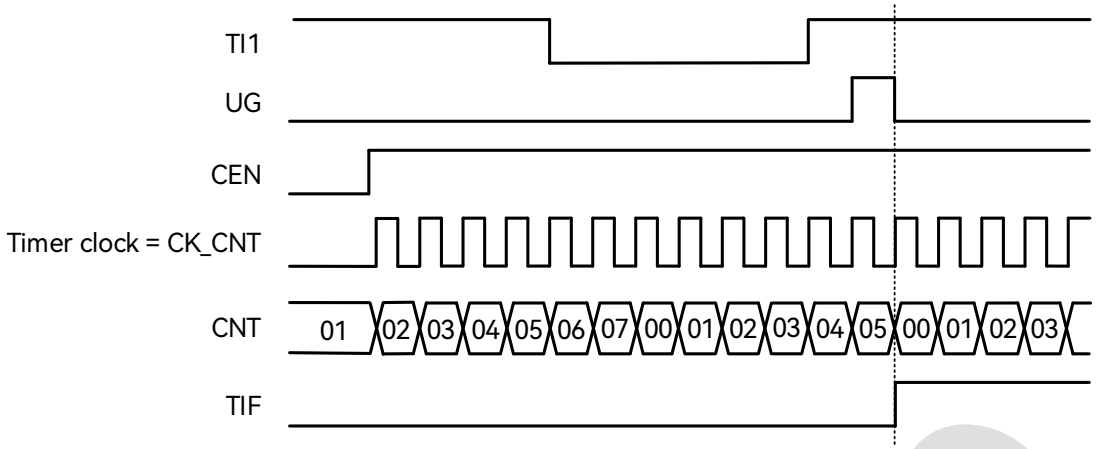

Figure 19-33: Timing Diagram in Reset Mode

19.5.14.2 Gated Mode

In this mode, the counter can be enabled depending on the level of a selected input. The interrupt flag is triggered whenever a level shift causes the counter to start or stop counting.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to gated mode by setting TIM_SMCR.SMS[2:0] = 101.
7. Enable the channel by setting TIM_CCER[0] = 1.
8. Enable the counter by setting TIM_CR1[0] = 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

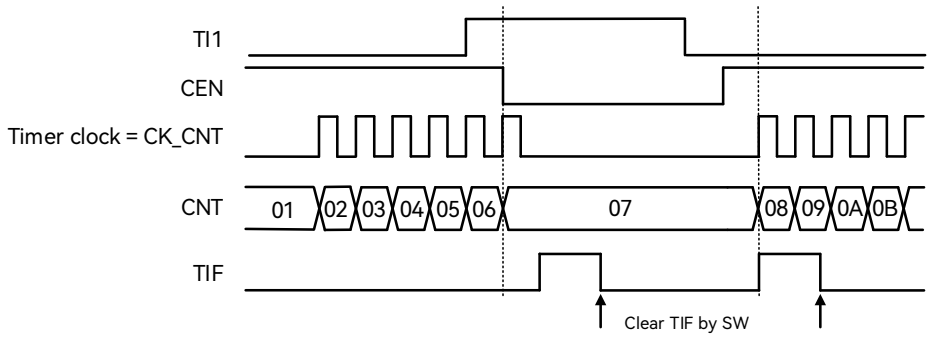

Figure 19-34: Timing Diagram in Gated Mode

19.5.14.3 Trigger Mode

The counter can start in response to an event on a selected input.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pin as TIM_CH1.
2. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
3. Select the input channel by setting TIM_CCMR1[1:0] = 01.
4. Select the active counting edge by setting TIM_CCER[1] = 0.
5. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
6. Configure the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110.
7. Enable the channel by setting TIM_CCER[0] = 1.

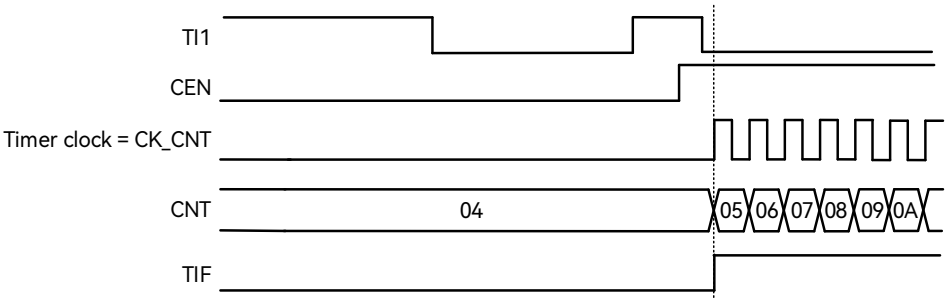

Figure 19-35: Timing Diagram in Trigger Mode

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.14.4 External Clock Mode 2 + Trigger Mode

In this mode, ETR can be set as the counting clock, while another external input is used as a trigger signal to start the counter. For instance, the counter begins counting on the rising edge of the ETR input after detecting the rising edge of TI1.

The configuration in the following figure example is as follows:

1. In GPIO module, configure the corresponding pins as TIM_CH1 and TIM_ETR.
2. Select the ETP edge by setting TIM_SMCR[15] = 0.
3. Set the ETR division ratio by setting TIM_SMCR.ETPS[1:0] = 01.
4. Configure the input filter duration by setting TIM_SMCR.ETF[3:0] = 0000.
5. Set the ECE register and enable the external clock mode 2 by setting TIM_SMCR[14] = 1.
6. Disable the channel by setting TIM_CCER[0] = 0 to ensure the success of subsequent channel configuration.
7. Select the input channel by setting TIM_CCMR1[1:0] = 01.
8. Select the active counting edge by setting TIM_CCER[1] = 0.
9. Select TI1FP1 as the trigger input source by setting TIM_SMCR.TS[2:0] = 101.
10. Configure the slave mode controller to trigger mode by setting TIM_SMCR.SMS[2:0] = 110.
11. Enable the channel by setting TIM_CCER[0] = 1.

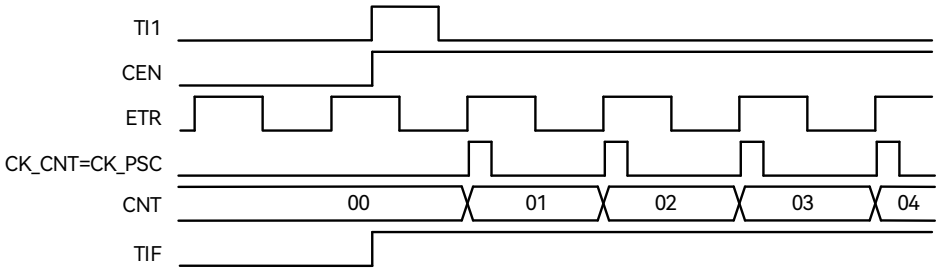

Figure 19-36: Timing Diagram in External Clock Mode 2 + Trigger Mode

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.5.15 Timer Synchronization

Timers can be cascaded together through trigger events to achieve synchronization or cascading operation. A timer can utilize four internal trigger inputs, allowing the trigger signal output from one timer to connect to the internal trigger input of other timers.

19.5.16 DMA Access

TIM supports 7 types of DMA requests, namely 4 CC channel requests, external trigger request, user software trigger request and COM trigger request.

Each CC channel generates a DMA request, which is used to transfer the content of CCRx to RAM in capture mode, and to write the data in RAM to CCRx in compare mode. The DMA request can be configured as single-transfer or burst-transfer (CCxBURSTEN), wherein the former accesses only the CCRx register, while the latter accesses a specific set of registers based on the DCR register configuration.

In addition, DMA requests can also be generated from external trigger event, software trigger event and COM trigger event, and at the occurrence of these requests, DMA burst transfer will be started to write data to one or more registers within TIM or to read one or more register values from TIM.

Table 19-3: Seven DMA Requests Supported by TIM

DMA Request	CCxBURSTEN	DMA.CHxCTRL.DIR	DMA Access Object	Single-transfer Length
TIM_CH1	0	0	Read CCR1	1
		1	Write CCR1	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_CH2	0	0	Read CCR2	1
		1	Write CCR2	
	1	0	Read DMAR	DBL

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

DMA Request	CCxBURSTEN	DMA.CHxCTRL.DIR	DMA Access Object	Single-transfer Length
		1	Write DMAR	
TIM_CH3	0	0	Read CCR3	1
		1	Write CCR3	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_CH4	0	0	Read CCR4	1
		1	Write CCR4	
	1	0	Read DMAR	DBL
		1	Write DMAR	
TIM_TRIG	N/A	0	Read DMAR	DBL
		1	Write DMAR	
TIM_UEV	N/A	0	Read DMAR	DBL
		1	Write DMAR	
TIM_COM	N/A	0	Read DMAR	DBL
		1	Write DMAR	

19.5.17 DMA Burst

TIM supports DMA and DMA-burst access. A DMA request can be generated at a specific event, so as to write the capture result in CCR to RAM or write the content of one or more registers in RAM to the preload register in TIM.

DMA-burst allows to generate multiple successive DMA requests upon a single event. The main purpose is to update the content of multiple registers in a row each time a given timer event is triggered, thus making it possible to dynamically modify the output waveform in real time.

The DMA controller destination is unique and must be directed to the virtual register TIM_DMAR. On a given timer event, the timer launches a sequence of DMA requests (burst). Each DMA write access to the TIM_DMAR register will be redirected to the actual function register by TIM.

The DBL bits in the register set the DMA burst length, and the DBA bits define the base address for DMA access to TIM (an offset starting from the address of the TIM_CR register).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

In DMA-burst mode, all DMA access shall be directed to the virtual register DMAR, and TIM automatically accumulates the internal offset address according to the access. The DBA bits in the register are used to specify the destination address of the first DMA transfer within TIM, while the DBL bits are used to specify the burst length.

19.5.18 Input XOR Function

The XOR output of the input signals of channels 1–3 can be connected to the filter and edge circuit input of channel 1 for input capture or trigger.

The TI1S bit in the TIM_CR2 register is used to select whether the input to channel 1 comes from the XOR of the three channel inputs.

19.5.19 Debug Mode

When the CPU enters debug mode, the timer can either stop or continue working, and its behavior is defined by registers in the chip system.

When the timer is stopped during debugging, its output will be disabled (MOE is cleared). Depending on the register configuration, the output signal can be forced to be inactive or controlled by the GPIO module.

19.6 Register Description

TIM2 register base address: 0x4600_A000

TIM3 register base address: 0x4600_A400

TIM4 register base address: 0x4600_A800

TIM5 register base address: 0x4600_AC00

TIM9 register base address: 0x4700_9000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

TIM10 register base address: 0x4700_A000

TIM11 register base address: 0x4700_B000

TIM12 register base address: 0x4600_D000

TIM13 register base address: 0x4600_E000

TIM14 register base address: 0x4600_F000

The registers are listed below:

Table 19-4: General-purpose Timers (TIM2 ~ TIM5 & TIM9 ~ TIM14)

Offset Address	Name	Description
0x00	TIM_CR1	Control register 1
0x04	TIM_CR2	Control register 2
0x08	TIM_SMCR	Slave mode control register
0x0C	TIM_DIER	DMA and interrupt enable register
0x10	TIM_SR	Status register
0x14	TIM_EGR	Event generation register
0x18	TIM_CCMR1	Capture/compare register 1
0x1C	TIM_CCMR2	Capture/compare register 2
0x20	TIM_CCER	Capture/compare enable register
0x24	TIM_CNT	Counter register
0x28	TIM_PSC	Prescaler register
0x2C	TIM_ARR	Auto-reload register
0x34	TIM_CCR1	Capture/compare register 1
0x38	TIM_CCR2	Capture/compare register 2
0x3C	TIM_CCR3	Capture/compare register 3
0x40	TIM_CCR4	Capture/compare register 4
0x48	TIM_DCR	DMA control register
0x4C	TIM_DMAR	DMA access register

Registers are detailed in the following sections.

19.6.1 Control Register 1 (TIM_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:10	RSV	-	-	Reserved
9:8	CKD	R/W	0x0	Dead time and digital filter clock frequency division register (division ratio relative to CK_INT): 00: $t_{DTS} = t_{CK_INT}$ 01: $t_{DTS} = 2 * t_{CK_INT}$ 10: $t_{DTS} = 4 * t_{CK_INT}$ 11: Reserved, prohibited
7	APRE	R/W	0x0	Auto-reload preload enable: 0: ARR not preloaded 1: ARR preloaded
6:5	CMS	R/W	0x0	Counter alignment mode selection: 00: Edge-aligned mode 01: Center-aligned mode 1; output compare interrupt flags are set only when the counter is counting down. 10: Center-aligned mode 2; output compare interrupt flags are set only when the counter is counting up. 11: Center-aligned mode 3; output compare interrupt flags are set both when the counter is counting up or down.
4	DIR	R/W	0x0	Counting direction register: 0: Count up 1: Count down Note: This register is read-only when the timer is configured in center-aligned mode or encoder mode.
3	OPM	R/W	0x0	One-pulse mode output: 0: The counter does not stop at the occurrence of update event. 1: The counter stops at the occurrence of update event (CEN cleared automatically).
2	URS	R/W	0x0	Update request source: 0: An update interrupt or DMA request will be generated by any of the following events: ● Counter overflow/underflow

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				● Software setting the UG bit ● Update generated from the slave mode controller 1: An update interrupt or DMA request will be generated only at counter overflow or underflow.
1	UDIS	R/W	0x0	Update disable: 0: Update event enabled; the update event can be generated by any of the following events: ● Counter overflow/underflow ● Software setting the UG bit ● Update generated from the slave mode controller 1: Update event disabled, shadow register not updated; the counter and the prescaler will be reinitialized if the UG bit is set or if the slave mode controller receives a hardware reset.
0	CEN	R/W	0x0	Counter enable: 0: Disabled 1: Enabled Note: The external trigger mode can automatically set the CEN bit.

19.6.2 Control Register 2 (TIM_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	TI1S	R/W	0x0	T1 input selection: 0: T1 input from CH1 pin 1: T1 input from XOR combination of CH1, CH2 and CH3 pins
6:4	MMS	R/W	0x0	Master mode selection, selecting the TRGO trigger mode:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				000: Reset—TRGO is generated by the UG bit in the EGR register. 001: Enable—TRGO is generated by the counter enable signal, including the CEN control bit and external trigger. 010: Update—TRGO is generated by the update event. 011: Compare pulse—TRGO is generated when an input capture or compare event occurs that sets CC1F to 1. 100: Compare—TRGO is generated by OC1REF. 101: Compare—TRGO is generated by OC2REF. 110: Compare—TRGO is generated by OC3REF. 111: Compare—TRGO is generated by OC4REF.
3	CCDS	R/W	0x0	Capture/compare DMA selection: 0: CCx DMA request sent when CCx event occurs 1: CCx DMA request sent when update event occurs
2:0	RSV	-	-	Reserved

19.6.3 Slave Mode Control Register (TIM_SMCR)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	ETP	R/W	0x0	External trigger polarity: 0: Active at high level or rising edge 1: Active at low level or falling edge
14	ECE	R/W	0x0	Enable clock enable: 0: External clock mode 2 disabled 1: External clock mode 2 enabled; the counter is clocked by any active edge on the ETRF signal.
13:12	ETPS	R/W	0x0	External trigger prescaler:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				The frequency of external trigger signal ETRP must be at most 1/4 of TIM clock frequency. A prescaler can be enabled to reduce ETRP frequency when inputting fast external clocks. 00: Prescaler off 01: Frequency divided by 2 10: Frequency divided by 4 11: Frequency divided by 8
11:8	ETF	R/W	0x0	External trigger filter frequency and length selection: 0000: No filter 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS}$ / 2, N = 6 0101: $f_{SAMPLING} = f_{DTS}$ / 2, N = 8 0110: $f_{SAMPLING} = f_{DTS}$ / 4, N = 6 0111: $f_{SAMPLING} = f_{DTS}$ / 4, N = 8 1000: $f_{SAMPLING} = f_{DTS}$ / 8, N = 6 1001: $f_{SAMPLING} = f_{DTS}$ / 8, N = 8 1010: $f_{SAMPLING} = f_{DTS}$ / 16, N = 5 1011: $f_{SAMPLING} = f_{DTS}$ / 16, N = 6 1100: $f_{SAMPLING} = f_{DTS}$ / 16, N = 8 1101: $f_{SAMPLING} = f_{DTS}$ / 32, N = 5 1110: $f_{SAMPLING} = f_{DTS}$ / 32, N = 6 1111: $f_{SAMPLING} = f_{DTS}$ / 32, N = 8
7	MSM	R/W	0x0	Master/slave mode selection: 0: No action 1: The effect of an event on the trigger input (TRGI) is delayed to allow a perfect synchronization between the current timer and its slaves (through TRGO).
6:4	TS	R/W	0x0	Trigger selection for selecting the trigger source to be used to synchronize the counter: 000: Internal trigger 0 (ITR0) 001: Internal trigger 1 (ITR1) 010: Internal trigger 2 (ITR2)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				011: Internal trigger 3 (ITR3) 100: TI1 edge detector (TI1F_ED) 101: Filtered timer input 1 (TI1FP1) 110: Filtered timer input 2 (TI2FP2) 111: External trigger input (ETRF) Note: These bits can be changed only when the slave mode is disabled (i.e. SMS = 000).
3	RSV	-	-	Reserved
2:0	SMS	R/W	0x0	Slave mode selection: 000: Slave mode disabled—if CEN is enabled, then the prescaler is clocked directly by the internal clock. 001: Encoder mode 1—counter counts up/down on TI2FP1 edge depending on TI1FP2 level. 010: Encoder mode 2—counter counts up/down on TI1FP2 edge depending on TI2FP1 level. 011: Encoder mode 3—counter counts up/down on both TI1FP1 and TI2FP2 edges depending on the level of other inputs. 100: Reset mode—rising edge of the selected trigger input (TRGI) reinitializes the counter and generates an update of the registers. 101: Gated mode—the counter clock is enabled when TRGI is high, and stops as soon as it becomes low. 110: Trigger mode—the counter starts at the rising edge of TRGI (but it is not reset). 111: External clock mode 1—rising edges of TRGI directly clock the counter.

19.6.4 DMA / Interrupt Enable Register (TIM_DIER)

Offset address: 0x0C

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	Reserved
19	CC4OF_DISABLE	R/W	0x0	CC4OF interrupt enable: 0: Enabled 1: Disabled
18	CC3OF_DISABLE	R/W	0x0	CC3OF interrupt enable: 0: Enabled 1: Disabled
17	CC2OF_DISABLE	R/W	0x0	CC2OF interrupt enable: 0: Enabled 1: Disabled
16	CC1OF_DISABLE	R/W	0x0	CC1OF interrupt enable: 0: Enabled 1: Disabled
15	RSV	-	-	Reserved
14	TDE	R/W	0x0	External trigger DMA request enable: 0: In slave mode, external trigger DMA request disabled 1: In slave mode, external trigger DMA request enabled (can be used to automatically update the preload register)
13	RSV	-	-	Reserved
12	CC4DE	R/W	0x0	Capture/compare channel 4 DMA request enable: 0: CC4 DMA request disabled 1: CC4 DMA request enabled
11	CC3DE	R/W	0x0	Capture/compare channel 3 DMA request enable: 0: CC3 DMA request disabled 1: CC3 DMA request enabled
10	CC2DE	R/W	0x0	Capture/compare channel 2 DMA request enable: 0: CC2 DMA request disabled 1: CC2 DMA request enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
9	CC1DE	R/W	0x0	Capture/compare channel 1 DMA request enable: 0: CC1 DMA request disabled 1: CC1 DMA request enabled
8	UDE	R/W	0x0	Update DMA request enable: 0: Update DMA request disabled 1: Update DMA request enabled
7	RSV	-	-	Reserved
6	TIE	R/W	0x0	Trigger interrupt enable: 0: Trigger interrupt disabled 1: Trigger interrupt enabled
5	RSV	-	-	Reserved
4	CC4IE	R/W	0x0	Capture/compare channel 4 interrupt enable: 0: CC4 interrupt disabled 1: CC4 interrupt enabled
3	CC3IE	R/W	0x0	Capture/compare channel 3 interrupt enable: 0: CC3 interrupt disabled 1: CC3 interrupt enabled
2	CC2IE	R/W	0x0	Capture/compare channel 2 interrupt enable: 0: CC2 interrupt disabled 1: CC2 interrupt enabled
1	CC1IE	R/W	0x0	Capture/compare channel 1 interrupt enable: 0: CC1 interrupt disabled 1: CC1 interrupt enabled
0	UIE	R/W	0x0	Update interrupt enable: 0: Update interrupt disabled 1: Update interrupt enabled

19.6.5 Status Register (TIM_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	CC4OF	R/W0C	0x0	Capture/compare channel 4 overcapture flag: This register is valid only when the corresponding channel is configured in input capture mode. This flag is set by hardware and cleared by software via writing it to 0. 0: No overcapture event 1: A new capture occurs while CC4IF flag is 1.
11	CC3OF	R/W0C	0x0	Capture/compare channel 3 overcapture flag: This register is valid only when the corresponding channel is configured in input capture mode. This flag is set by hardware and cleared by software via writing it to 0. 0: No overcapture event 1: A new capture occurs while CC3IF flag is 1.
10	CC2OF	R/W0C	0x0	Capture/compare channel 2 overcapture flag: This register is valid only when the corresponding channel is configured in input capture mode. This flag is set by hardware and cleared by software via writing it to 0. 0: No overcapture event 1: A new capture occurs while CC2IF flag is 1.
9	CC1OF	R/W0C	0x0	Capture/compare channel 1 overcapture flag: This register is valid only when the corresponding channel is configured in input capture mode. This flag is set by hardware and cleared by software via writing it to 0. 0: No overcapture event 1: A new capture occurs while CC1IF flag is 1.
8:7	RSV	-	-	Reserved
6	TIF	R/W0C	0x0	Trigger interrupt flag is set by hardware and cleared by software via writing it to 0.
5	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	CC4IF	R/W0C	0x0	Capture/compare channel 4 interrupt flag: If channel CC4 is configured as output: the CC4IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC4 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR4.
3	CC3IF	R/W0C	0x0	Capture/compare channel 3 interrupt flag: If channel CC3 is configured as output: the CC3IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC3 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR3.
2	CC2IF	R/W0C	0x0	Capture/compare channel 2 interrupt flag: If channel CC2 is configured as output: the CC2IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC2 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR2.
1	CC1IF	R/W0C	0x0	Capture/compare channel 1 interrupt flag: If channel CC1 is configured as output: the CC1IF flag is set when the counter matches the compare value, and cleared by software via writing it to 0. If channel CC1 is configured as input: this flag is set by hardware on a capture. It is cleared by software via writing it to 0 or automatically cleared by software reading TIM_CCR1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	UIF	R/W0C	0x0	Update interrupt flag is set by hardware and cleared by software via writing it to 0. UIF is set and the shadow register is updated at the following events: Counter overflow occurs if repetition counter = 0 and UDIS = 0. The counter is reinitialized by software setting the UG bit if URS = 0 and UDIS = 0. The counter is reinitialized by a trigger event if URS = 0 and UDIS = 0.

19.6.6 Event Generation Register (TIM_EGR)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved
6	TG	W	0x0	This bit can be set by software to generate a trigger event, and it is automatically cleared by hardware.
5	RSV	-	-	Reserved
4	CC4G	W	0x0	Capture/compare channel 4 software trigger: If channel CC4 is configured as output, CC4G flag is set, and corresponding interrupt and DMA request will be sent if enabled. If channel CC4 is configured as input: the current counter value is captured in the TIM_CCR4 register, the CC4G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
3	CC3G	W	0x0	Capture/compare channel 3 software trigger: If channel CC3 is configured as output, CC3G flag is set, and corresponding interrupt and DMA request will be sent if enabled. If channel CC3 is configured as input: the current counter value is captured in the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				TIM_CCR3 register, the CC3G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
2	CC2G	W	0x0	Capture/compare channel 2 software trigger: If channel CC2 is configured as output, CC2G flag is set, and corresponding interrupt and DMA request will be sent if enabled. If channel CC2 is configured as input: the current counter value is captured in the TIM_CCR2 register, the CC2G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
1	CC1G	W	0x0	Capture/compare channel 1 software trigger: If channel CC1 is configured as output, CC1G flag is set, and corresponding interrupt and DMA request will be sent if enabled. If channel CC1 is configured as input: the current counter value is captured in the TIM_CCR1 register, the CC1G is set, and if enabled, it can generate the corresponding interrupt and DMA request.
0	UG	W	0x0	This bit can be set by software to generate an update event, and is automatically cleared by hardware. When the software sets UG, the counter is reinitialized, the shadow register is updated, and the prescaler counter is cleared.

19.6.7 Capture/Compare Register 1 (TIM_CCMR1)

Offset address: 0x18

Reset value: 0x0000 0000

This register can be used in output (compare mode) or in input (capture mode).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Output compare mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	OC2CE	R/W	0x0	Output compare 2 clear enable, refer to OC1CE description
14:12	OC2M	R/W	0x0	Output compare 2 mode configuration, refer to OC1M description
11	OC2PE	R/W	0x0	Output compare 2 preload enable, refer to OC1PE description
10	OC2FE	R/W	0x0	Output compare 2 fast enable, refer to OC1FE description
9:8	CC2S	R/W	0x0	Capture/compare channel 2 selection: 00: CC2 channel is configured as output. 01: CC2 channel is configured as input, IC2 is mapped on TI2. 10: CC2 channel is configured as input, IC2 is mapped on TI1. 11: CC2 channel is configured as input, IC2 is mapped on TRC. Note: CC2S bits are writable only when the channel is OFF (CC2E = 0).
7	OC1CE	R/W	0x0	Output compare 1 clear enable: 0: OC1REF is not affected by ETRF input. 1: OC1REF is automatically cleared once a high level is detected on ETRF input.
6:4	OC1M	R/W	0x0	Output compare 1 mode: these bits define the behavior of the output reference signal OC1REF. 000: The comparison between the output compare register CCR1 and the counter CNT has no effect on the outputs. 001: Set OC1REF high when CCR1 = CNT (falling edge) 010: Set OC1REF low when CCR1 = CNT (falling edge)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				011: Toggle OC1REF when CCR1 = CNT (falling edge) 100: Force OC1REF low (inactive) 101: Force OC1REF high (active) 110: PWM mode 1—in up-counting, OC1REF is set high when CNT < CCR1, otherwise it is set low; in down-counting, OC1REF is set low when CNT ≥ CCR1, otherwise it is set high. 111: PWM mode 2—in up-counting, OC1REF is set low when CNT < CCR1, otherwise it is set high; in down-counting, OC1REF is set high when CNT ≥ CCR1, otherwise it is set low.
3	OC1PE	R/W	0x0	Output compare 1 preload enable: 0: Preload register disabled; CCR1 can be written directly. 1: Preload register on CCR1 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
2	OC1FE	R/W	0x0	Output compare 1 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enabled, the trigger input will immediately change OC1REF to the output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
1:0	CC1S	R/W	0x0	Capture/compare channel 1 selection: 00: CC1 channel is configured as output.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				01: CC1 channel is configured as input, IC1 is mapped on TI1. 10: CC1 channel is configured as input, IC1 is mapped on TI2. 11: CC1 channel is configured as input, IC1 is mapped on TRC. Note: CC1S bits are writable only when the channel is OFF (CC1E = 0).

- Input Capture Mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	IC2F	R/W	0x0	Input capture 2 filter
11:10	IC2PSC	R/W	0x0	Input capture 2 prescaler
9:8	CC2S	R/W	0x0	Capture/compare channel 2 selection: 00: CC2 channel is configured as output. 01: CC2 channel is configured as input, IC2 is mapped on TI2. 10: CC2 channel is configured as input, IC2 is mapped on TI1. 11: CC2 channel is configured as input, IC2 is mapped on TRC. Note: CC2S bits are writable only when the channel is OFF (CC2E = 0).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:4	IC1F	R/W	0x0	Input capture 1 filter: This bit-field defines the frequency used to sample TI1 input and the length of the digital filter applied to TI4. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8 1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5 1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
3:2	IC1PSC	R/W	0x0	Input capture 1 prescaler: 00: No prescaler 01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC1PSC register is reset when CC1E = 0.
1:0	CC1S	R/W	0x0	Capture/compare channel 1 selection: 00: CC1 channel is configured as output. 01: CC1 channel is configured as input, IC1 is mapped on TI1. 10: CC1 channel is configured as input, IC1 is mapped on TI2. 11: CC1 channel is configured as input, IC1 is mapped on TRC. Note: CC1S bits are writable only when the channel is OFF (CC1E = 0).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.6.8 Capture/Compare Register 2 (TIM_CCMR2)

Offset address: 0x1C

Reset value: 0x0000 0000

This register can be used in output (compare mode) or in input (capture mode).

- Output compare mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	OC4CE	R/W	0x0	Output compare 4 clear enable: 0: OC4REF is not affected by ETRF input. 1: OC4REF is automatically cleared once a high level is detected on ETRF input.
14:12	OC4M	R/W	0x0	Output compare 4 mode configuration: These bits define the behavior of the output reference signal OC4REF. 000: The comparison between the output compare register CCR4 and the counter CNT has no effect on the outputs. 001: Set OC4REF high when CCR4 = CNT. 010: Set OC4REF low when CCR4 = CNT. 011: Toggle OC4REF when CCR4 = CNT. 100: Force OC4REF low (inactive) 101: Force OC4REF high (active) 110: PWM mode 1 ● In up-counting, OC4REF is set high when CNT < CCR4, otherwise it is set low. ● In down-counting, OC4REF is set low when CNT > CCR4, otherwise it is set high. 111: PWM mode 2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				● In up-counting, OC4REF is set low when CNT < CCR4, otherwise it is set high. ● In down-counting, OC4REF is set high when CNT > CCR4, otherwise it is set low.
11	OC4PE	R/W	0x0	Output compare 4 preload enable: 0: Preload register disabled; CCR4 can be written directly. 1: Preload register on CCR4 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
10	OC4FE	R/W	0x0	Output compare 4 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enable is on, the trigger input will immediately change OC4REF to the output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
9:8	CC4S	R/W	0x0	Capture/compare channel 4 selection: 00: CC4 channel is configured as output. 01: CC4 channel is configured as input, IC4 is mapped on TI4. 10: CC4 channel is configured as input, IC4 is mapped on TI3. 11: CC4 channel is configured as input, IC4 is mapped on TRC. Note: CC4S bits are writable only when the channel is OFF (CC4E = 0).
7	OC3CE	R/W	0x0	Output compare 3 clear enable: 0: OC3REF is not affected by ETRF input.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: OC3REF is automatically cleared once a high level is detected on ETRF input.
6:4	OC3M	R/W	0x0	Output compare 3 mode: these bits define the behavior of the output reference signal OC3REF. 000: The comparison between the output compare register CCR3 and the counter CNT has no effect on the outputs. 001: Set OC3REF high when CCR3 = CNT. 010: Set OC3REF low when CCR3 = CNT. 011: Toggle OC3REF when CCR3 = CNT. 100: Force OC3REF low (inactive) 101: Force OC3REF high (active) 110: PWM mode 1—in up-counting, OC3REF is set high when CNT < CCR3, otherwise it is set low; in down-counting, OC3REF is set low when CNT > CCR3, otherwise it is set high. 111: PWM mode 2—in up-counting, OC3REF is set low when CNT < CCR3, otherwise it is set high; in down-counting, OC3REF is set high when CNT > CCR3, otherwise it is set low.
3	OC3PE	R/W	0x0	Output compare 3 preload enable: 0: Preload register disabled; CCR3 can be written directly. 1: Preload register on CCR3 enabled; read/write operations access the preload register; the preload value is shifted to the shadow register at each update event.
2	OC3FE	R/W	0x0	Output compare 3 fast enable: 0: Fast disabled, the trigger input will not affect the comparison output. 1: Fast enabled, the trigger input will immediately change OC3REF to the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				output when the comparison values match, regardless of the actual current comparison. This function acts only if the channel is configured in PWM1 or PWM2 mode.
1:0	CC3S	R/W	0x0	Capture/compare channel 3 selection: 00: CC3 channel is configured as output. 01: CC3 channel is configured as input, IC3 is mapped on TI3. 10: CC3 channel is configured as input, IC3 is mapped on TI4. 11: CC3 channel is configured as input, IC3 is mapped on TRC. Note: CC3S bits are writable only when the channel is OFF (CC3E = 0).

- Input Capture Mode

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	IC4F	R/W	0x0	Input capture 4 filter: This bit-field defines the frequency used to sample TI4 input and the length of the digital filter applied to TI4. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5 1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
11:10	IC4PSC	R/W	0x0	Input capture 4 prescaler: 00: No prescaler 01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC4PSC register is reset when CC4E = 0.
9:8	CC4S	R/W	0x0	Capture/compare channel 4 selection: 00: CC4 channel is configured as output. 01: CC4 channel is configured as input, IC4 is mapped on TI4. 10: CC4 channel is configured as input, IC4 is mapped on TI3. 11: CC4 channel is configured as input, IC4 is mapped on TRC. Note: CC4S bits are writable only when the channel is OFF (CC4E = 0).
7:4	IC3F	R/W	0x0	Input capture 3 filter: This bit-field defines the frequency used to sample TI3 input and the length of the digital filter applied to TI4. 0000: No filter, sampling is done at f_{DTS} 0001: $f_{SAMPLING} = f_{CK_INT}$, N = 2 0010: $f_{SAMPLING} = f_{CK_INT}$, N = 4 0011: $f_{SAMPLING} = f_{CK_INT}$, N = 8 0100: $f_{SAMPLING} = f_{DTS} / 2$, N = 6 0101: $f_{SAMPLING} = f_{DTS} / 2$, N = 8 0110: $f_{SAMPLING} = f_{DTS} / 4$, N = 6 0111: $f_{SAMPLING} = f_{DTS} / 4$, N = 8 1000: $f_{SAMPLING} = f_{DTS} / 8$, N = 6 1001: $f_{SAMPLING} = f_{DTS} / 8$, N = 8 1010: $f_{SAMPLING} = f_{DTS} / 16$, N = 5 1011: $f_{SAMPLING} = f_{DTS} / 16$, N = 6 1100: $f_{SAMPLING} = f_{DTS} / 16$, N = 8 1101: $f_{SAMPLING} = f_{DTS} / 32$, N = 5

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1110: $f_{SAMPLING} = f_{DTS} / 32$, N = 6 1111: $f_{SAMPLING} = f_{DTS} / 32$, N = 8
3:2	IC3PSC	R/W	0x0	Input capture 3 prescaler: 00: No prescaler 01: Capture is done once every 2 events 10: Capture is done once every 4 events 11: Capture is done once every 8 events The IC3PSC register is reset when CC3E = 0.
1:0	CC3S	R/W	0x0	Capture/compare channel 3 selection: 00: CC3 channel is configured as output. 01: CC3 channel is configured as input, IC3 is mapped on TI3. 10: CC3 channel is configured as input, IC3 is mapped on TI4. 11: CC3 channel is configured as input, IC3 is mapped on TRC. Note: CC3S bits are writable only when the channel is OFF (CC3E = 0).

19.6.9 Capture/Compare Enable Register (TIM_CCER)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	-	Reserved
13	CC4P	R/W	0x0	Capture/compare 4 output polarity: CC4 channel configured as output: 0: OC4 is OC4REF. 1: OC4 is the inverse OC4REF. CC4 channel configured as input: 0: Non-inverted: Capture is done on the rising edge of IC4. 1: Inverted: Capture is done on the falling edge of IC4.
12	CC4E	R/W	0x0	Capture/compare 4 output enable: CC4 channel configured as output:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: OC4 not active 1: OC4 active CC4 channel configured as input: 0: Capture disabled 1: Capture enabled
11:10	RSV	-	-	Reserved
9	CC3P	R/W	0x0	Capture/compare 3 output polarity: CC3 channel configured as output: 0: OC3 is OC3REF. 1: OC3 is the inverse OC3REF. CC3 channel configured as input: 0: Non-inverted: Capture is done on the rising edge of IC3. 1: Inverted: Capture is done on the falling edge of IC3.
8	CC3E	R/W	0x0	Capture/compare 3 output enable: CC3 channel configured as output: 0: OC3 not active 1: OC3 active CC3 channel configured as input: 0: Capture disabled 1: Capture enabled
7:6	RSV	-	-	Reserved
5	CC2P	R/W	0x0	Capture/compare 2 output polarity: CC2 channel configured as output: 0: OC2 is OC2REF. 1: OC2 is the inverse OC2REF. CC2 channel configured as input: 0: Non-inverted: Capture is done on the rising edge of IC2. 1: Inverted: Capture is done on the falling edge of IC2.
4	CC2E	R/W	0x0	Capture/compare 2 output enable: CC2 channel configured as output: 0: OC2 not active 1: OC2 active CC2 channel configured as input:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Capture disabled 1: Capture enabled
3:2	RSV	-	-	Reserved
1	CC1P	R/W	0x0	Capture/compare 1 output polarity: CC1 channel configured as output: 0: OC1 is OC1REF. 1: OC1 is the inverse OC1REF. CC1 channel configured as input: 0: Non-inverted: Capture is done on the rising edge of IC1. 1: Inverted: Capture is done on the falling edge of IC1.
0	CC1E	R/W	0x0	Capture/compare 1 output enable: CC1 channel configured as output: 0: OC1 not active 1: OC1 active CC1 channel configured as input: 0: Capture disabled 1: Capture enabled

19.6.10 Counter Register (TIM_CNT)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CNT	R/W	0x0	Counter value

19.6.11 Prescaler Register (TIM_PSC)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	PSC	R/W	0x0	Counter clock (CK_CNT) prescaler value: $f_{CK_CNT} = f_{CK_PSC} / (PSC[15:0] + 1)$ This is a preload register whose content are transferred into the shadow register at each update event.

19.6.12 Auto-reload Register (TIM_ARR)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ARR	R/W	0x0	Auto-reload value at counter overflow: This is a preload register whose content are transferred into the shadow register at each update event.

19.6.13 Capture/Compare Register 1 (TIM_CCR1)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR1	R/W	0x0	Capture/compare channel 1 register: If channel CC1 is configured as output: This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC1 output. If channel CC1 is configured as input: CCR1 stores the counter value at the time of the most recent input capture event. In this case, CCR1 is read-only.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.6.14 Capture/Compare Register 2 (TIM_CCR2)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR2	R/W	0x0	Capture/compare channel 2 register: **If channel CC2 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC2 output. **If channel CC2 is configured as input:** CCR2 stores the counter value at the time of the most recent input capture event. In this case, CCR2 is read-only.

19.6.15 Capture/Compare Register 3 (TIM_CCR3)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR3	R/W	0x0	Capture/compare channel 3 register: **If channel CC3 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC3 output. **If channel CC3 is configured as input:** CCR3 stores the counter value at the time of the most recent input capture event. In this case, CCR3 is read-only.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

19.6.16 Capture/Compare Register 4 (TIM_CCR4)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CCR4	R/W	0x0	Capture/compare channel 4 register: **If channel CC4 is configured as output:** This is a preload register, the contents of which are loaded into the shadow register and used for comparison with the counter to generate the OC4 output. **If channel CC4 is configured as input:** CCR4 stores the counter value at the time of the most recent input capture event. In this case, CCR4 is read-only.

19.6.17 DMA Control Register (TIM_DCR)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12:8	DBL	R/W	0x0	DMA burst length: A read or write access to the TIM_DMAR register will trigger DMA transfer with a burst length of 1−18. 00000: Burst length = 1 00001: Burst length = 2 00010: Burst length = 3 …… 10001: Burst length = 18 Others: Invalid value, write prohibited
7:5	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4:0	DBA	R/W	0x0	DMA base address, defined as the offset address directed to the register: 00000: TIM_CR1 00001: TIM_CR2 00010: TIM_SMCR …… Note: When DBA + DBL exceeds the TIM register address range, the actual burst transfer stops automatically when it reaches the highest TIM register address, i.e., the burst length is shortened.

19.6.18 DMA Access Register (TIM_DMAR)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	DMAR	R/W	0x0	DMA register for burst access: When using DMA burst transfer, set the DMA channel peripheral address to TIM_DMAR. Accesses to this register will point to the register specified in TIM_DCR, and TIM will generate multiple DMA requests based on the DBL value.

19.7 Operation Procedure

19.7.1 Counting Mode

1. Enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_CR1[2] to 1 so that an update interrupt or DMA request will be generated only at counter overflow or underflow.
7. Set TIM_CR1[1] to 0 to enable the update event.
8. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
9. Set TIM_CR1[0] to 1 to enable the counter.
10. Set TIM_DIER[0] to 1 to enable the update event interrupt.

19.7.2 PWM Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. According to the output channel, set TIM_CCMRx[1:0/9:8] to 0 to configure channel x as output.
7. Configure TIM_CCMRx[6:4/14:12] to set PWM mode 1/2.
8. Configure TIM_CCER[1/5/9/13] to set the output polarity.
9. Set TIM_CCER[0/4/8/12] to 1 to enable channel x output.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10. Set TIM_CR1[2] to 1 so that an update interrupt or DMA request will be generated only at counter overflow or underflow.
11. Set TIM_CR1[1] to 0 to enable the update event.
12. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
13. Set TIM_CR1[0] to 1 to enable the counter.
14. Set TIM_DIER[0] to 1 to enable the update event interrupt.
15. Configure TIM_CCRx[31:0] to set the compare value of channel x.

19.7.3 Input Capture Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR1[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Configure TIM_CCMRx[1:0/9:8] to set channel CCx as input, and perform mapping as required.
7. Configure TIM_CCER[1/5/9/13] to set the capture polarity.
8. Configure TIM_CCMRx[7:4/15:12] to set the sampling frequency and filter length, generally set to 0.
9. Configure TIM_CCMRx[3:2/11:10] to set the prescaler value of input capture.
10. Set TIM_CCER[0/4/8/12] to 1 to enable the capture function.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

11. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
12. Set TIM_CR1[0] to 1 to enable the counter.
13. Set TIM_DIER[1/2/3/4] to 1 to enable the capture interrupt of channel x.

19.7.4 Encoder Interface Mode

1. Configure the GPIO alternate function to enable the TIMx clock in RCM module.
2. Configure TIM_CR[4] to set the counting direction.
3. Set TIM_CR1[7] to 1 to enable the auto-reload preload.
4. Configure TIM_PSC[15:0] to set the prescaler value.
5. Configure TIM_ARR[31:0] to set the auto-reload value.
6. Set TIM_CCMR1[1:0] to 1 to configure channel CC1 as input with IC1 mapped on TI1.
7. Set TIM_CCMR1[9:8] to 1 to configure channel CC2 as input with IC2 mapped on TI2.
8. Configure TIM_CCER[1] to set the capture polarity.
9. Configure TIM_CCER[5] to set the capture polarity.
10. Configure TIM_CCMR1[7:4] to set the sampling frequency and filter length, generally set to 0.
11. Configure TIM_CCMR1[15:12] to set the sampling frequency and filter length, generally set to 0.
12. Configure TIM_SMCR[2:0] to set encoder mode 1/2/3.
13. Set ATIM_CCER[0] to 1 to enable the capture function of channel 1.
14. Set ATIM_CCER[4] to 1 to enable the capture function of channel 2.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

15. Set TIM_EGR[0] to 1 so that when the software sets TIM_EGR[0], the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.
16. Set TIM_CR1[0] to 1 to enable the counter.
17. Set TIM_DIER[1] to 1 to enable capture interrupt of channel 1.

19.7.5 DMA Mode

In input capture mode, the channel capture value of TIMx is transferred to SRAM via DMA:

1. In input capture mode, before setting TIM_EGR[0] bit by software and enabling the counter, add the following configurations:
2. Configure TIM_DCR[12:8] to set the DMA burst length.
3. Configure TIM_DCR[4:0] to set the DMA base address. Generally, the base address here selects the capture/compare register corresponding to the capture channel.
4. Set TIM_DIER[9/10/11/12] to 1 to enable the CCx DMA request.
5. Set TIM_CR2[3] to 0 to enable the CCxDMA request generation at CCx event.
6. For details on DMA controller configuration, please refer to Chapter 11.
7. After initiating DMA transfer, when a capture event occurs on the channel, DMA will transfer the value stored in base address to SRAM.

In output compare mode, the value in SRAM is transferred via DMA to the compare register of TIMx.

1. In PWM mode, before setting TIM_EGR[0] bit by software and enabling the counter, add the following configurations:
2. Configure TIM_DCR[12:8] to set the DMA burst length.
3. Configure TIM_DCR[4:0] to set the DMA base address. Generally, the base address here selects the capture/compare register corresponding to the compare channel.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Set TIM_DIER[9/10/11/12] to 1 to enable the CCx DMA request.
5. Set TIM_CR2[3] to 0 to enable the CCxDMA request generation at CCx event.
6. For details on DMA controller configuration, please refer to Chapter 11.
7. After the DMA transfer is started, when the counter value matches the compare value, DMA will transfer the value in SRAM to the base address.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20 Basic Timer (TIM6 & TIM7)

20.1 Overview

The basic timer consists of a 32-bit auto-reload counter driven by a programmable prescaler.

20.2 Main Features

- 32-bit auto-reload upcounter
- 16-bit programmable prescaler allowing real-time adjustment of the counter clock division
- Interrupt generation on the following event:
  - Counter overflow, counter initialization (triggered by software or hardware)

20.3 Functional Description

20.3.1 Time-base Unit

The main block of the time-base unit is a 32-bit counter with its related auto-reload register. The counter clock can be divided by a 16-bit prescaler.

The counter, the auto-reload register and the prescaler register can be written or read by software, which is true even when the counter is running.

The time-base unit includes:

- Counter register (TIM_CNT)
- Prescaler register (TIM_PSC)
- Auto-reload register (TIM_ARR)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The auto-reload register is preloaded, which is controlled by the auto-reload preload enable (ARPE) bit in the register. When ARPE = 0, write to the ARR register, and the written data is directly transferred to the shadow register. When ARPE = 1, the data written to the ARR register is transferred to the shadow register when an update event (TIM_CNT overflow or underflow) occurs. The update event of ARR can also be actively triggered by software via register operation.

The counter TIM_CNT is clocked by the prescaler output TIM_PSC, which is enabled only when the counter enable bit (CEN) in the register is set. When CNT = ARR, this round of counting is over and the update event is sent.

TIM_PSC is a synchronous prescaler that can divide the counter clock frequency by any factor between 1 and 65536. The PSC register is also buffered, and overwriting PSC does not actually overwrite the shadow register unless a new update event occurs. Thus, the PSC register can be changed in real time on the fly, and the new prescaler ratio is taken into account at the next update event.

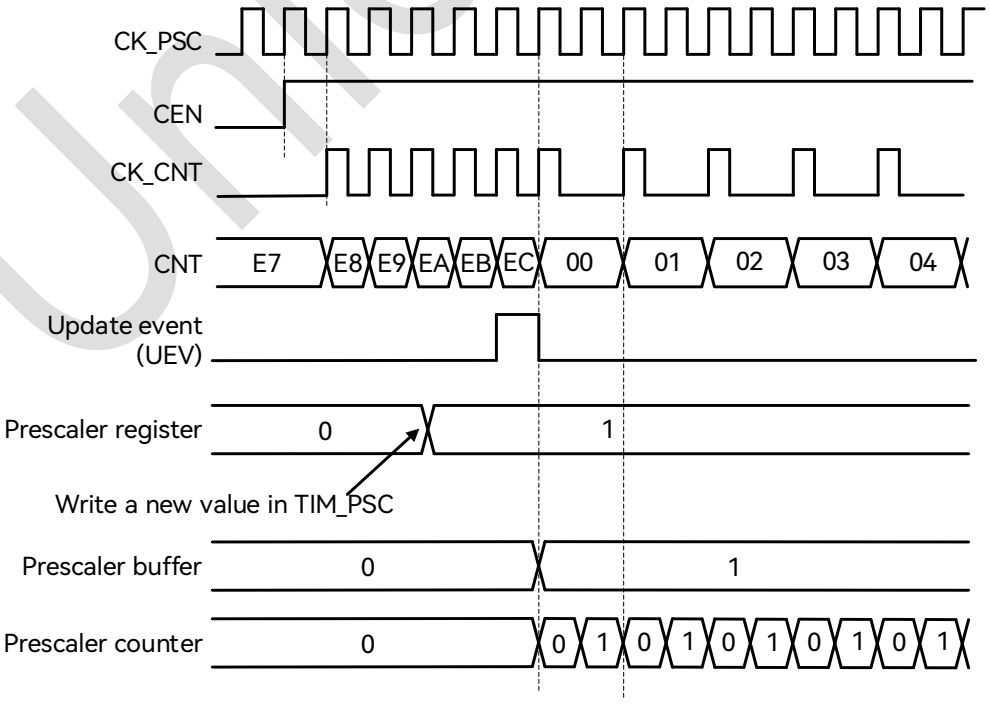

Figure 20-1: Counter Timing Diagram with Prescaler Division Changing from 1 to 2

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

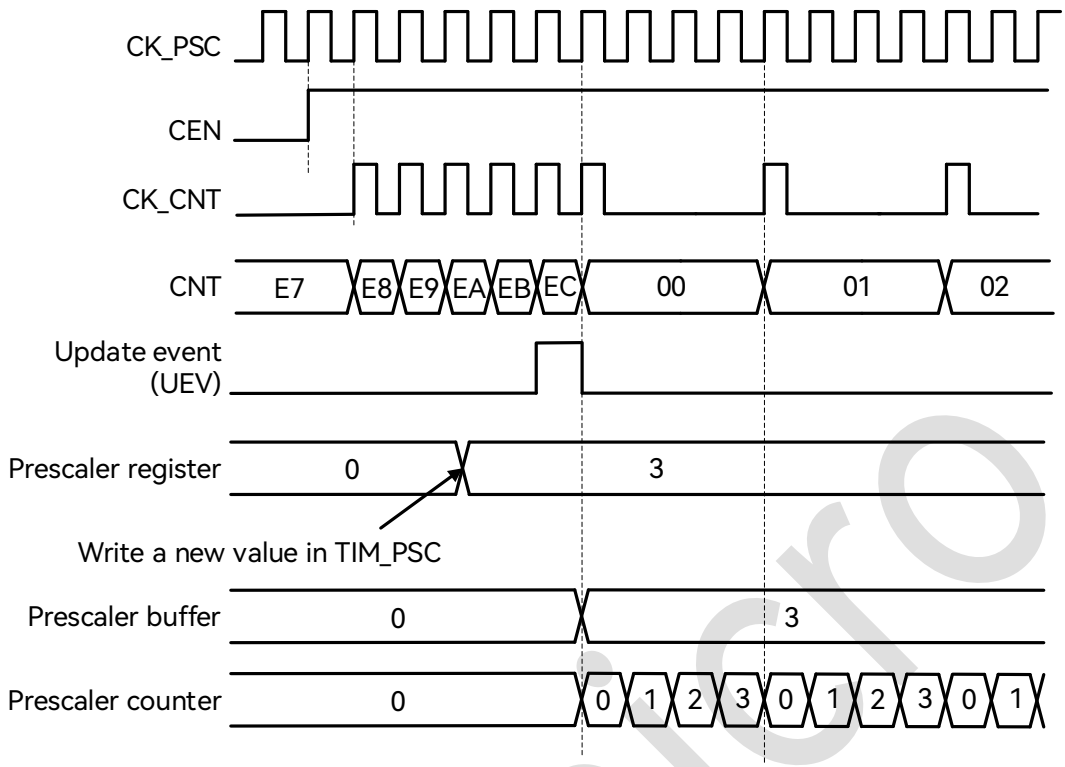

Figure 20-2: Counter Timing Diagram with Prescaler Division Changing from 1 to 4

20.3.2 Counter Operation Mode

The basic timer only supports up-counting mode.

In up-counting mode, the counter counts from 0 to the auto-reload value, i.e. CNT = ARR, generating an overflow event, and then restarts counting from 0.

If the repetition counter is enabled, the counter repeats the above process a number of times (RCR + 1) as defined in RCR before generating an underflow event.

The software can directly trigger an update event by setting the UG bit in the register, at which time the CNT and the prescaler registers are automatically cleared. Whether setting the UG register triggers UIF (update interrupt flag) is determined by the setting of the URS register.

The update event can be disabled by setting the UDIS bit in the register to avoid updating the shadow register while writing new values in the preload registers.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When an update event occurs, the following registers are updated and the UIF bit is set:

- The auto-reload shadow register ARR is reloaded with the content of TIM_ARR register.
- The prescaler shadow register PSC is reloaded with the content of TIM_PSC register.

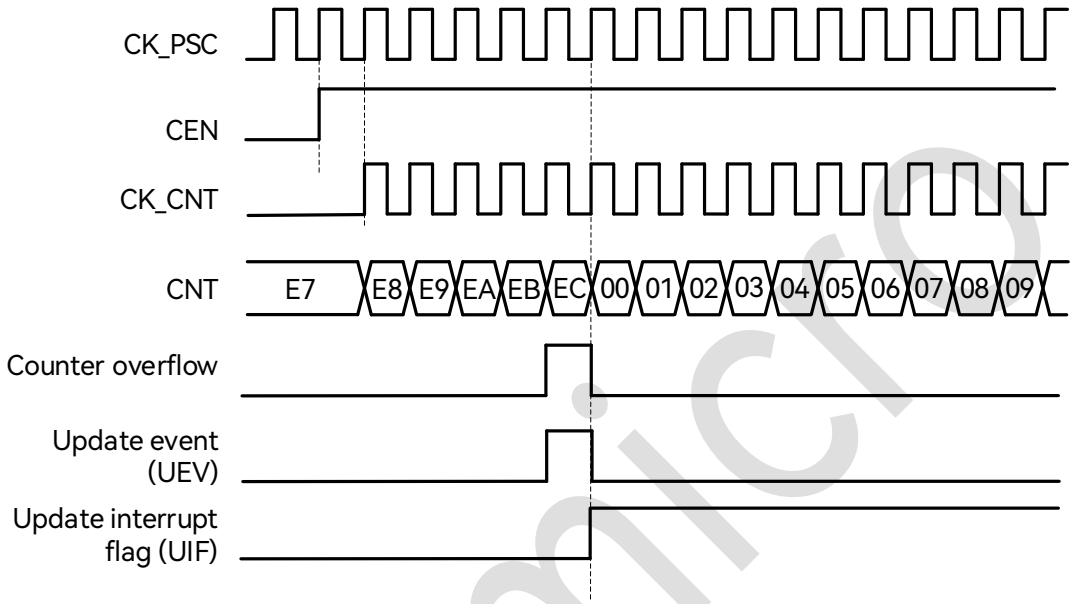

Figure 20-3: Up-counting Waveform Diagram, Internal Clock not Divided

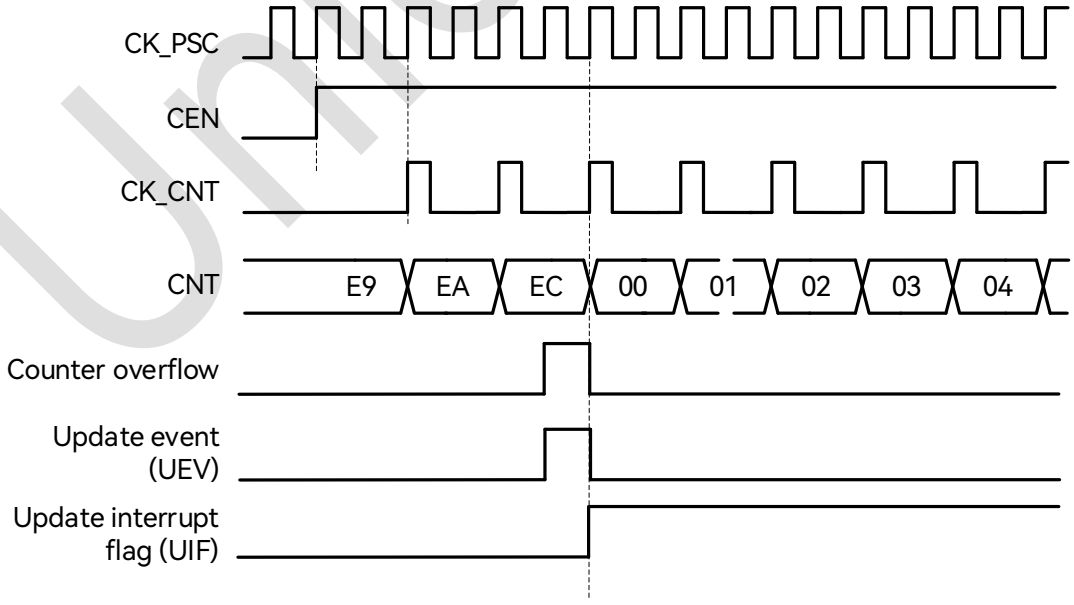

Figure 20-4: Up-counting Waveform Diagram, Internal Clock Divided by 2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

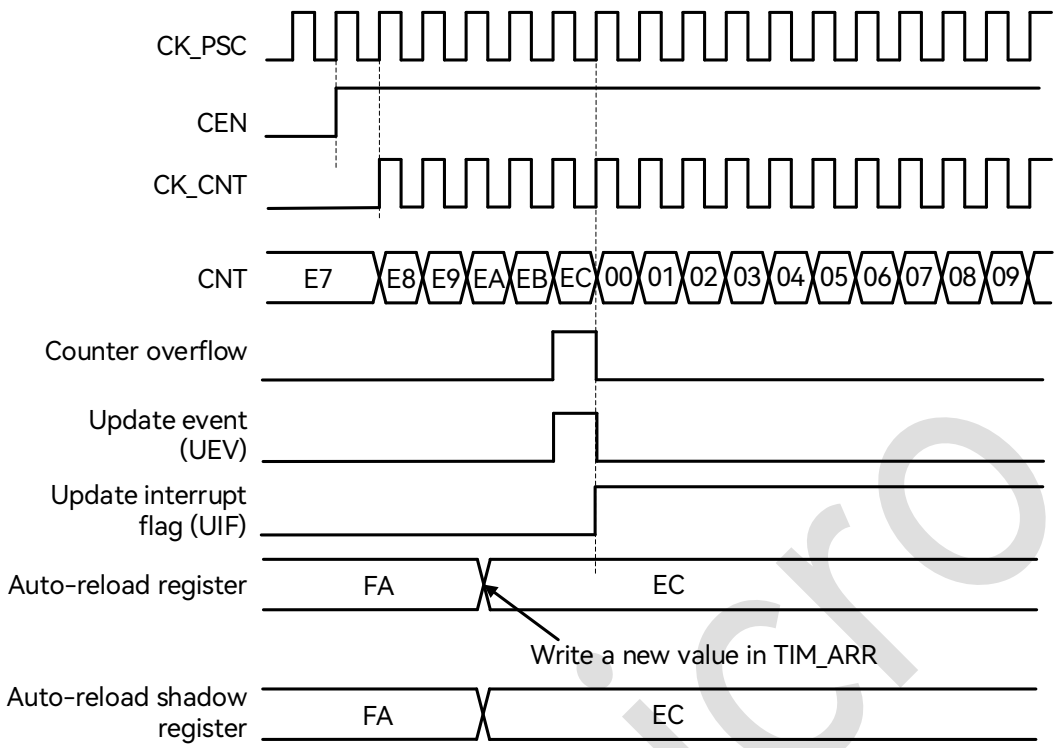

Figure 20-5: Counter Timing Diagram, Update Event when ARPE = 0 (TIM_ARR not Preloaded)

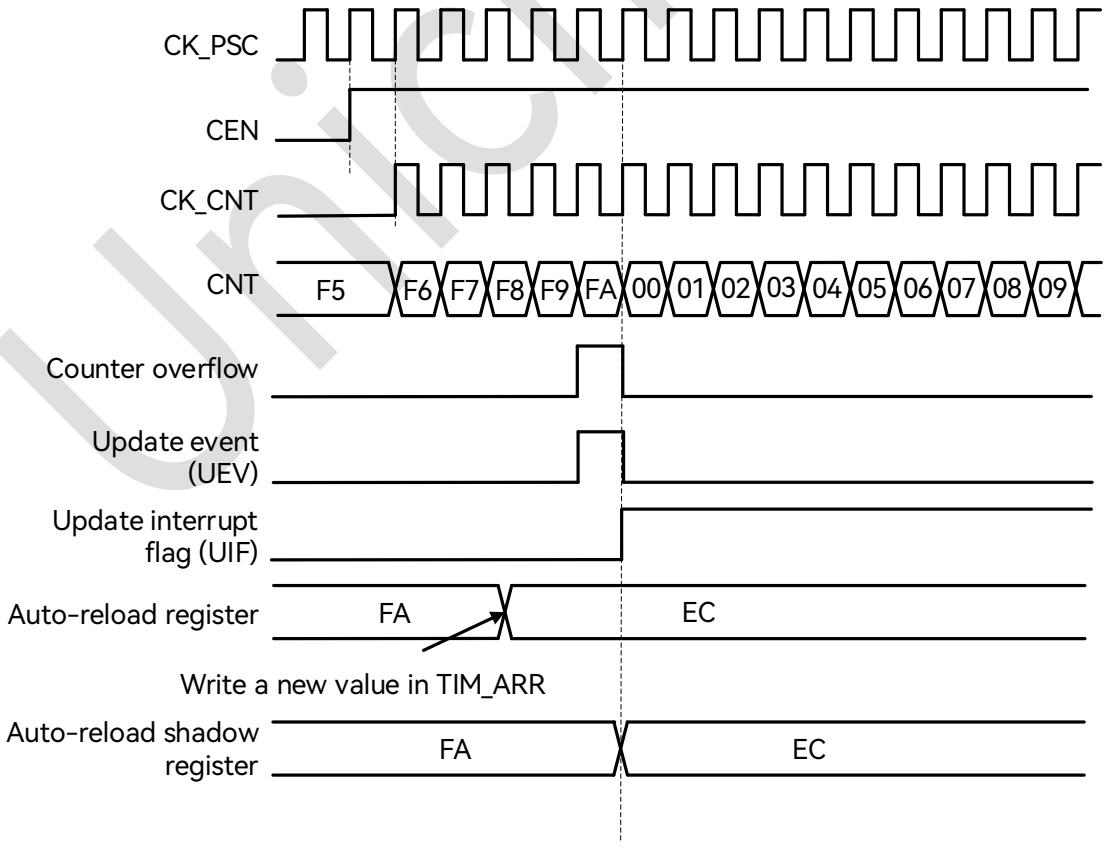

Figure 20-6: Counter Timing Diagram, Update Event when ARPE = 1 (TIM_ARR Preloaded)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20.3.3 Preload Register

- The following functional registers support the preload function:
  - Auto-reload register TIM_ARR
  - Prescaler register TIM_PSC (preload function cannot be disabled)
  - Channel control register TIM_CCR
  - CCxE and CCxNE control register
  - OCxM control register

The preload function can be enabled or disabled by software for all of the above registers except TIM_PSC.

- Registers with preload function contain two sets of physical entities:
  - Shadow register: the register being used by the actual timer
  - Preload register: the register accessible to software
- When the preload function is disabled, the register with preload function has the following characteristics:
  - The preload register can be accessed and overwritten by software in real time.
  - The shadow register is updated synchronously with the preload register.
- If the preload function is enabled, then:
  - All software operations access the preload register.
  - At the occurrence of update event, the content of all preload registers will be synchronously transferred to the corresponding shadow registers.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20.3.4 Counter Clock

The counter clock can be provided by the following clock sources:

Internal clock: Timerx_clk

20.3.4.1 Internal Clock Source

If the slave mode controller is disabled (SMS = 000), then the CEN, DIR and UG bits are controlled by software.

After the UG bit is set and the update signal is synchronized by CLK_PSC, the counter value is reinitialized.

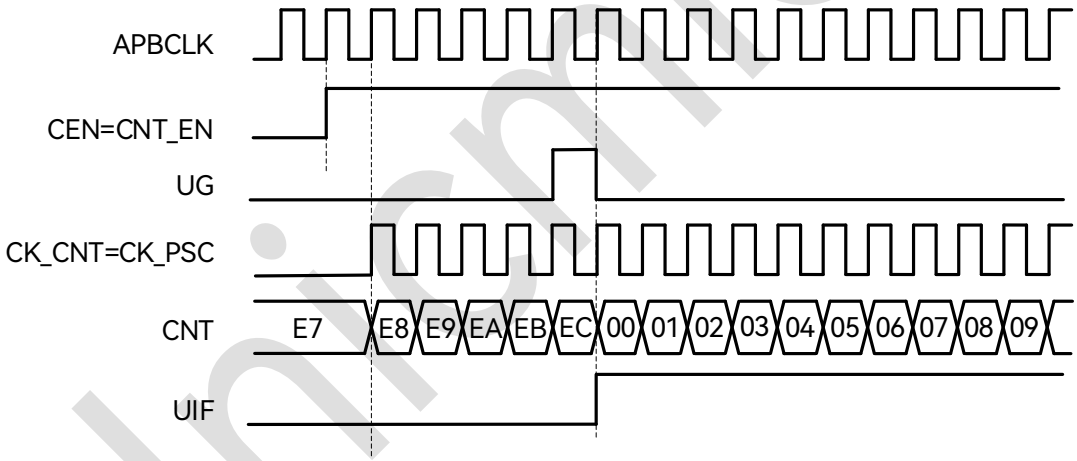

Figure 20-7: Timing Diagram in Internal Clock Source Mode, Clock Divided by 1

20.3.5 Debug Mode

When the CPU enters debug mode, the timer can either stop or continue working, and its behavior is defined by registers in the chip system.

When the timer is stopped during debugging, its output will be disabled (MOE is cleared). Depending on the register configuration, the output signal can be forced to be inactive or controlled by the GPIO module.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20.4 Register Description

TIM6 register base address: 0x4600_B000

TIM7 register base address: 0x4600_B400

The registers are listed below:

Table 20-1: List of TIM6 & TIM7 Registers

Offset Address	Name	Description
0x00	TIM_CR1	Control register 1
0x04	TIM_CR2	Control register 2
0x0C	TIM_DIER	DMA and interrupt enable register
0x10	TIM_SR	Status register
0x14	TIM_EGR	Event generation register
0x24	TIM_CNT	Counter register
0x28	TIM_PSC	Prescaler register
0x2C	TIM_ARR	Auto-reload register

Registers are detailed in the following sections.

20.4.1 Control Register 1 (TIM_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	APRE	R/W	0x0	Auto-reload preload enable: 0: ARR not preloaded 1: ARR preloaded
6:4	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	OPM	R/W	0x0	One-pulse mode output: 0: The counter does not stop at the occurrence of update event. 1: The counter stops at the occurrence of update event (CEN cleared automatically).
2	URS	R/W	0x0	Update request source: 0: An update interrupt or DMA request will be generated by any of the following events: ● Counter overflow/underflow ● Software setting the UG bit ● Update generated from the slave mode controller 1: An update interrupt or DMA request will be generated only at counter overflow or underflow.
1	UDIS	R/W	0x0	Update disable: 0: Update event enabled; the update event can be generated by any of the following events: ● Counter overflow/underflow ● Software setting the UG bit 1: Update event disabled, shadow register not updated; the counter and the prescaler will be reinitialized if the UG bit is set or if the slave mode controller receives a hardware reset.
0	CEN	R/W	0x0	Counter enable: 0: Disabled 1: Enabled Note: The external trigger mode can automatically set the CEN bit.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

20.4.2 Control Register 2 (TIM_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved
6:4	MMS	R/W	0x0	Master mode selection, selecting the TRGO trigger mode: 000: reset—TRGO is generated by the UG bit in the EGR register. 001: enable—TRGO is generated by the counter enable signal, including the CEN control bit and external trigger. 010: Update—TRGO is generated by the update event. 011: Compare pulse—TRGO is generated when an input capture or compare event occurs and sets CC1F to 1. 100: Compare—TRGO is generated by OC1REF. 101: Compare—TRGO is generated by OC2REF. 110: Compare—TRGO is generated by OC3REF. 111: Compare—TRGO is generated by OC4REF.
3:0	RSV	-	-	Reserved

20.4.3 DMA / Interrupt Enable Register (TIM_DIER)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	UDE	R/W	0x0	Update DMA request enable: 0: Update DMA request disabled 1: Update DMA request enabled
7:1	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	UIE	R/W	0x0	Update interrupt enable: 0: Update interrupt disabled 1: Update interrupt enabled

20.4.4 Status Register (TIM_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	UIF	R/W0C	0x0	Update interrupt flag is set by hardware and cleared by software via writing it to 0. UIF is set and the shadow register is updated at the following events: ● Counter overflow occurs if repetition counter = 0 and UDIS = 0. ● The counter is reinitialized by software setting the UG bit if URS = 0 and UDIS = 0. ● The counter is reinitialized by a trigger event if URS = 0 and UDIS = 0.

20.4.5 Event Generation Register (TIM_EGR)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	UG	W	0x0	This bit can be set by software to generate an update event, and is automatically cleared by hardware. When the software sets UG, the counter will be reinitialized, the shadow register will be updated, and the prescaler counter will be cleared.

20.4.6 Counter Register (TIM_CNT)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CNT	R/W	0x0	Counter value

20.4.7 Prescaler Register (TIM_PSC)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	PSC	R/W	0x0	Counter clock (CK_CNT) prescaler value: $f_{CK_CNT} = f_{CK_PSC} / (PSC[15:0] + 1)$ This is a preload register whose content are transferred into the shadow register at each update event.

20.4.8 Auto-reload Register (TIM_ARR)

Offset address: 0x2C

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	ARR	R/W	0x0	Auto-reload value at counter overflow: This is a preload register whose content are transferred into the shadow register at each update event.

20.5 Operation Procedure

20.5.1 Counter Startup

1. Enable the TIMx clock in RCM module.
2. Configure TIM_ARR[31:0] to set the auto-reload value.
3. Configure TIM_PSC[15:0] to set the prescaler value.
4. Set TIM_EGR[0] to 1 to generate a software update event and update the shadow register.
5. Set TIM_SR[0] to 0 to clear the update interrupt flag.
6. Set TIM_CR1[0] to 1 to enable the counter.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21 Low-power Timer (LPTIM)

21.1 Overview

LPTIM is a 16-bit low-power timer/counter module running in always-on power domain. By selecting suitable clock source, LPTIM is able to keep running in various low-power modes with extremely low power consumption. LPTIM can be used as an external pulse counter in low-power mode even with no internal clock source. Also, in combination with an external input trigger signal, LPTIM is able to realize timeout wake-up from low-power modes.

This chip is provided with two low-power timers: LPTIM1 and LPTIM2. LPTIM1/2 are capable of waking up the system from sleep and stop modes through interrupts. After entering Standby0 mode, LPTIM1 can wake the system up through counter overflow interrupt, compare match interrupt, and external pulse counting interrupt (all interrupts generated by LPTIM1 can wake it up from Standby0). In contrast, LPTIM2 only supports wakeup from Standby0 through internal counter overflow interrupt and compare match interrupt; it does not support wakeup by external pulse counting interrupt.

21.2 Main Features

- 16-bit upcounter
- 3-bit asynchronous prescaler with 8 possible dividing factors (1, 2, 4, 8, 16, 32, 64, 128)
- Selectable clock source:
  - Internal clock sources: LSCLK (RCL or XTL), CLK1Hz, APB0 (PCLK0)
  - External clock source: LPTIMx_IN

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- 16-bit compare register
- 16-bit destination register
- Selectable software / hardware input trigger
- Configurable input polarity
- External pulse counting with no clock source
- Externally triggered timeout wakeup from low-power modes
- Timed wakeup
- 16-bit PWM

21.3 System Block Diagram

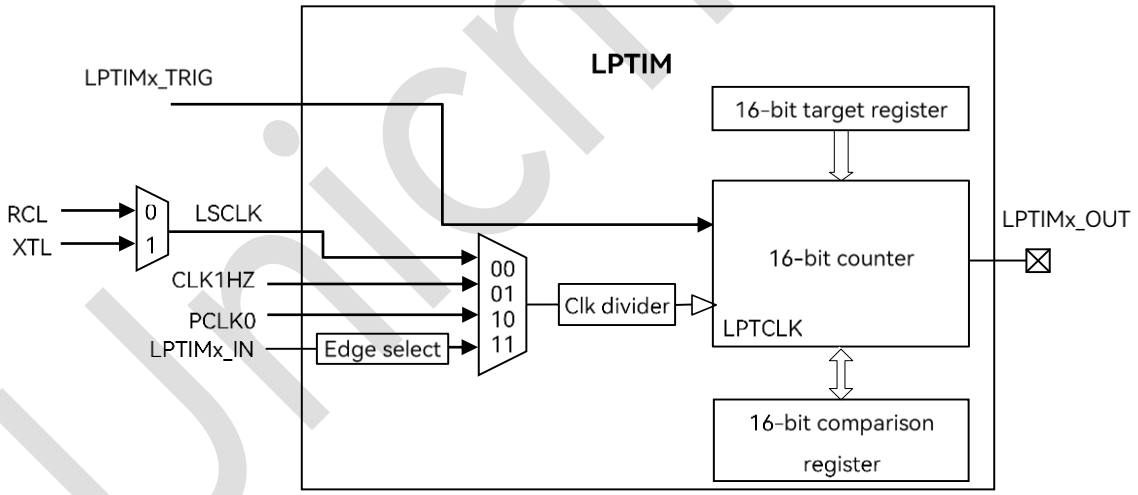

Figure 21-1: System Block Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.4 Pin Description

Table 21-1: LPTIM Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
LPTIM1_IN	PC0	Input	LPTimer signal input (for external asynchronous pulse counting mode)
LPTIM1_TRIG	PC3	Input	LPTimer clock input (input capture)
LPTIM1_OUT	PC1	Output	LPTimer signal output (PWM)
LPTIM2_IN	PD10	Input	LPTIM signal input (for external asynchronous pulse counting mode)
LPTIM2_TRIG	PD11	Input	LPTimer clock input (input capture)
LPTIM2_OUT	PE4	Output	LPTimer signal output (PWM)

Note: In Standby0 Mode, LPTIM1 can wake up the system through pins PC0 and PC3. LPTIM2 can only wake up the system through internal timing and cannot be awakened by the external pins PD10 or PD11.

21.5 Functional Description

21.5.1 General-purpose Timer

LPTIM can operate using either internal or external clocks, and after enabling there is a synchronization process of two counting clock cycles before operation begins.

21.5.2 Pulse-trigger Counting

LPTIM works with an internal clock, sampling the external asynchronous trigger signal, and can count the rising edge, falling edge, or both edges of the trigger signal. When the number of pulses reaches the set comparison value or overflows, the corresponding interrupt flag bit will be set to 1. There is a synchronization process of two counting clock cycles before and after enabling.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.5.3 External Asynchronous Pulse Counting

LPTIM directly uses the external input pulse as the working clock, with polarity configurable to rising edge or falling edge. Once enabled, it starts working immediately without a synchronization process.

21.5.4 Timeout Mode

LPTIM operates with an internal or external clock, sampling the external asynchronous trigger signal. The counter will be started at the first sampling of the trigger signal, and will be cleared and restarted if another trigger signal is sampled after startup. If there is no new trigger before the counter overflows, an overflow interrupt will be generated, the counting will be stopped, and the enable will be cleared. There is a synchronization process of two counting clock cycles after enabling.

21.5.5 Counting Mode

The LPTIM features two counting modes:

Continuous counting mode: The counter keeps running after being enabled until it is disabled. Upon reaching the target value, it returns to 0 to restart counting, generating an overflow interrupt.

One-shot counting mode: The counter counts to the target value upon being triggered, then resets to 0 and automatically stops, generating an overflow interrupt. As the overflow signal and the LPTEN enable signal are in different clock domains, disabling the enable signal is implemented by asynchronous reset and synchronous release.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.5.6 Externally Triggered Timeout Wakeup

LPTIM can be enabled by an external trigger signal or by software. In timeout mode, the first valid edge of the external trigger input will start the counter, while the subsequent trigger signal will clear the counter. If there is no effective trigger signal before the counter reaches the comparison value, a timeout interrupt is generated to wake up MCU.

The valid edge of the external trigger signal can be configured via registers, and since the external trigger signal is considered asynchronous input, there is a delay of at least two counting clock cycles for sampling and judgment.

21.5.7 16-bit PWM

After enabling PWM mode, LPTIM starts counting from 0x0000. The output goes high when the count value is equal to the comparison value, and goes low when the count value is equal to the final value. The PWM period is determined by the target value register, and the duty cycle is determined by the compare value register.

21.6 Register Description

LPTIM1 register base address: 0x40B0_9000

LPTIM2 register base address: 0x40B0_A000

The registers are listed below:

Table 21-2: List of LPTIM Registers

Offset Address	Name	Description
0x00	LPTIM_CFG	Configuration register
0x04	LPTIM_CNT	Counter register
0x08	LPTIM_CMP	Compare value register
0x0C	LPTIM_TARGET	Target value register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x10	LPTIM_IE	Interrupt enable register
0x14	LPTIM_IF	Interrupt status register
0x18	LPTIM_CTRL	Control register

Registers are detailed in the following sections.

21.6.1 Configuration Register (LPTIM_CFG)

Offset address: 0x00

Reset value: 0x0000 0200

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	FLTEN	R/W	0x0	External trigger signal filter enable: When enabled, external trigger signals with a holding time shorter than two counting clock cycles will be filtered out. 0: Filter disabled 1: Filter enabled
14:13	RSV	-	-	Reserved
12:10	DIVSEL	R/W	0x0	Counter clock division selection: 000: Divided by 1 001: Divided by 2 010: Divided by 4 011: Divided by 8 100: Divided by 16 101: Divided by 32 110: Divided by 64 111: Divided by 128

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
9:8	CLKSEL	R/W	0x2	Counter clock source selection: 00: LSCLK (low-speed system clock RCL or XTL) selected as counter clock 01: RCLP selected as counter clock 10: PCLK of LPTIM selected as counter clock 11: LPTIM (external pin selected by SYSCFG->SYSCTRL1[11] or CLK_1Hz) selected as counter clock
7	EDGESEL	R/W	0x0	LPTIM input edge selection: 0: LPTIM counts at rising edge 1: LPTIM counts at falling edge
6:5	TRIGCFG	R/W	0x0	External trigger edge selection: 00: Rising edge for external trigger input 01: Falling edge for external trigger input 10/11: Rising/falling edge for external trigger input
4	POLARITY	R/W	0x0	Counter clock division selection: 0: Positive waveform, that is, when the first count value = the comparison value, the rising edge of the output waveform is generated. 1: Negative waveform, that is, when the first count value = the comparison value, the falling edge of the output waveform is generated.
3	PWM	R/W	0x0	Pulse width modulation mode: 0: Periodic square wave output 1: PWM output

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	MODE	R/W	0x0	Counting mode: 0: Continuous counting mode: the counter keeps running after being triggered until it is disabled. After the counter reaches the target value, it returns to 0 to restart counting, and an overflow interrupt is generated. 1: One-shot counting mode: After being triggered, the counter counts to the target value, then returns to 0 and automatically stops, generating an overflow interrupt.
1:0	TMODE	R/W	0x0	Operation mode selection: 00: General timer mode with waveform output 01: Pulse-trigger counting mode 10: External asynchronous pulse counting mode 11: Timeout mode

21.6.2 Counter Register (LPTIM_CNT)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	CNT	R	0x0	Counter value

21.6.3 Compare Value Register (LPTIM_CMP)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	CMP	R/W	0x0	Compare value register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.6.4 Target Value Register (LPTIM_TARGET)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	TARGET	R/W	0x0	Target value

21.6.5 Interrupt Enable Register (LPTIM_IE)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2	TRIGIE	R/W	0x0	External trigger valid edge interrupt enable: 1: Enabled 0: Disabled
1	OVIE	R/W	0x0	Counter overflow interrupt enable: 1: Enabled 0: Disabled
0	COMPIE	R/W	0x0	Compare match interrupt enable: 1: Interrupt enabled when the counter value matches the compare value 0: Interrupt disabled when the counter value matches the compare value

21.6.6 Interrupt Flag Register (LPTIM_IF)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	TRIGIF	R/W1C	0x0	**External trigger interrupt enable:** 1: External trigger interrupt enabled 0: External trigger interrupt disabled
1	OVIF	R/W1C	0x0	**Counter overflow interrupt enable:** 1: Counter overflow interrupt enabled 0: Counter overflow interrupt disabled
0	COMPIF	R/W1C	0x0	**Compare match interrupt enable:** 1: Interrupt enabled when the counter value matches the compare value 0: Interrupt disabled when the counter value matches the compare value

21.6.7 Control Register (LPTIM_CTRL)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	LPTEN	R/W	0x0	LPTIM enable: 0: Counter counting disabled 1: Counter counting enabled

21.7 Operation Procedure

21.7.1 Enabling LPTIM Clock

1. Write 0xABCD to the PMU_CPR register to enable its write operation.
2. Configure PMU_FCCR[3:2] to enable the LPTIM1/2 controller clock.
3. Configure PMU_FRCR[3:2] and release the LPTIM1/2 controller reset.
4. Write 0x459E to the PMU_CPR register to end its write operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

21.7.2 LPTIM Counting

1. Enable the LPTIM clock.
2. Configure the LPTIM_CFG register, and write 0 to clear previous configurations.
3. Configure LPTIM_CFG[12:10] to set the clock division factor, configure LPTIM_CFG[9:8] to set the LPTIM clock source, configure LPTIM_CFG[2] to set the LPTIM counting mode, and configure LPTIM_CFG[1:0] to set the LPTIM operating mode (general timer mode with waveform output).
4. Configure LPTIM_TARGET [15:0] to set the target count value.
5. Configure LPTIM_IE[1] to enable the LPTIM overflow interrupt.
6. Configure LPTIM_CTRL[0] to enable LPTIM.

21.7.3 LPTIM PWM Output

1. Enable the LPTIM clock.
2. Configure the pin where LPTIM_OUT is located to enable its clock and multiplex its function to LPTIM_OUT.
3. Configure the LPTIM_CFG register, and write 0 to clear previous configurations.
4. Configure LPTIM_CFG[12:10] to set the clock division factor, configure LPTIM_CFG[9:8] to set the LPTIM clock source, configure LPTIM_CFG[2] to set the LPTIM counting mode, configure LPTIM_CFG[1:0] to set the LPTIM operating mode (general timer mode with waveform output), and set the pulse width modulation mode of PWM bit to PWM output.
5. Configure LPTIM_CMP [15:0] to set the count compare value.
6. Configure LPTIM_TARGET [15:0] to set the target count value.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Configure LPTIM_IE[0] to enable the LPTIM compare match interrupt.

8. Configure LPTIM_CTRL[0] to enable LPTIM.

21.7.4 LPTIM External Trigger Count Interrupt

1. Enable the LPTIM clock.

2. Configure the pin where LPTIM_TRIGGRT is located to enable its clock and multiplex its function to LPTIM_TRIGGER.

3. Configure the LPTIM_CFG register, and write 0 to clear previous configurations.

4. Configure LPTIM_CFG[12:10] to set the clock division factor, configure LPTIM_CFG[9:8] to set the LPTIM clock source, configure LPTIM_CFG[2] to set the LPTIM counting mode, configure LPTIM_CFG[1:0] to set the LPTIM operating mode (pulse-trigger counting mode), configure the TRIGCFG bit to set the valid edge for external trigger signal, and set FLTEN bit 1 to enable trigger signal filtering.

5. Configure LPTIM_TARGET [15:0] to set the target count value.

6. Configure LPTIM_IE[2] to enable the LPTIM trigger interrupt.

7. Configure LPTIM_CTRL[0] to enable LPTIM.

21.7.5 LPTIM Timeout Mode

1. Enable the LPTIM clock.

2. Configure the pin where LPTIM_TRIGGRT is located to enable its clock and multiplex its function to LPTIM_TRIGGER.

3. Configure the LPTIM_CFG register, and write 0 to clear previous configurations.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Configure LPTIM_CFG[12:10] to set the clock division factor, configure LPTIM_CFG[9:8] to set the LPTIM clock source, configure LPTIM_CFG[2] to set the LPTIM counting mode, configure LPTIM_CFG[1:0] to set the LPTIM operating mode (timeout mode), configure the TRIGCFG bit to set the valid edge for external trigger signal, and set FLTEN bit 1 to enable trigger signal filtering.
5. Configure the LPTM_TARGET register to set the target count value.
6. Configure the OVIE bit of LPTIM_IE register to enable the LPTIM overflow interrupt.
7. Configure the LPTEN bit of LPTIM_CTRL register to enable LPTIM.

21.7.6 LPTIM External Clock Source Count Overflow Interrupt

1. Enable the LPTIM clock.
2. Configure the pin where LPTIM_IN is located to enable its clock and multiplex its function to LPTIM_IN.
3. Configure the LPTIM_CFG register, and write 0 to clear previous configurations.
4. Configure the CLKSEL, DIVSEL and TMODE bits of the LPTIM_CFG register to correspondingly set the LPTIM clock source (with the CLKSEL bit set to 11), the clock division factor, and the LPTIM operating mode (external asynchronous pulse counting mode), and configure the EDGESEL bit to set the counting edge of the external clock source.
5. Configure LPTIM_TARGET [15:0] to set the target count value.
6. Configure LPTIM_IE[1] to enable the LPTIM overflow interrupt.
7. Configure LPTIM_CTRL[0] to enable LPTIM.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

22 Independent Watchdog Timer (IWDT)

22.1 Overview

The watchdog timer can generate a non-maskable interrupt or reset when the counter reaches the given timeout value. It can be used to regain control when the system fails to respond as expected due to software errors or external device failures.

22.2 Main Features

- 32-bit downcounter with programmable load
- Independent watchdog timer enabled
- Interrupt generation logic with interrupt masking
- Lockout register for software runaway protection
- Software boot function: reset enabling / disabling in IWDT control register

22.3 Register Description

IWDT register base address: 0x40B0_6000

The registers are listed below:

Table 22-1: List of IWDT Registers

Offset Address	Name	Description
0x00	IWDT_LOAD	Load register
0x04	IWDT_CNT	Counter register
0x08	IWDT_CTRL	Control register
0x0C	IWDT_CLR	Clear register
0x10	IWDT_INTRAW	Raw interrupt status register
0x14	IWDT_MINTS	Interrupt status register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x18	IWDT_STALL	Clock division register
0x1C	IWDT_LOCK	Counter lock register

Registers are detailed in the following sections.

22.3.1 Load Register (IWDT_LOAD)

Offset address: 0x00

Reset value: 0x0000 3FFF

Bit	Name	Attribute	Reset Value	Description
31:0	LOAD	R/W	0x0000 3FFF	IWDOG initial load value

22.3.2 Counter Register (IWDT_CNT)

Offset address: 0x04

Reset value: 0x0000 3FFF

Bit	Name	Attribute	Reset Value	Description
31:0	CNT	R	0x0000 3FFF	Counter value in IWDOG If the clock source frequency is 32,768 Hz, the default overflow time is about 0.5 s.

22.3.3 Control Register (IWDT_CTRL)

Offset address: 0x08

Reset value: 0x8000 0004

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31	WRC	R	0x1	The IWDT load value is set or the IWDT_CTRL register takes effect. When writing to the IWDT_LOAD or IWDT_CTRL register, there will be a delay (3 – 4 IWDTCLK cycles without prescaling) before the set bit takes effect: 0: The setting has not taken effect yet. 1: The setting has taken effect.
30:3	RSV	-	-	Reserved
2	RST_MODE	R/W	0x1	IWDT overflow reset mode selection (without interrupt enabled): 0: Overflow reset shall be counted twice for the first time 1: Overflow reset shall be counted once
1	RSTEN	R/W	0x0	IWDT overflow reset enable: 0: Disabled 1: Enabled
0	INTEN	R/W	0x0	IWDT interrupt enable: 0: Disabled 1: Enabled Note: Overflow interrupt has higher priority than overflow reset.

Notes:

When Rst_mode is set to 0 without enabling the interrupt, the overflow reset shall be counted twice for the first time.

1. If the load register (IWDT_LOAD) is configured before the first overflow, it will reset the count twice. Therefore, as long as the load register is configured before the first overflow, it will reset the count twice. However, if the load register is configured once after the first overflow, subsequent configurations of the load register will only reset the overflow count once.
2. After the first overflow, if the load register (IWDT_LOAD) is configured, then each time the load register is configured before the next overflow, it will only reset the overflow count once.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

22.3.4 Clear Register (IWDT_CLR)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CLR_CARRY	W	0x0	Writing any value to this register will clear the IWDT overflow status, thereby clearing both the interrupt and the reset. At the same time, the value of the load register IWDT_LOAD will be refreshed to the counter register IWDT_CNT (equivalent to a watchdog feeding action).

22.3.5 Raw Interrupt Status Register (IWDT_INTRAW)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	INTRAW	R	0x0	Raw interrupt register, unmasked by interrupt enable: 0: No overflow occurred in IWDT 1: Overflow occurred in IWDT

22.3.6 Interrupt Status Register (IWDT_MINTS)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	INTMS	R	0x0	IWDT interrupt flag bit: 0: No interrupt occurred in IWDT 1: Interrupt occurred in IWDT

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

22.3.7 Clock Division Register (IWDT_STALL)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	CLK_DIV	R/W	0x0	IWDT counter clock division value: 0x0: No frequency division 0x1: Divided by 2 0x2: Divided by 3 …. 0xFFFF: 0xFFFF+ divided by 1
15:9	RSV	-	-	Reserved
8	STALL	R/W	0x0	Enable bit for IWDT not counting when the chip is in HALT state: 0: Counter stop function disabled in HALT state 1: Counter stop function enabled in HALT state
7:0	RSV	-	-	Reserved

22.3.8 Counter Lock Register (IWDT_LOCK)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	LOCK	R/W	0x0	IWDT lock function enable: It is not locked by default. When the LOCK function is enabled, all WDT registers except this one are not writable. Write any value except 0x1ACCE551 to this register can enable IWDT LOCK function, while write 0x1ACCE551 to this register can clear the LOCK function. Read this register: returning 1 indicates a locked state, while returning 0 indicates an unlocked state.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

22.4 Operation Procedure

22.4.1 IWDT Clock Enabling

1. Write 0xABCD to the PMU_CPR register to enable its write operation.
2. Configure PMU_FCCR[1] to enable the IWDT controller clock.
3. Configure PMU_FRCR[1] to release the IWDT controller reset.
4. Write 0x459E to the PMU_CPR register to end its write operation.

22.4.2 IWDT Load Value Setting

1. Write 0x1ACCE551 to the IWDT_LOCK register to unlock the register.
2. Configure IWDT_LOAD[31:0] to set the IWDT counter load value.
3. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
4. Write any value except 0x1ACCE551 to the IWDT_LOCK register to lock the register.

22.4.3 IWDT Counter Overflow Interrupt

1. Enable the IWDT clock.
2. Write 0x1ACCE551 to the IWDT_LOCK register to unlock the register.
3. Configure IWDT_STALL[31:16] to set the IWDT clock division.
4. Initialize the IWDT counter overflow interrupt by setting IWDT_CTRL[1] to 1 to enable the IWDT interrupt.
5. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
6. Configure IWDT_LOAD[31:0] to set the IWDT counter load value.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
8. Write any value except 0x1ACCE551 to the IWDT_LOCK register to lock the register.

22.4.4 IWDT Counter Overflow Reset

1. Enable the IWDT clock.
2. Write 0x1ACCE551 to the IWDT_LOCK register to unlock the register.
3. Configure IWDT_STALL[31:16] to set the IWDT clock division.
4. Configure IWDT_CTRL[2] to set the reset condition to require either one complete count or two complete counts for the first reset.
5. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
6. Configure IWDT_CTRL[1] to enable the IWDT overflow reset.
7. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
8. Write IWDT_LOAD[31:0] to set the IWDT counter load value.
9. Wait for IWDT_CTRL[31] to be set, i.e., the setting takes effect.
10. Write any value except 0x1ACCE551 to the IWDT_LOCK register to lock the register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

23 System Window Watchdog (WWDT)

23.1 Overview

The system window watchdog is a watchdog running synchronously with CPU, aiming at monitoring the running status of CPU in real time, so that it can reset CPU in the case of abnormal operation to avoid unpredictable consequences.

To ensure synchronization and real-time performance, the WWDT operates using the PCLK clock, with an internal prescaler circuit to generate a synchronized count enable signal.

23.2 Main Features

- Counter clock PCLK0 (APB0 PCLK)
- Up-counting mode: the counter counts from 0 to the overflow time.
- The overflow time can be selected as 1/4/16/64/128/256/512/1024/2048/4096/8192/16384/32768/65536 times of 4096 PCLK0 cycles.
- The window period is defined as the period when the counter is greater than or equal to 50% of the overflow time.
- With early warning interrupt capability, an interrupt will occur when the count reaches 75% of the overflow time.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

23.3 Functional Description

WWDT is disabled by default after the chip is reset, and the software needs to write 0x5A to the control register (WWDT_CTRL) to activate WWDT. After WWDT is activated, if the software writes 0xAC to WWDT control register (WWDT_CTRL) during the window period, the counter will be cleared. Once WWDT is enabled, it cannot be disabled again except by a reset.

The WWDT operates using PCLK with a built-in 4096 times prescaler circuit. The overflow time can be selected as 1/4/16/64/128/256/512/1024/2048/4096/8192/16384/32768/65536 times of 4096 PCLK0 cycles. The overflow time length is calculated as follows:

$$t_{WWDT} = f_{PCLK} \times 4096 \times N_{CFG}$$

The following table shows some calculation examples:

Table 23-1: WWDT Counter Overflow Time Calculation

PCLK0 Frequency (f_{PCLK})	**Overflow Length Configuration (N_{CFG})**	**Overflow Time (t_{WWDT}) (ms)**
42 MHz	1	0.097523
	4	0.390095
	16	1.560381
	64	6.241524
	128	12.483047
	256	24.966095
	512	49.93219
	1024	99.864381
	2048	199.728762
	4096	399.457524
	8192	798.915048
	16384	1597.830096
	32768	3195.660192
	65536	6391.320384

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

WWDT can only be cleared during the window period, otherwise a reset will be triggered directly. The enable window is the second half of the counter cycle, and the software should pay attention to query the count value before clearing the watchdog.

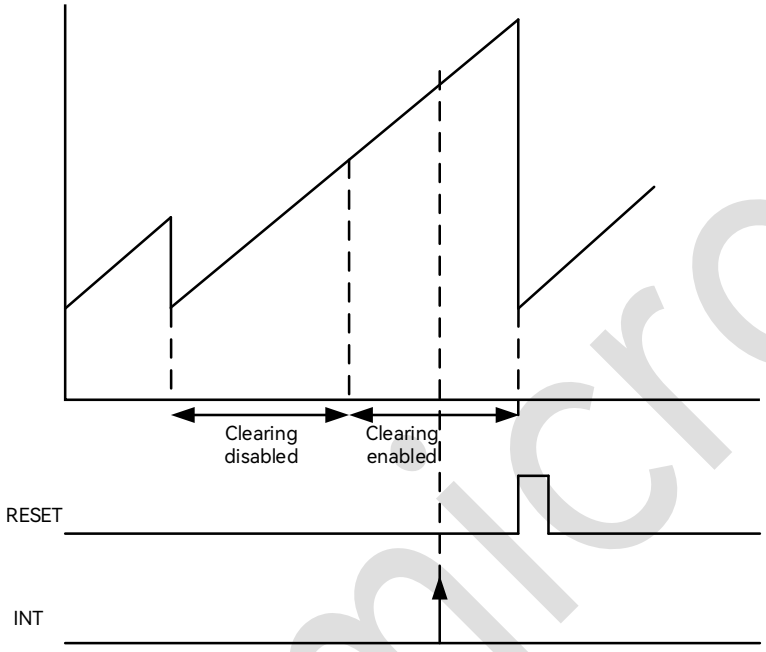

Figure 23-1: WWDT Counter Refresh Timing Diagram

WWDT will generate a CPU reset when any of the following events occurs:

- Counter overflow
- Write a value other than 0xAC to the WWDT control register (which can be used to trigger CPU software reset).
- Write 0xAC to the WWDT control register during window close period.

An early warning interrupt will be triggered when the counter reaches 75% of the overflow time.

23.4 Register Description

WWDT register base address: 0x40B0_5000

The registers are listed below:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 23-2: List of WWDT Registers

Offset Address	Name	Description
0x00	WWDT_CTRL	Control register
0x04	WWDT_CFG	Configuration register
0x08	WWDT_CNT	Counter register
0x0C	WWDT_IE	Interrupt enable
0x10	WWDT_IF	Interrupt flag register
0x14	WWDT_DIV_CNT	PCLK prescaler counter register

23.4.1 Control Register (WWDT_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CTRL	W	0x0	CPU writing 0x5A to this bit will start WWDT. After starting WWDT, CPU writing 0xAC to this bit will clear the counter. After starting WWDT, CPU writing a value other than 0xAC to this bit will reset the system.

23.4.2 Configuration Register (WWDT_CFG)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3:0	CFG	R/W	0x0	Configure watchdog overflow time: 0000: t_{PCLK} × 4096 × 1 0001: t_{PCLK} × 4096 × 4 0010: t_{PCLK} × 4096 × 16 0011: t_{PCLK} × 4096 × 64 0100: t_{PCLK} × 4096 × 128 0101: t_{PCLK} × 4096 × 256 0110: t_{PCLK} × 4096 × 512

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0111: t_{PCLK} × 4096 × 1024 1000: t_{PCLK} × 4096 × 2048 1001: t_{PCLK} × 4096 × 4096 1010: t_{PCLK} × 4096 × 8192 1011: t_{PCLK} × 4096 × 16384 1100: t_{PCLK} × 4096 × 32768 1101: t_{PCLK} × 4096 × 65536 1110: Reserved 1111: Reserved

23.4.3 Counter Register (WWDT_CNT)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	CNT	R	0x0	WWDT counter register value, which the software can query to know the WWDT timing progress.

23.4.4 Interrupt Enable Register (WWDT_IE)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	IE	R/W	0x0	WWDT interrupt enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

23.4.5 Interrupt Flag Register (WWDT_IF)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	IF	R/W1C	0x0	WWDT 75% timing interrupt flag: 0: No interrupt generated 1: The interrupt flag is set, and is cleared by writing 1.

23.4.6 PCLK Prescaler Counter Register (WWDT_DIV_CNT)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	DIV_CNT	R	0x0	This register is the PCLK prescaler counter. For every 4096 PCLK cycles, the value of the WWDT counter register increments by 1.

23.5 Operation Procedure

23.5.1 Interrupt Mode

1. Write 0xA5A55A5A to RCM_RCMPR register to enable its write operation.
2. Configure RCM_APB0CKENR[10] to enable the WWDT controller clock.
3. Configure RCM_APB0RSTR[10] to release the WWDT controller reset.
4. Write 0xFFFFFFFF to RCM_RCMPR register to end its write operation.
5. Configure WWDT_CFG[3:0] to set the overflow time of WWDT counter.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Initialize the WWDT interrupt, and configure WWDT_IE[0] to enable WWDT interrupt.
7. Write 0x5A to the WWDT_CTRL register to start the WWDT counter.
8. The WWDT window period is defined as the period when the counter is greater than or equal to 50% of the overflow time, and the following operations are performed in different time periods:
   A. When the count value is less than 50% of the overflow time, the watchdog is fed, and the system is reset directly.
   B. When the count value is greater than 50% of the overflow time, the watchdog is fed, the WWDT counter is cleared, and the counting restarts.
   C. When the count value reaches 75% of the overflow time, a WWDT warning interrupt is triggered.
   D. When the count value exceeds the overflow time, the system generates a reset.
9. Wait for the counter register (WWDT_CNT) to count to 75% of the overflow time, an interrupt is triggered. If the configuration register (WWDT_CFG) is configured with t_{PCLK} × 4096 × 1, then based on the count value in the PCLK prescaler counter register (WWDT_DIV_CNT), the interrupt is triggered when it counts to 0xC00.
10. The interrupt flag register (WWDT_IF) is cleared by writing 1 to it.

23.5.2 WWDT Clear Counter

Write 0xAC to the WWDT_CTRL register to clear the counter.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24 Real-time Clock (RTC)

24.1 Overview

The battery backup unit (BBU) is a set of functional modules that includes a real-time clock (RTC) with a calendar, an alarm, a periodic wake-up source, a tamper detector, and an 80-byte backup register.

24.2 Main Features

The BBU supports the following features:

- The backup system is always running.
  - Real-time clock (RTC) module
  - Tamper detector
  - 20-word (i.e., 80-byte) backup register
- Programmable alarm capable of generating interrupts

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.3 Block Diagram of Modules with System Connection

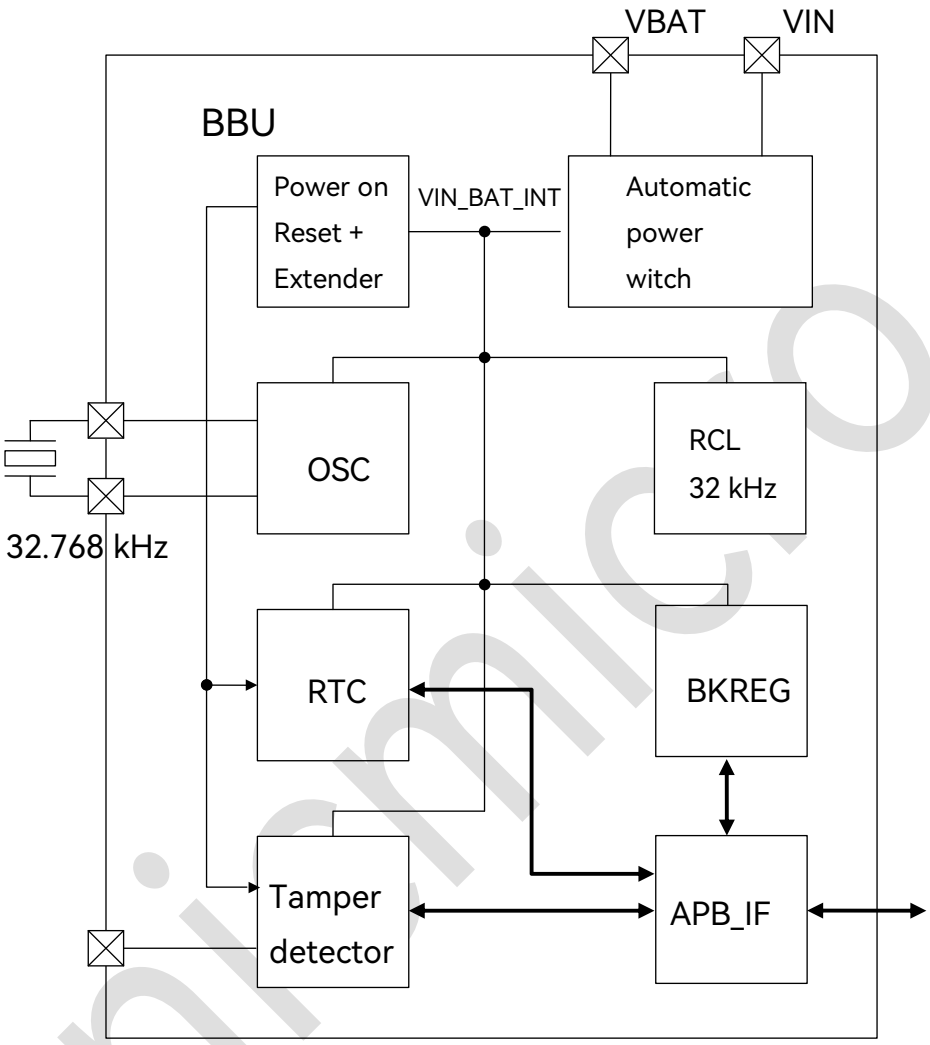

Figure 24-1 : BBU Module with Analog Module

Signal boundaries cross different power domains require isolation and level shifter units, as shown in Figure 24-1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.4 Module Description

24.4.1 RTC

24.4.1.1 Time and Calendar Counters

The RTC module is capable of providing real-time date and time information that can be accessed by the core and peripheral devices.

Date and time counters:

- Centisecond (1% of a second) counter (CENTISEC)
- Second counter (SECOND)
- Minute counter (MINUTE)
- Hour counter (HOUR)
- Week counter (WEEK)
- Day counter (DAY)
- Month counter (MONTH)
- Year counter (YEAR and CENTURY)

YEAR stands for the last two digits of the year. CENTURY indicates the 2000s (21st century) or 2100s (22nd century).

Each register is coded in BCD (binary-coded decimal) format.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.4.1.2 Centisecond Counter (CENTISEC)

This counter continuously increments from 00 to 99 every second. It increments once every 326~329 low-speed clock cycles, depending on the second period after calibration. The jitter in one second is controlled within only one HCLK cycle.

24.4.1.3 Time Counter (SECOND, MINUTE and HOUR)

The time counter behaves as follows:

SECOND: When the second counter counts from 59 to 00 (carry), the minute counter will be incremented by 1.

MINUTE: When the minute counter counts from 59 to 00 (carry), the hour counter will be incremented by 1.

HOUR: The hour counter consists of H20_PA (1 bit) and HOUR19 (5 bits).

- In 12-hour format, the day counter is incremented by 1 when [H20_PA, HOUR19] is incremented from [1,11] (11:00 p.m.) to [0,12] (12:00 a.m.).
- In 24-hour format, the day counter is incremented by 1 when [H20_PA, HOUR19] is incremented from [1,3] (23:00) to [0,0] (00:00).

By setting the HOUR12_24 bit in the RTC_TIME register, the hour format can be selected from either the 12-hour or 24-hour format.

- 0: 12-hour format
- 1: 24-hour format

As shown in the table below, HOUR is represented by two registers: HOUR19 and H20_PA, with the value of HOUR19 encoded in BCD format. The time format depends on the clock system

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

settings. In the 12-hour clock mode, H20_PA indicates AM or PM. In the 24-hour clock mode, H20_PA represents the hour of 20 plus.

Table 24-1: HOUR Register Coding

Time	24-hour Format		12-hour Format	
	H20_PA	HOUR19	H20_PA	HOUR19
0 a.m.	0	0	0	12
1 a.m.	0	1	0	1
2 a.m.	0	2	0	2
3 a.m.	0	3	0	3
4 a.m.	0	4	0	4
5 a.m.	0	5	0	5
6 a.m.	0	6	0	6
7 a.m.	0	7	0	7
8 a.m.	0	8	0	8
9 a.m.	0	9	0	9
10 a.m.	0	10	0	10
11 a.m.	0	11	0	11
0 p.m.	0	12	1	12
1 p.m.	0	13	1	1
2 p.m.	0	14	1	2
3 p.m.	0	15	1	3
4 p.m.	0	16	1	4
5 p.m.	0	17	1	5
6 p.m.	0	18	1	6
7 p.m.	0	19	1	7
8 p.m.	1	0	1	8
9 p.m.	1	1	1	9
10 p.m.	1	2	1	10
11 p.m.	1	3	1	11

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.4.1.4 Date Counter

Week counter (WEEK)

The counter value ranges from 0 to 6, and a 7 is treated as 0 when entered (coding rules: 0 or 7 = Sunday, 1 = Monday, ..., 5 = Friday, 6 = Saturday). The relationship between the WEEK counter and the actual day of the week is user-defined.

Calendar counter (DAY, MONTH, YEAR, CENTURY)

The automatic calendar function provides calendar numbers as shown below:

- DAY: the counter ranges from:
  - 01 to 31: In January, March, May, July, August, October, December
  - 01 to 30: In April, June, September, November
  - 01 to 29: In February of a leap year
  - 01 to 28: In February of a non-leap year

  When the day counter jumps back to 01, the month counter will carry forward.
- MONTH: it ranges from 01 to 12, and jumping back to 01 will carry it to the year counter.
- YEAR: it ranges from 00 to 99, and jumping back to 00 will clear the century counter.
- CENTURY:
  - When the century counter is 0, YEAR = 04, 08, …, 92, 96 are leap years.
  - When the century counter is 1, YEAR = 00, 04, 08, …, 92, 96 are leap years.
  - The leap year can be correctly identified in the range from 2000 to 2399.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.4.1.5 Time and Date Setting

To prevent accidental carry when writing the time and date counters, all timers will be frozen until the end of the write operation.

The time/date setting operation should follow the following sequence:

1. Write 1 to RTC_ACCESS[2], and the counter will stop timing.
2. Complete the write operation as soon as possible. **Note that any non-existing time or date settings will be ignored.**
3. Write 1 to RTC_ACCESS[3] and start timing from this point.

24.4.1.6 Time and Date Reading

Before reading the time and date registers, RTC needs to copy the real-time values of the time/date registers into the shadow reading register. This isolation ensures the integrity of the data being read, preventing changes to the time/date values due to time updates. The time/date read operation shall follow the sequence below:

1. Write 1 to RTC_ACCESS[0].
2. Complete the read operation as soon as possible.
3. Write 1 to RTC_ACCESS[1].

24.4.1.7 Alarm Clock

The BBU provides two programmable alarms: Alarm 1 and alarm 2.

There are two alarm configuration registers RTC_ALM1TIME and RTC_ALM1DATE, and one alarm enable register RTC_ALM1EN.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The RTC_ALM1EN register controls the status of the alarm. When all bits of the RTC_ALM1EN register are 0, alarm 1 is disabled. For each bit, 1 indicates that the alarm is only enabled when the corresponding register matches, while 0 indicates that the alarm is enabled regardless of the corresponding register value. The alarm flag and interrupt will be generated when the time increases to match the enabled time set by the RTC_ALM1DATE and RTC_ALM1TIME registers.

Example of alarm 1 setting is shown in the table below:

Table 24-2: Example of Alarm 1 Setting

ALM1 Date/Time Registers								ALM1_EN Register								Condition of Alarm 1
ALM1_YEAR	ALM1_MONTH	ALM1_DAY	ALM1_WEEK	ALM1_HOUR	ALM1_MINUTE	ALM1_SECOND	ALM1_CENTISEC	ALM1_EN_YEAR	ALM1_EN_MON	ALM1_EN_DAY	ALM1_EN_WEEK	ALM1_EN_HOUR	ALM1_EN_MIN	ALM1_EN_SEC	ALM1_EN_CS	
2	3	4	5	6	7	8	9	0	0	0	0	0	0	0	0	Disabled
2	3	4	5	6	7	8	9	0	0	0	0	0	1	1	0	07:08 in every hour
2	3	4	5	6	7	8	9	1	1	1	0	1	1	1	0	06:07:08 on March 4, 2002
2	3	4	5	6	7	8	9	0	0	1	0	1	1	1	0	06:07:08 on the 4th of every month
2	3	4	5	6	7	8	9	0	0	0	1	1	1	1	0	06:07:08 on every Friday

Alarm 2 generates periodic interrupts. Depending on the register settings, the interrupt interval varies from every minute to every 1/128th of a second.

Alerts can be checked by registers RTC_INTRAW and RTC_INTSTA.

Registers RTC_INTEN and RTC_INTCLR control the output of the above interrupts to the CPU, and the interrupt status retains its value until it is cleared by APB access.

24.4.1.8 Oscillator Calibration

As shown in the figure below, this module can calibrate the input frequency of the low-speed clock to within ±5 ppm of the nominal 32.768 kHz. The calibration function changes the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

maximum value of the clock counter from 32768 to the specified value. The calibration step is ± 1 cycle, and the maximum calibration range is from -128 cycles to +127 cycles.

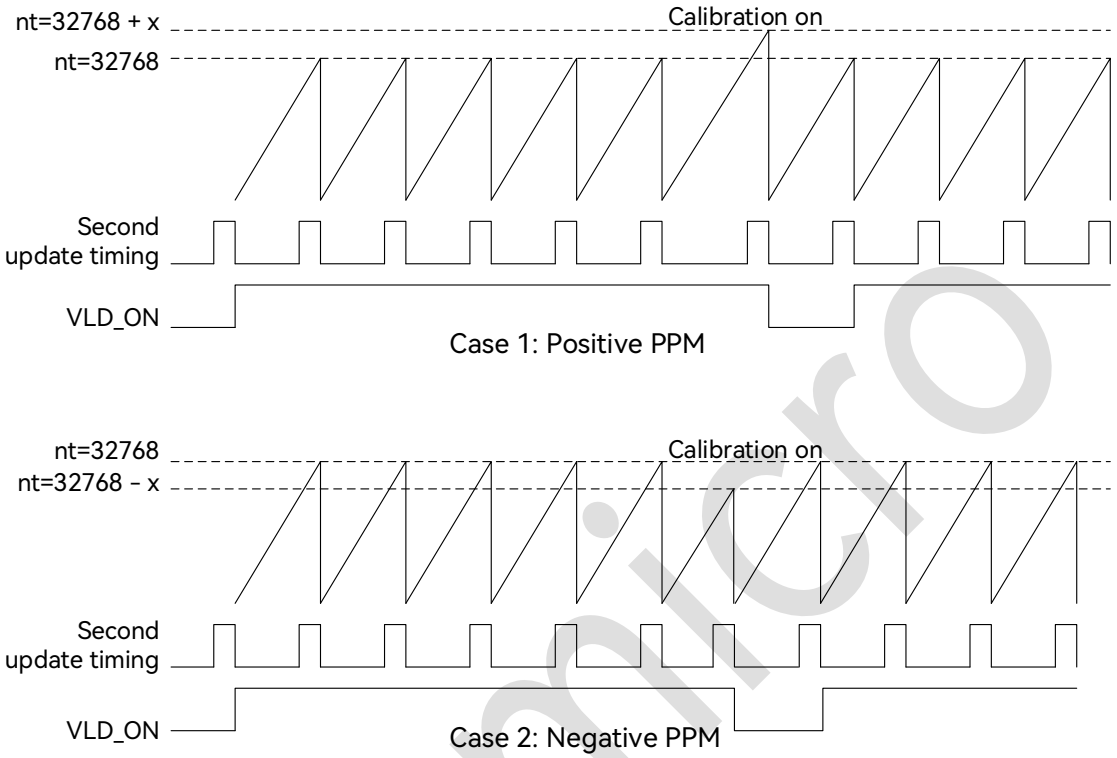

Figure 24-2: Oscillator Calibration

The 8-bit trim register (TRIM_VALUE[7:0], or TRIM[7:0] for short) in RTC_TRIM register stores a signed value that can adjust the maximum value of the low-speed clock counter. TRIM[7] is its symbol, where “0” represents a non-negative value and “1” represents a negative value. When TRIM[7] is “1”, TRIM[6:0] is considered a binary complement.

The TRIM_MODE register defines the frequency at which this alignment is performed.

- 0x0: once every 60 seconds (whenever SECOND = 00)
- 0x1: once every 30 seconds (whenever SECOND = 00 or 30)
- 0x2: once every 15 seconds (whenever SECOND = 00, 15, 30 or 45)
- 0x3: once every 6 seconds (whenever SECOND = 00, 06, 12, 18, 24, 30, 36, 42, 48 or 54)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

This compensation scheme ensures that the minute refresh point is independent of the oscillator frequency accuracy.

Examples of oscillator calibration calculations are listed in tables Table 24-3 to 24-6.

TRIM_MODE = 0x0

Table 24-3 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x0

Binary TRIM[7:0]	Compensation Value	Total Clock Cycle in 60 Seconds	[ppm]
10000000	-128	32768 * 59 + 32640 * 1	-65.1
11000000	-64	32768 * 59 + 32704 * 1	-32.2
11111111	-1	32768 * 59 + 32767 * 1	-0.5
00000000	0	32768 * 60	0.0
00000001	1	32768 * 59 + 32769 * 1	0.5
00111111	63	32768 * 59 + 32831 * 1	32
01111111	127	32768 * 59 + 32895 * 1	64.6

TRIM_MODE = 0x1

Table 24-4 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x1

Binary TRIM[7:0]	Compensation Value	Total Clock Cycle in 30 Seconds	[ppm]
10000000	-128	32768 * 29 + 32640 * 1	-130.2
11000000	-64	32768 * 29 + 32704 * 1	-65.1
11111111	-1	32768 * 29 + 32767 * 1	-1
00000000	0	32768 * 30	0.0
00000001	1	32768 * 29 + 32769 * 1	1
00111111	63	32768 * 29 + 32831 * 1	64.6
01111111	127	32768 * 29 + 32895 * 1	129.2

TRIM_MODE = 0x2

Table 24-5 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x2

Binary TRIM[7:0]	Compensation Value	Total Clock Cycle in 15 Seconds	[ppm]
10000000	-128	32768 * 14 + 32640 * 1	-244.2
11000000	-64	32768 * 14+ 32704 * 1	-122

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Binary TRIM[7:0]	Compensation Value	Total Clock Cycle in 15 Seconds	[ppm]
11111111	-1	32768 * 14 + 32767 * 1	-1.9
00000000	0	32768 * 15	0.0
00000001	1	32768 * 14 + 32769 * 1	1.9
00111111	63	32768 * 14 + 32831 * 1	120.2
01111111	127	32768 * 14 + 32895 * 1	242.2

TRIM_MODE = 0x3

Table 24-6 : Example of Oscillator Calibration Calculation with TRIM_MODE = 0x3

Binary TRIM[7:0]	Compensation Value	Total Clock Cycle in 6 Seconds	[ppm]
10000000	-128	32768 * 5 + 32640 * 1	-651
11000000	-64	32768 * 5 + 32704 * 1	-325.5
11111111	-1	32768 * 5 + 32767 * 1	-5.1
00000000	0	32768 * 6	0.0
00000001	1	32768 * 5 + 32769 * 1	5.1
00111111	63	32768 * 5 + 32831 * 1	320.4
01111111	127	32768 * 5 + 32895 * 1	646

For a given target ppm value, TRIM and TRIM_MODE shall be calculated correctly to make the actual ppm as close to the target as possible. It is also recommended that adjustments be performed more frequently at smaller steps by selecting a higher TRIM_MODE setting whenever possible.

The output signal VLD_ON_O is set to 1 for tunning the clock calibration. VLD_ON_O outputs a high level for 6, 15, 30 or 60 seconds according to the VLD_MODE setting, which facilitates timing measurements using an external reference clock.

24.4.2 Tamper Detector

The tamper input pin can be used to monitor specific locations in the system. A change in the state of the tamper input may indicate that a tamper event has occurred. The event to be monitored can be configured as either input (rising or falling or double) edge(s) detection or

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

(high or low) level detection. In the case of level detection, the debounce circuit can filter out pulses shorter than 6 milliseconds (the duration is configurable).

When tampering is detected, the backup system behaves as follows:

- Store the timestamp (year, month, date, hour, minute and second) into one of three sets of registers. These registers always retain information about the last three tamper events, which are accessible by the ARM core in normal power consumption mode.
- The maximum number of tamper events detected is 63.
- If enabled, an interrupt request is sent to the CPU. In battery backup mode, the logic circuits are powered down and therefore any interrupt requests are ignored.

24.4.3 Backup Register

The backup register is organized with 32 bits as one word, with a total space of 20 words.

It can be accessed by ARM CPU through APB interface.

24.5 Register Description

RTC register base address: 0x40B0_0000

The registers are listed below:

Table 24-7: List of RTC Registers

Offset Address	Name	Description
0x00	RTC_TIME	RTC time register
0x04	RTC_DATE	RTC date register
0x08	RTC_ACCESS	RTC counter read-write control register
0x10	RTC_TRIM	RTC calibration control register
0x14	RTC_TEST	RTC incremental debugging function register
0x20	RTC_ALM1TIME	Alarm 1 time setting register

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x24	RTC_ALM1DATE	Alarm 1 date setting register
0x28	RTC_ALM1EN	Alarm 1 enable register
0x2C	RTC_ALM2SETTING	Alarm 2 setting register
0x30	RTC_TAMPCTRL	Tamper detector control and configuration register
0x34	RTC_TAMPCNT	Tamper count record register
0x38	RTC_TAMP3TIME	Third new tamper event timestamp register
0x3C	RTC_TAMP3DATE	Third new tamper event datestamp register
0x40	RTC_TAMP2TIME	Second new tamper event timestamp register
0x44	RTC_TAMP2DATE	Second new tamper event datestamp register
0x48	RTC_TAMP1TIME	Latest tamper event timestamp register
0x4C	RTC_TAMP1DATE	Latest tamper event datestamp register
0x50	RTC_INTEN	RTC interrupt enable register
0x54	RTC_INTRAW	RTC raw interrupt status register
0x58	RTC_INTSTA	RTC interrupt valid status register
0x5C	RTC_INTCLR	RTC interrupt status clear register
0x60	RTC_BKREG0	Backup register, 20 words or 80 bytes in total
0x64	RTC_BKREG1	
…	…	
0xAC	RTC_BKREG19	

Note: Since all registers of RTC are in the system low-speed (32 kHz) clock domain, reading or writing any register requires two low-speed clock cycles.

24.5.1 RTC Time Register (RTC_TIME)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	HOUR12_24	R/W	0x0	Setting of hour format: 0: 12-hour format 1: 24-hour format
30	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
29	H20_PA	R/W	0x0	In the 12-hour format, H20_PA indicates AM or PM. 0: AM 1: PM In the 24-hour format, H20_PA represents the hour of 20 plus. For details, please refer to section 23.4.1.3, “Hour Format”.
28:24	HOUR19	R/W	0x0	The value of HOUR19 is encoded in BCD (binary-coded decimal) format. The hour format depends on HOUR12_24. In 12-hour format, when [H20_PA, HOUR19] counts from [1, 0x11] (11:00 p.m.) to [0, 0x12] (12:00 a.m.), the day counter will be incremented. In 24-hour format, when [H20_PA, HOUR19] counts from [1, 3] (23:00) to [0, 0] (00:00), the day counter will be incremented.
23	RSV	-	-	Reserved
22:16	MINUTE	R/W	0x0	The value of the minute counter is encoded in BCD format, within the range from 0x00 to 0x59. When it counts from 0x59 to 0x00, the hour counter will increment.
15	RSV	-	-	Reserved
14:8	SECOND	R/W	0x0	The value of the second counter is encoded in BCD format, within the range from 0x00 to 0x59. When it counts from 0x59 to 0x00, the minute counter will increment.
7:0	CENTISEC	R	0x0	The value of the centisecond counter is encoded in BCD format, within the range from 0x00 to 0x99. When it counts from 0x99 to 0x00, the second counter will increment.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Notes:

1. The RTC_ACCESS register shall be written before and after reading and writing this register.
2. Invalid time value written to RTC_TIME will be ignored. The validity of values written to the HOUR, MINUTE and SECOND registers is judged separately, and an invalid write to one register will not affect the writing of valid values to the other registers.

24.5.2 RTC Date Register (RTC_DATE)

Offset address: 0x04

Reset value: 0x8001 0106

Bit	Name	Attribute	Reset Value	Description
31	CENTURY	R/W	0x1	Century counter When it becomes 0, it will no longer increment with the year counter. 0: 2100s to 2300s (22nd to 24th century) 1: 2000s (21st century)
30	RSV	-	-	Reserved
29:22	YEAR	R/W	0x0	This register represents the last two digits of the decimal year. This value is encoded in BCD format, within the range from 0x00 to 0x99. When the year counter counts from 0x99 to 0x00, the century counter will become 0. When CENTURY = 0, the years 04, 08, …, 92 and 96 are leap years, while the year 00 is a non-leap year. When CENTURY = 1, the years 00, 04, 08, …, 92 and 96 are leap years.
21	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
20:16	MONTH	R/W	0x1	The value of the month counter is encoded in BCD format, within the range from 0x1 to 0x12. When it counts from 0x12 to 0x1, the year counter will increment.
15:14	RSV	-	-	Reserved
13:8	DAY	R/W	0x1	The value of the day counter is encoded in BCD format, within the range from: 0x1 to 0x31, in January, March, May, July, August, October and December. 0x1 to 0x30, in April, June, September and November. 0x1 to 0x29, in February of a leap year. 0x1 to 0x28, in February of a normal year. When the day counter counts from the end of the month to 0x1, the month counter will be incremented.
7:3	RSV	-	-	Reserved
2:0	WEEK	R/W	0x6	The week counter ranges from 0 to 6. Wherein, 0 means Sunday, 1 means Monday, and so on. The relationship between the week counter and the date is user-defined.

Notes:

1. The RTC_ACCESS register shall be written before and after reading and writing this register.

2. Invalid date values (combination of date and month) written to RTC_DATE will be ignored.

24.5.3 RTC Counter Read-write Control Register (RTC_ACCESS)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	0x0	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	WRSTP	W	0x0	Writing 1 to this bit indicates that CPU has finished writing to the RTC_TIME and RTC_DATE registers. After this bit is written with 1, the CENTISEC counter will continue to count.
2	WRSTA	W	0x0	Writing 1 to this bit indicates that CPU has started writing to the RTC_TIME and RTC_DATE registers. After this bit is written with 1, the CENTISEC counter will retain its written value until 1 is written to WRSTP.
1	RDSTP	W	0x0	Writing 1 to this register indicates that CPU has finished reading from the RTC_TIME and RTC_DATE registers. After this bit is written with 1, the shadow registers RTC_TIME and RTC_DATE will latch the values of the time and date counters.
0	RDSTA	W	0x0	Writing 1 to this register indicates that CPU has started reading from the RTC_TIME and RTC_DATE registers. After this bit is written with 1, the shadow registers RTC_TIME and RTC_DATE will continuously change with the time and date counters.

24.5.4 RTC Calibration Control Register (RTC_TRIM)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	0x0	Reserved
13	VLD_EN	R/W	0x0	Validation enable: 0: Validation disabled 1: Validation enabled; the VLD_ON_O pulse is set periodically.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12:11	VLD_MODE	R/W	0x0	Setting of validation duration for VLD_ON_O: 0: VLD_ON_O remains high for 60 s. 1: VLD_ON_O remains high for 30 s. 2: VLD_ON_O remains high for 15 s. 3: VLD_ON_O remains high for 6 s.
10	TRIM_EN	R/W	0x0	Calibration enable: 0: Calibration disabled 1: Calibration enabled
9:8	TRIM_MODE	R/W	0x0	Calibration frequency setting: 0: Once every 60 s 1: Once every 30 s 2: Once every 15 s 3: Once every 6 s
7:0	TRIM_VALUE	R/W	0x0	Signed compensation constant, indicating the compensation amount used to align the clock error.

24.5.5 RTC Counter Increment Function Test Register (RTC_TEST)

The increment function of RTC counter is used for testing or debugging.

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	TEST_EN	R/W	0x0	Increment function enable: 0: Normal operation, with increment function disabled 1: Increment function enabled
30:5	RSV	-	0x0	Reserved
4	MON_CNT_UP	W	0x0	Month counter increment enable: 0: No action 1: Both the month counter and the second counter are incremented.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	DAY_CNT_UP	W	0x0	Day counter increment enable: 0: No action 1: Both the day counter and the second counter are incremented.
2	HR_CNT_UP	W	0x0	Hour counter increment enable: 0: No action 1: Both the hour counter and the second counter are incremented.
1	MIN_CNT_UP	W	0x0	Minute counter increment enable: 0: No action 1: Both the minute counter and the second counter are incremented.
0	SEC_CNT_UP	W	0x0	Second counter increment enable: 0: No action 1: The second counter is incremented.

Notes:

1. The RTC_TEST[31] bit must be set to 1 before setting any other increment function bits. When the RTC_TEST[31] bit switches from 0 to 1, any simultaneous write operation to other increment function bits will be invalid.
2. Every time the increment function is enabled, the second counter will count up. If multiple increment bits are set in a single write operation, the second counter will still increment by 1.
3. If the time or date counters generate a carry during incrementing, it will cause a cascading increment of the higher-level counters.
4. If the day counter increments, the week counter will also increment. However, if the month counter increments while the day counter does not increment, the week counter will remain unchanged.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.6 Alarm 1 Time Setting Register (RTC_ALM1TIME)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	ALM1_HOUR12_24	R/W	0x0	Hour format setting for alarm 1: 0: 12-hour format 1: 24-hour format
30	RSV	-	0x0	Reserved
29	ALM1_H20_PA	R/W	0x0	Hour setting for alarm 1: The hour format follows the setting of ALM1_HOUR12_24.
28:24	ALM1_HOUR19	R/W	0x0	
23	RSV	-	0x0	Reserved
22:16	ALM1_MINUTE	R/W	0x0	Minute setting for alarm 1
15	RSV	-	0x0	Reserved
14:8	ALM1_SECOND	R/W	0x0	Second setting for alarm 1
7:0	ALM1_CENTISEC	R/W	0x0	Centisecond setting for alarm 1

24.5.7 Alarm 1 Date Setting Register (RTC_ALM1DATE)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved
29:22	ALM1_YEAR	R/W	0x0	Year setting for alarm 1
21	RSV	-	0x0	Reserved
20:16	ALM1_MONTH	R/W	0x0	Month setting for alarm 1
15:14	RSV	-	0x0	Reserved
13:8	ALM1_DAY	R/W	0x0	Date setting for alarm 1
7:3	RSV	-	0x0	Reserved
2:0	ALM1_WEEK	R/W	0x0	Week setting for alarm 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.8 Alarm 1 Setting Enable Register (RTC_ALM1EN)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	ALM1_EN	R/W	0x0	Alarm 1 enable: 0: Alarm 1 disabled 1: Alarm 1 enabled
30:8	RSV	-	0x0	Reserved
7	ALM1_EN_YEAR	R/W	0x0	Year condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
6	ALM1_EN_MON	R/W	0x0	Month condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
5	ALM1_EN_DAY	R/W	0x0	Date condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
4	ALM1_EN_WEEK	R/W	0x0	Week condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
3	ALM1_EN_HOUR	R/W	0x0	Hour condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
2	ALM1_EN_MIN	R/W	0x0	Minute condition control bit for alarm 1: 0: Condition not applied 1: Condition applied

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	ALM1_EN_SEC	R/W	0x0	Second condition control bit for alarm 1: 0: Condition not applied 1: Condition applied
0	ALM1_EN_CS	R/W	0x0	Centisecond condition control bit for alarm 1: 0: Condition not applied 1: Condition applied

24.5.9 Alarm 2 Setting Register (RTC_ALM2SETTING)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	ALM2_EN	R/W	0x0	Alarm 2 enable: 0: Alarm 2 disabled 1: Alarm 2 enabled
30:4	RSV	-	-	Reserved
3:0	ALM2_SETTING	R/W	0x0	Alarm 2 interrupt output cycle: 0: No output 1: 1 s 2: 1/2 s 3: 1/4 s 4: 1/8 s 5: 1/16 s 6: 1/32 s 7: 1/64 s 8: 1/128 s 9: 1 min Others: 1 s

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.10 Tamper Detector Control and Configuration Register (RTC_TAMPCTRL)

Offset address: 0x30

Reset value: 0x0000 0030

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved
6	TAMP_CNT_CLR	W	0x0	Clear bit of the tamper counter Write 1 to this bit to clear the tamper counter TAMP_CNT.
5:4	TAMP_DBNC	R/W	0x3	Debounce time for TAMPER_IN: 0: No debounce 1: 2 ms 2: 4 ms 3: 6 ms
3:1	TAMP_EDGE	R/W	0x0	Events to be detected on the TAMPER_IN pin: 0: Rising edge 1: Falling edge 2: Both edges 3: High level lasting longer than the set value of TAMP_DBNC 4: Low level lasting longer than the set value of TAMP_DBNC Others: Tamper detector disabled
0	TAMP_EN	R/W	0x0	Tamper detector enable bit for the TAMPER_IN pin: 0: Tamper detector disabled 1: Tamper detector enabled

24.5.11 Tamper Count Record Register (RTC_TAMPCNT)

Offset address: 0x34

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	0x0	Reserved
5:0	TAMP_CNT	R	0x0	Tamper event counter: 0x00: No tamper event 0x01–0x3E: Number of tamper events 0x3F: 63 or more tamper events

24.5.12 Third New Tamper Event Timestamp Register (RTC_TAMP3TIME)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved
29	TAMP3_H20_PA	R	0x0	Hour of the third new tamper event The hour format follows the setting of HOUR12_24.
28:24	TAMP3_HOUR19	R	0x0	
23	RSV	-	0x0	Reserved
22:16	TAMP3_MINUTE	R	0x0	Minute of the third new tamper event
15	RSV	-	0x0	Reserved
14:8	TAMP3_SECOND	R	0x0	Second of the third new tamper event
7:0	RSV	-	0x0	Reserved

24.5.13 Third New Tamper Event Datestamp Register (RTC_TAMP3DATE)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
29:22	TAMP3_YEAR	R	0x0	Year of the third new tamper event
21	RSV	-	0x0	Reserved
20:16	TAMP3_MONTH	R	0x0	Month of the third new tamper event
15:14	RSV	-	0x0	Reserved
13:8	TAMP3_DAY	R	0x0	Date of the third new tamper event
7:0	RSV	-	0x0	Reserved

24.5.14 Second New Tamper Event Timestamp Register (RTC_TAMP2TIME)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved
29	TAMP2_H20_PA	R	0x0	Hour of the second new tamper event The hour format follows the setting of HOUR12_24.
28:24	TAMP2_HOUR19	R	0x0	
23	RSV	-	0x0	Reserved
22:16	TAMP2_MINUTE	R	0x0	Minute of the second new tamper event
15	RSV	-	0x0	Reserved
14:8	TAMP2_SECOND	R	0x0	Second of the second new tamper event
7:0	RSV	-	0x0	Reserved

24.5.15 Second New Tamper Event Datestamp Register (RTC_TAMP2DATE)

Offset address: 0x44

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	--	I	0x0	Reserved
29:22	TAMP2_YEAR	R	0x0	Year of the second new tamper event
21	--	I	0x0	Reserved
20:16	TAMP2_MONTH	R	0x0	Month of the second new tamper event
15:14	--	I	0x0	Reserved
13:8	TAMP2_DAY	R	0x0	Date of the second new tamper event
7:0	--	I	0x0	Reserved

24.5.16 Latest Tamper Event Timestamp Register (RTC_TAMP1TIME)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved
29	TAMP1_H20_PA	R	0x0	Hour of the latest tamper event The hour format follows the setting of HOUR12_24.
28:24	TAMP1_HOUR19	R	0x0	
23	RSV	-	0x0	Reserved
22:16	TAMP1_MINUTE	R	0x0	Minute of the latest tamper event
15	RSV	-	0x0	Reserved
14:8	TAMP1_SECOND	R	0x0	Second of the latest tamper event
7:0	RSV	-	0x0	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.17 Latest Tamper Event Datestamp Register (RTC_TAMP1DATE)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	0x0	Reserved
29:22	TAMP1_YEAR	R	0x0	Year of the latest tamper event
21	RSV	-	0x0	Reserved
20:16	TAMP1_MONTH	R	0x0	Month of the latest tamper event
15:14	RSV	-	0x0	Reserved
13:8	TAMP1_DAY	R	0x0	Date of the latest tamper event
7:0	RSV	-	0x0	Reserved

24.5.18 RTC Interrupt Request Enable Register (RTC_INTEN)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	0x0	Reserved
2	TAMP_INT_EN	R/W	0x0	Tamper interrupt request enable: 0: Tamper interrupt request disabled 1: Tamper interrupt request enabled
1	ALM2_INT_EN	R/W	0x0	Alarm 2 interrupt request enable: 0: Alarm 2 interrupt request disabled 1: Alarm 2 interrupt request enabled
0	ALM1_INT_EN	R/W	0x0	Alarm 1 interrupt request enable: 0: Alarm 1 interrupt request disabled 1: Alarm 1 interrupt request enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.19 RTC Raw Interrupt Status Register (RTC_INTRAW)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	0x0	Reserved
2	TAMP_INT_RAW	R	0x0	Tamper raw interrupt status: 0: Not triggered 1: Triggered
1	ALM2_INT_RAW	R	0x0	Alarm 2 raw interrupt status: 0: Not triggered 1: Triggered
0	ALM1_INT_RAW	R	0x0	Alarm 1 raw interrupt status: 0: Not triggered 1: Triggered

24.5.20 Interrupt Status Register after RTC Enable (RTC_INTSTA)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	0x0	Reserved
2	TAMP_INT_STA	R	0x0	Interrupt status after tamper is enabled: 0: Not triggered 1: Triggered
1	ALM2_INT_STA	R	0x0	Interrupt status after alarm 2 is enabled: 0: Not triggered 1: Triggered
0	ALM1_INT_STA	R	0x0	Interrupt status after alarm 1 is enabled: 0: Not triggered 1: Triggered

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.5.21 RTC Interrupt Status Clear Register (RTC_INTCLR)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	0x0	Reserved
2	TAMP_INT_CLR	W	0x0	Tamper interrupt status clear: 0: No action 1: Tamper interrupt status cleared
1	ALM2_INT_CLR	W	0x0	Alarm 2 interrupt status clear: 0: No action 1: Alarm 2 interrupt status cleared
0	ALM1_INT_CLR	W	0x0	Alarm 1 interrupt status clear: 0: No action 1: Alarm 1 interrupt status cleared

24.5.22 Backup Register (RTC_BKREG_n)

Backup register: 20 words (i.e., 80 bytes) in total

Offset address: 0x60 + 0x4 × n, wherein n = 0, 1, ..., 19

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	BKREG_n	R/W	0x0	Backup register n (n = 0, 1, ..., 19)

24.6 Operation Procedure

24.6.1 RTC Clock Enabling

1. Write 0xABCD to the PMU_CPR register to enable its write operation.
2. Configure PMU_FCCR[6] to enable the RTC controller clock.
3. Configure PMU_FRCR[6] to release the RTC controller reset.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Configure PMU_SASR[0] to set the external VBAT power supply (that is, select the external VBAT power supply).

5. Write 0x459E to the PMU_CPR register to end its write operation.

24.6.2 RTC Time Reading

1. Enable the RTC clock.

2. Start initializing the RTC date and time by writing 0x4 to the RTC_ACCESS register to enable RTC write access.

3. Write the date and time to the RTC_DATA and RTC_TIME registers.

4. End the initialization of the RTC date and time by writing 0x8 to the RTC_ACCESS register to finish RTC writing.

5. Write 0x1 to the RTC_ACCESS register to update the date and time.

6. Write 0x2 to the RTC_ACCESS register to save the current date and time.

7. Read the RTC_DATA and RTC_TIME registers to obtain the date and time.

24.6.3 RTC Alarm 1 Setting

1. Enable the RTC clock.

2. Start initializing the RTC date and time by writing 0x4 to the RTC_ACCESS register to enable RTC write access.

3. Write the date and time to the RTC_DATA and RTC_TIME registers.

4. End the initialization of the RTC date and time by writing 0x8 to the RTC_ACCESS register to finish RTC writing.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. Set the alarm 1 timing by writing the date and time to the RTC_ALM1DATE and RTC_ALM1TIME registers.
6. Initialize the alarm 1 interrupt by configuring RTC_INTEN[0] to enable the alarm 1 interrupt.
7. Enable the corresponding condition application bit in the RTC_ALM1EN register by configuring RTC_ALM1EN[31] to enable alarm 1.

24.6.4 RTC Alarm 2 Setting

1. Enable the RTC clock.
2. Start initializing the RTC date and time by writing 0x4 to the RTC_ACCESS register to enable RTC write access.
3. Write the date and time to the RTC_DATA and RTC_TIME registers.
4. End the initialization of the RTC date and time by writing 0x8 to the RTC_ACCESS register to finish RTC writing.
5. Configure RTC_ALM2SETTING[3:0] to set the interrupt output period for alarm 2.
6. Initialize the alarm 2 interrupt by configuring RTC_INTEN[1] to enable the alarm 2 interrupt.
7. Configure RTC_ALM2SETTING[31] to enable alarm 2.

24.6.5 RTC Backup Register

1. Write 0xABCD to the PMU_CPR register to enable its write operation.
2. Configure PMU_SASR[0] to set the external VBAT power supply (that is, select the external VBAT power supply).
3. Write 0x459E to the PMU_CPR register to end its write operation.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Write data to the RTC_BKREGn registers (one backup register stores 4 bytes, totaling 80 bytes).
5. With the external VBAT power supply connected, VDDH is powered off first and then re-powered on to read the backup register again, which is consistent with that before power off.

24.6.6 RTC Tamper Detection

1. Configure GPIOC13 pin as an input.
2. Write 0xABCD to the PMU_CPR register to enable its write operation.
3. Configure PMU_PDWKCR[3] to set the RTC_TAMPER wakeup event to active.
4. Write 0x459E to the PMU_CPR register to end its write operation.
5. Enable the RTC clock.
6. Start initializing the RTC date and time by writing 0x4 to the RTC_ACCESS register to enable RTC write access.
7. Write the date and time to the RTC_DATA and RTC_TIME registers.
8. End the initialization of the RTC date and time by writing 0x8 to the RTC_ACCESS register to finish RTC writing.
9. Configure RTC_TAMPCTRL[3:1] to set the levels of the TAMPER_IN pin to be monitored.
10. Set RTC_TAMPCTRL[6] to 1 to clear the tamper counter value.
11. Set RTC_INTEN[2] to 1 to enable the tamper interrupt.
12. Set RTC_TAMPCTRL[0] to 1 to enable the tamper detection.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

24.6.7 RTC Tamper Counter Value Clearing

Set RTC_TAMP_CTRL[6] to 1 to clear the tamper counter value.

24.6.8 RTC Calibration

1. Enable the RTC clock.
2. Start initializing the RTC date and time by writing 0x4 to the RTC_ACCESS register to enable RTC write access.
3. Write the date and time to the RTC_DATA and RTC_TIME registers.
4. End the initialization of the RTC date and time by writing 0x8 to the RTC_ACCESS register to finish RTC writing.
5. Configure RTC_TRIM[9:8] and RTC_TRIM[7:0] to set the calibration frequency and compensation constant, and set RTC_TRIM[10] to 1 to enable the calibration function.
6. Configure PC5 to alternate function 0, which can output a parity level.

24.6.9 RTC Increment Test

1. Enable the RTC clock.
2. Clear the RTC_TEST register and configure it to increment the corresponding counter.
3. Write 0x1 to the RTC_ACCESS register to update the date and time.
4. Write 0x2 to the RTC_ACCESS register to save the current date and time.
5. Read the RTC_DATA and RTC_TIME registers to obtain the date and time.
6. Repeat steps 2 to 5 to observe the corresponding counter increment.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25 CAN Bus Controller (CAN)

25.1 Overview

The CAN controller being compliant with CAN2.0 A/B protocol can be used in the fields of automotive electronics and industrial control.

25.2 Main Features

- In compliant with CAN2.0A/B protocol
- Supporting CAN format
- Up to 8 bytes data frame
- Up to 1 Mbps in CAN2.0B
- 64-byte RX FIFO
- 16-byte TX FIFO
- Transmission stop available
- Readable error counter

25.3 Pin Description

Table 25-1: CAN Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
CAN1_TX	PA12, PB9, PD1	Output	Transmitting data
CAN1_RX	PA11, PB8, PD0	Input	Receiving data
CAN2_TX	PB6, PB13	Output	Transmitting data
CAN2_RX	PB5, PB12	Input	Receiving data

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.4 Functional Description

25.4.1 Standard Frame Memory Buffer Layout

Messages to be transmitted/received are stored in the transmit/receive buffer, where the transmit buffer can hold one message at a time, while the receive FIFO buffer can hold multiple messages.

The memory address cannot be used directly and can only be accessed indirectly through the TXBUFn and RXBUFn registers. The layout of the transmit/receive buffers for a standard frame is as follows:

Table 25-2: Standard Frame Transmit/Receive Buffer Layout Diagram

offset	data bits	Standard Frame buffer content for TX/RX RAM							
0x000	7:0	FF	RTR	X/0	X/0	DLC3	DLC2	DLC1	DLC0
	15:8	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3
	23:16	ID2	ID1	ID0	X/RTR	X/0	X/0	X/0	X/0
	31:24	data 1 (DLC<1 then free)							
0x001	7:0	data 2 (DLC<2 then free)							
	15:8	data 3 (DLC<3 then free)							
	23:16	data 4 (DLC<4 then free)							
	31:24	data 5 (DLC<5 then free)							
0x002	7:0	data 6 (DLC<6 then free)							
	15:8	data 7 (DLC<7 then free)							
	23:16	data 8 (DLC<8 then free)							
	31:24	free							

Notes:

- FF (frame format): 1 for extended frame, 0 for standard frame
- RTR (remote transmission request bit): 1 for remote frame, 0 for data frame
- X: ignored
- DLC: Data length code
- ID: CAN identifier
- Data (1...8): Data bytes (7-MSB, 0-LSB)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.4.2 Extended Frame Memory Buffer Layout

Table 25-3: Extended Frame Memory Buffer Layout Diagram

offset	data bits	Extended Frame buffer content for TX/RX RAM							
0x000	7:0	FF	RTR	X/0	X/0	DLC3	DLC2	DLC1	DLC0
	15:8	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21
	23:16	ID20	ID19	ID18	ID17	ID16	ID15	ID14	ID13
	31:24	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5
0x001	7:0	ID4	ID3	ID2	ID1	ID0	X/RTR	X/0	X/0
	15:8	data 1 (DLC<1 then free)							
	23:16	data 2 (DLC<2 then free)							
	31:24	data 3 (DLC<3 then free)							
0x002	7:0	data 4 (DLC<4 then free)							
	15:8	data 5 (DLC<5 then free)							
	23:16	data 6 (DLC<6 then free)							
	31:24	data 7 (DLC<7 then free)							
0x003	7:0	data 8 (DLC<8 then free)							
	15:8	free							
	23:16	free							
	31:24	free							

Notes:

- FF (frame format): 1 for extended frame, 0 for standard frame
- RTR (remote transmission request bit): 1 for remote frame, 0 for data frame
- X: ignored
- DLC: Data length code
- ID: CAN identifier
- Data (1...8): Data bytes (7-MSB, 0-LSB)

25.5 List of Registers

CAN1 register base address: 0x40E0_0000

CAN2 register base address: 0x40E0_1000

The registers are listed below:

Table 25-4: List of CAN Registers

Offset Address	Name	Description	
0x00	CONFIG0	0x00	CAN_MR: mode register
		0x01	CAN_CMR: command register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description	
		0x02	CAN_SR: status register
		0x03	CAN_ISR: interrupt status register
0x04	CONFIG1	0x04	CAN_IMR: interrupt mask register
		0x05	CAN_RMC: Receive data count register
		0x06	CAN_BTR0: bus timing register 0
		0x07	CAN_BTR1: bus timing register 1
0x08	CAN_TXBUF	TX buffer register	
0x0C	CAN_RXBUF	RX buffer register	
0x10	CAN_ACR	RX filter match register	
0x14	CAN_AMR	RX filter mask register	
0x18	ERRCR	0x18	CAN_ECC: error code capture register
		0x19	CAN_RXERR: Receive error count register
		0x1A	CAN_TXERR: Transmit error count register
		0x1B	CAN_ALC: arbitration lost capture register

25.5.1 Mode Register (CAN_MR (CONFIG0[7:0]))

Offset address: 0x00

Reset value: 0x04

Bit	Name	Attribute	Reset Value	Description
7	RXF_CLR	W	0	RX_FIFO pointer clear bit: 1: Reset the read-write pointer of RX_FIFO, after which this bit will automatically return to 0. 0: No change
6:3	RSV	-	-	Reserved
2	RM	R/W	1	Reset mode set bit: 1: CAN works in reset mode. 0: CAN works in other modes. No data transmission and reception are performed in reset mode, which is used for some hardware configurations (e.g. some registers can only be written in reset mode). After reset mode, it can enter listening mode or normal mode.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	LOM	R/W	0	Listening mode set bit: 1: If RM = 0, CAN enters listening mode*. 0: If RM = 0, CAN enters normal mode. This bit can only be set in reset mode.
0	AFM	R/W	0	Hardware matching data selection bit: 1: Use single filter 0: Use double filters This bit can only be set in reset mode.

Note*: In listening mode, the CAN controller will not answer to the CAN bus (no ACK response will be sent) even if the message is successfully received. The error counter will stop at the current value. Listening mode is mainly used for bit rate detection without interfering with network traffic, and can also be used for the CAN bus analyzer.

25.5.2 Command Register (CAN_CMR (CONFIG0[15: 8]))

Offset address: 0x01

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7:3	RSV	-	-	Reserved
2	TR	W	0	Transmit request setting bit: 1: Frame transmission enabled 0: Transmission disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	AT	W	0	Enable bit for aborting transmission: 1: Transmission aborting enabled 0: Transmission aborting disabled Setting TR and AT at the same time can initiate a single-transmission, and frame retransmission will not be performed in the case of bus error or arbitration lost. Aborting only works for frames that are about to be transmitted, frames that have already been sent cannot be aborted. If the transmission is started by setting TR to 1 in the previous command, it cannot be canceled by setting TR bit to 0. In this case, the transmission can be canceled by setting AT to 1.
0	RSV	-	-	Reserved

25.5.3 Status Register (CAN_SR (CONFIG0[23:16]))

Offset address: 0x02

Reset value: 0x20

Bit	Name	Attribute	Reset Value	Description
7	RBS	R	0	RX FIFIO status: 1: At least one message in FIFO 0: No message in FIFO
6	DSO	R	0	Data overflow status: 1: RX FIFO overflow triggers an interrupt (if enabled). 0: No overflow has occurred since the last data overflow clearing.
5	TBS	R	1	TX buffer status: 1: TX buffer can be written by CPU. 0: TX buffer is locked. A message is being sent or waiting to be sent. If CPU tries to write to the TX buffer in the locked state (TBS = 0), the written data is not accepted.
4	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	RS	R	0	RX status bit: 1: CAN is receiving. 0: CAN is not in receive state.
2	TS	R	0	TX status bit: 1: CAN is transmitting. 0: CAN is not in transmit state.
1	ES	R	0	Error status bit: 1: At least one CAN error counter reaches the error warning limit of 96. 0: Normal status
0	BS	R	0	Bus status bit: 1: Off-line state The CAN controller is in reset mode and the error warning interrupt is triggered (if enabled). The transmit error counter is set to 127, and the receive error counter is set to 0. CAN will remain in reset mode until CPU clears the RM bit. After this operation, CAN will wait for the occurrence of 128 bus idle signals (11 consecutive recessive bits), and the transmit error counter will count down. Then the BS bit is cleared, the error counter is reset, and the error warning interrupt is triggered (if enabled). 0: Normal state; frame transmission and reception can be performed.

25.5.4 Interrupt Status Register (CAN_ISR (CONFIG0[31:24]))

Offset address: 0x03

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7	RSV	-	-	Reserved
6	ALI	R/W	0	Arbitration lost interrupt status bit: When an arbitration lost is detected during message transmission and CAN becomes the receiver, this bit can be set to read ALC register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				to check which bit in the arbitration segment is lost, and writing 1 clears this interrupt.
5	EWI	R/W	0	Error warning interrupt status bit: This error warning interrupt bit is set when the ES or BS bit in SR register changes. Therefore, it can be used to detect if CAN enters or exits the bus-off state. Write 1 to clear this interrupt.
4	EPI	R/W	0	Error passive interrupt status bit: This bit is set when the CAN bus controller reaches or exits the error passive level (i.e., the status changes from active to passive or vice versa). Write 1 to clear this interrupt.
3	RI	R/W	0	RX interrupt status bit: CAN sets this bit to 1 when there is at least one CAN frame data in RX FIFO. After reading the message, CPU must write the RI bit to 1 (message read acknowledgment) to decrement the count of RX message counter (RMC), which does not automatically decrement.
2	TI	R/W	0	TX interrupt status bit: The TX interrupt bit is set upon successful transmission. The write pointer can be reset to TX RAM by clearing the TI bit (via writing 1) before writing a new data frame.
1	BEI	R/W	0	Bus error interrupt status bit: Set BEI when CAN encounters a bus error in the course of sending or receiving messages. Write 1 to clear this interrupt.
0	DOI	R/W	0	RX data overflow interrupt status bit: DOI is set when an RX FIFO overflow occurs. Write 1 to clear this interrupt.

25.5.5 Interrupt Mask Register (CAN_IMR (CONFIG1[7:0]))

Offset address: 0x04

Reset value: 0x00

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7	RSV	-	-	Reserved
6	ALIM	R/W	0	Arbitration lost interrupt enable bit: enable the CAN transmitter to trigger an interrupt when an arbitration lost is detected during transmission and it becomes a CAN receiver: 1: ALI enabled 0: ALI disabled
5	EWIM	R/W	0	Error warning interrupt enable bit: enable to trigger an interrupt when the status of the BS or ES bit of CAN_SR register changes: 1: EWI enabled 0: EWI disabled
4	EPIM	R/W	0	Error passive interrupt enable bit: enable to trigger an interrupt when the CAN controller enters or exits error passive mode: 1: EPI enabled 0: EPI disabled
3	RIM	R/W	0	RX interrupt enable bit: 1: RI enabled 0: RI disabled
2	TIM	R/W	0	TX interrupt enable bit: 1: TI enabled 0: TI disabled
1	BEIM	R/W	0	Bus error interrupt enable bit: enable to trigger an interrupt when a bus error occurs during CAN transmission or reception: 1: BEI enabled 0: BEI disabled
0	DOIM	R/W	0	RX data overflow interrupt enable bit: 1: DoI enabled 0: DoI disabled

25.5.6 RX Data Bit Count Register (CAN_RMC (CONFIG1[15:8]))

Offset address: 0x05

Reset value: 0x00

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:5	RSV	-	-	Reserved
4:0	RMC	R	0	Number of CAN frames in RX FIFO, which can store up to 32 standard ID messages (or 16 extended ID messages). The following equation allows calculating the maximum number of messages to be stored in RX FIFO: $n = \frac{64}{3 + data_length_code}$ Note: Here data_length_code is at least 1. If the length of CAN data segment is 0, then data_length_code = 1.

25.5.7 Bus Timing Register (CAN_BTR0 (CONFIG1[23:16]))

This register can only be written in reset mode and can be read in any mode.

Offset address: 0x06

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7:6	SJW	R/W	0	Synchronization jump width: $t_{SJW} = t_{SCLK} \times (2 \times SJW.1 + SJW.0 + 1)$ In order to compensate the phase shift between clock oscillators of different CAN bus controllers, the bit period must be shortened or extended accordingly. SJW defines the maximum number of clock cycles for a resynchronization to change a bit period. During resynchronization, the hardware will synchronize with the received signal by adding 1+SJW t_{SCLK} periods within the PBS1 segment, or by subtracting 1 to (1+SJW) t_{SCLK} periods within the PBS2 segment.
5:0	BRP	R/W	0	Baud rate prescaler value: $t_{SCLK} = 2 \times t_{CLK} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$ Wherein, $t_{CLK} = 1 / f_{PCLK}$.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.5.8 Bus Timing Register (CAN_BTR1 (CONFIG1[31:24]))

This register can only be written in reset mode and can be read in any mode.

Offset address: 0x07

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7	SAM	R/W	0	Bus level sampling number selection bit: 1: Sample the bus level three times (for medium/low-speed buses) 0: Sample the bus level once (for high-speed bus)
6:4	TSEG2	R/W	0	Number of clock cycles for time segment 2: $t_{TSEG2} = t_{SCLK} \times (4 \times TSEG2.2 + 2 \times TSEG2.1 + TSEG2.0 + 1)$
3:0	TSEG1	R/W	0	Number of clock cycles for time segment 1: $t_{TSEG1} = t_{SCLK} \times (8 \times TSEG1.3 + 4 \times TSEG1.2 + 2 \times TSEG1.1 + TSEG1.0 + 1)$

The bit period structure of CAN is shown in the following diagram. Wherein, the synchronization segment (SYNC SEG) is 1 x t_{SCLK}, and the lengths of phase buffer segments 1 and 2 are determined by TSEG1 and TSEG2.

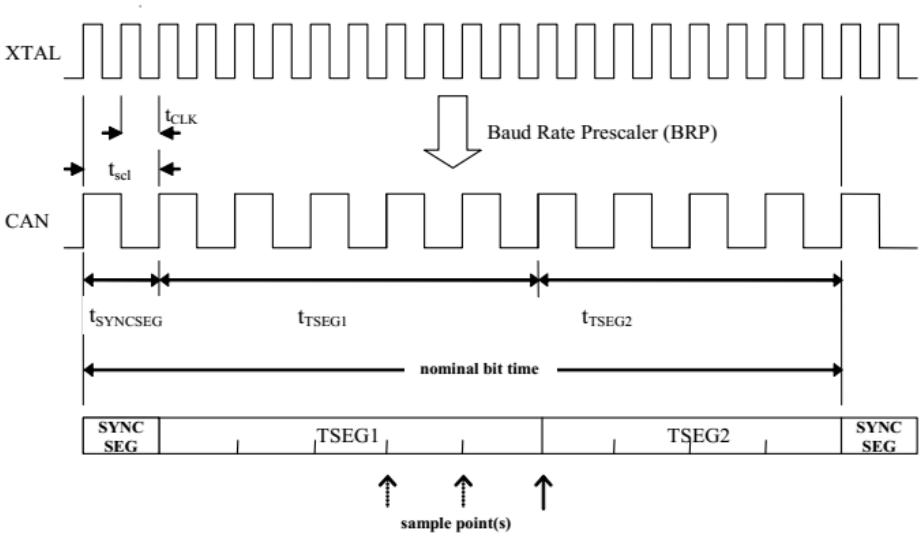

Figure 25-1: Bit Period Structure of CAN

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.5.9 Transmit Buffer Register (CAN_TXBUF)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	TXBUF	W	0	The transmit buffer register is used to write the CAN frames to be transmitted over the CAN network. Writing to this register performs automatic increment of the internal write pointer. By writing TI bit in ISR register, the write pointer can be reset to the address 0h of the transmit memory.

25.5.10 Receive Buffer Register (CAN_RXBUF)

Offset address: 0x0C

Reset value: uncertain

Bit	Name	Attribute	Reset Value	Description
31:0	RXBUF	R	Uncertain	The receive buffer register is used to read CAN frames received from the CAN network. Reading this register will automatically increment the read address pointer of the internal FIFO (increment after reading).

After receiving a frame of CAN data, the format of data read by RXBUF register is as follows (the length of databuf segment is determined by DLC segment, and the number is 0-8 bytes):

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

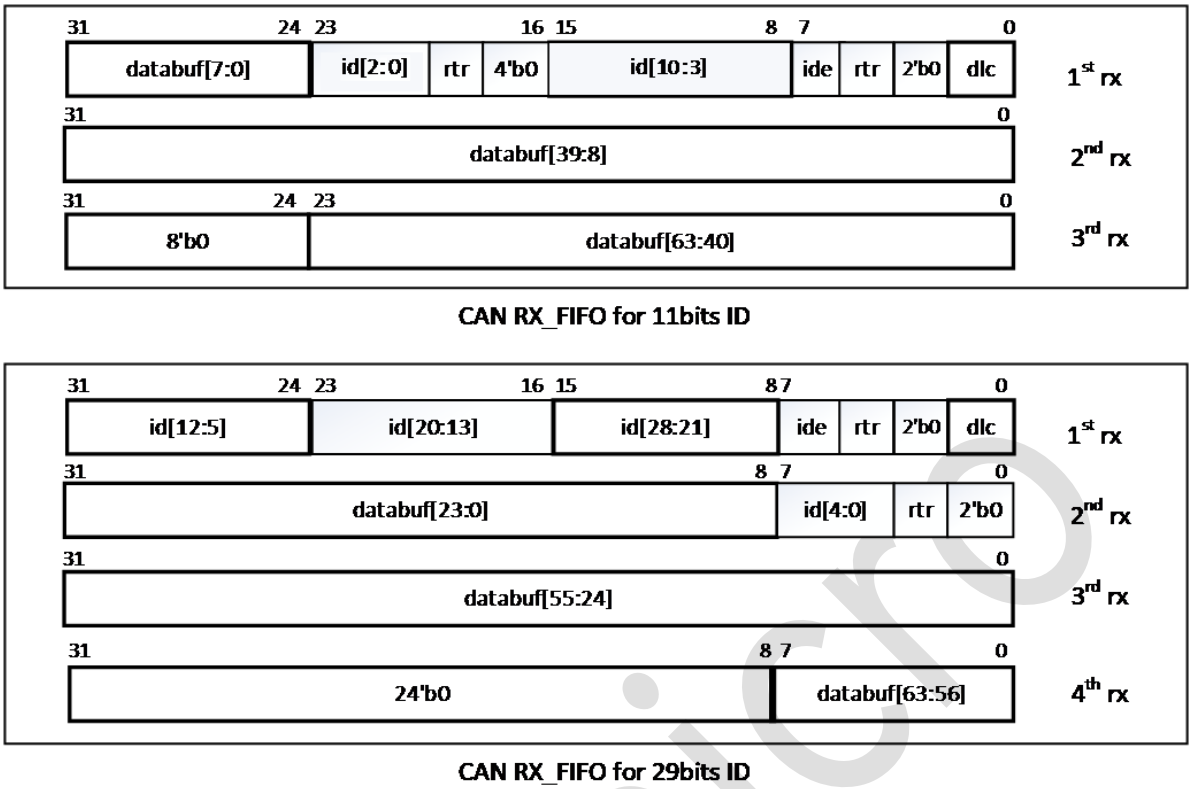

Figure 25-2: RXBUF Register Data Format

After receiving a frame of CAN data, the RMC register count is increased by 1, and the CAN controller will write data into RX FIFO one by one. RBS will be set when a 32-bit data is written. After a frame of data is written, the RI flag bit is set.

25.5.11 RX Filter Match Register (CAN_ACR)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ACR3-0	R/W	0	The RX filter match register contains the arbitration bit of the message to be received, and the corresponding RX filter mask register defines the bits to be compared and the irrelevant bits.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

25.5.12 RX Filter Mask Register (CAN_AMR)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	AMR3-0	R/W	0	The RX filter mask register defines the bits to be compared and the irrelevant bits. Set the corresponding bit to 1 indicating not to compare the corresponding bits in ACR register.

Only when the identifier bit of the received message is equal to the predefined bit in the RX filter can the RX filter in CAN controller pass the received message to RX FIFO. The RX filter consists of a RX filter match register (ACR3: ACR0) and a RX filter mask register (AMR3: AMR0). The AFM bit of the mode register can set single/double filters.

The bit formats corresponding to different filter settings and different arbitration lengths (11 bits in standard frame / 29 bits in extended frame) are as follows:

- In single filter configuration, the filter is 4 bytes long.

  If the received data is in standard frame mode, the first two bytes (including the arbitration bit, RTR bit, and data bits) can be received (the data byte is not necessarily received). All individual bit comparisons must be signaled to indicate successful reception of the data.

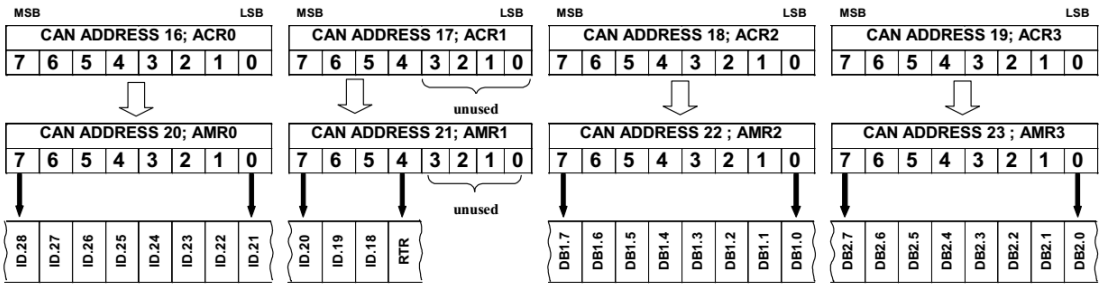

Figure 25-3: Bit Format Corresponding to Single Filter in Standard Frame

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If the received data is in extended frame format, the arbitration bit and RTR bit will be received. For bits not defined in the format, the filter will not perform comparisons.

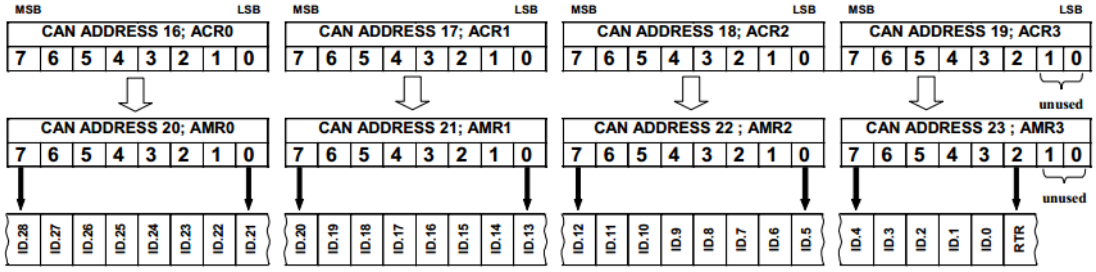

Figure 25-4: Bit Format Corresponding to Single Filter in Extended Frame

- In dual-filter configuration, two shorter filters are defined. The received data will be compared against both filters to decide whether the data shall be stored into RX FIFO. If at least one of the RX filters successfully matches, the received data will be stored in FIFO.

  If the received data is in standard frame format, after receiving the data, it will be compared with the ID (including the RTR bit) and the first received data byte of the first filter, or with the ID (including the RTR bit) of the second filter.

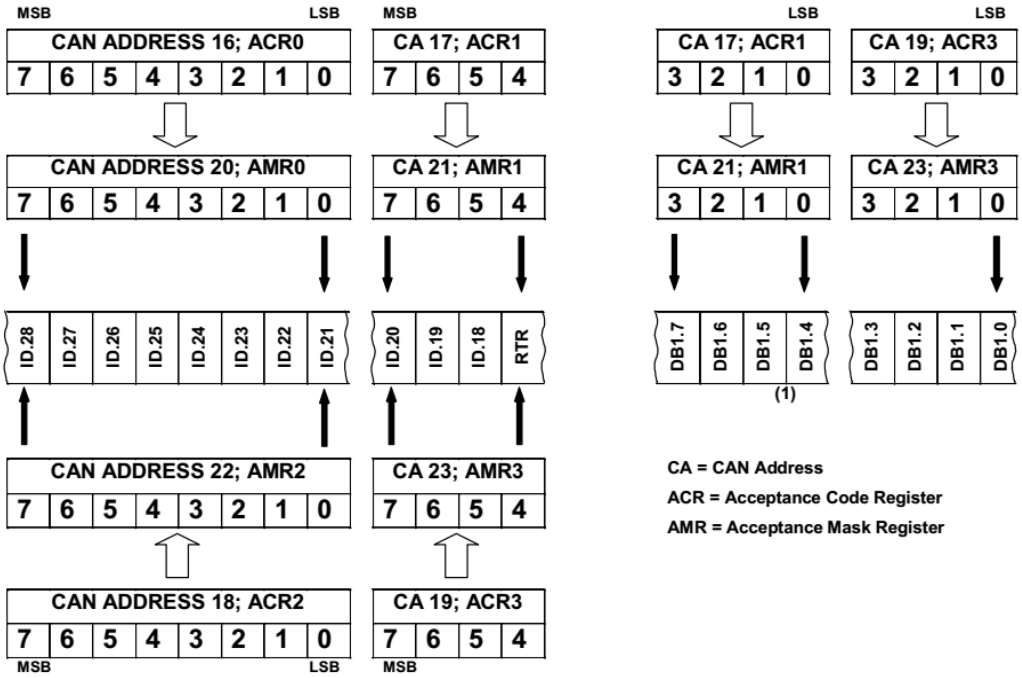

Figure 25-5: Bit Format Corresponding to Double Filters in Standard Frame

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If the received data is in extended frame format, the two filters will only compare the first two bytes of the extended identifier range. For successful reception, all bits of at least one filter must be compared.

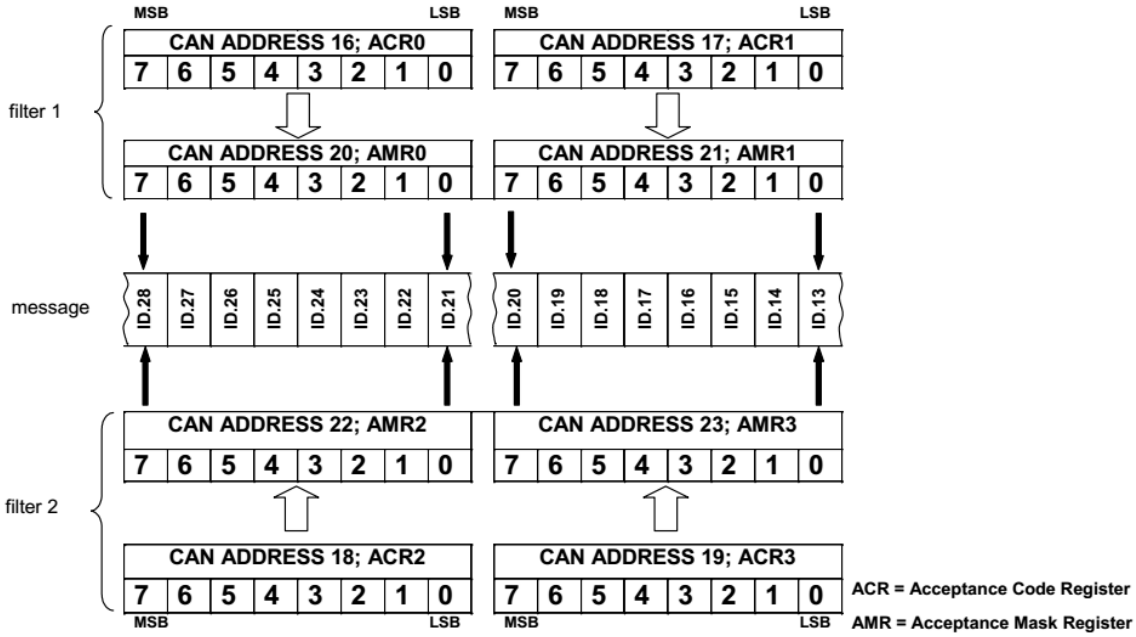

Figure 25-6: Bit Format Corresponding to Double Filters in Extended Frame

25.5.13 Error Code Capture Register (CAN_ECC (ERRCR [7:0]))

The ECC read-only register holds the error code regarding the last bus error that occurred on the CAN network. This register is read-only.

The CAN core will not update this register until the previous bus error is acknowledged (by acknowledging the bus error interrupt).

Offset address: 0x18

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7	RXWRN	R	0	Set this bit to 1 when the RXERR counter value is greater than or equal to 96.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	TXWRN	R	0	Set this bit to 1 when the TXERR counter value is greater than or equal to 96.
5	EDIR	R	0	Direction of data transmission at the occurrence of an error: 0: Transmitting 1: Receiving
4	ACKER	R	0	Set this bit to 1 when an ACK error occurs.
3	FRMER	R	0	Set this bit to 1 when a frame format error occurs.
2	CRCER	R	0	Set this bit to 1 when a CRC error occurs.
1	STFER	R	0	Set this bit to 1 when a padding error occurs.
0	BER	R	0	Set this bit to 1 when a bit error occurs.

25.5.14 Receive Error Count Register (CAN_RXERR (ERRCR [15:8]))

Offset address: 0x19

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7:0	RXERR	R	0	Current value of the receive error counter. If a bus shutdown event occurs, the receive error counter will be initialized to 0.

25.5.15 Transmit Error Count Register (CAN_TXERR (ERRCR [23:16]))

Offset address: 0x1A

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7:0	TXERR	R	0	Lower 8 bits of the current value of transmit error counter. If a bus-off event occurs, the transmit error counter will be initialized to 127 to calculate the minimum protocol-defined time (128 occurrences of the bus

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				idle signal). Reading TXERR during this time can obtain information about the bus-off recovery state.

25.5.16 Arbitration Lost Capture Register (CAN_ALC (ERRCR [31:24]))

The CAN controller is able to determine the exact in-frame location of the arbitration lost. Immediately thereafter an “arbitration lost interrupt” will be generated. In addition, the number of bits is captured in the arbitration lost capture register. Once the master controller reads the contents of this register, the capture function will be activated for the next arbitration lost. This feature allows CAN to monitor each CAN bus access. For diagnostics or during system configuration, each case of unsuccessful arbitration can be determined.

Offset address: 0x1B

Reset value: 0x00

Bit	Name	Attribute	Reset Value	Description
7:5	RSV	-	-	Reserved
4:0	ALC	R	0	Arbitration lost position

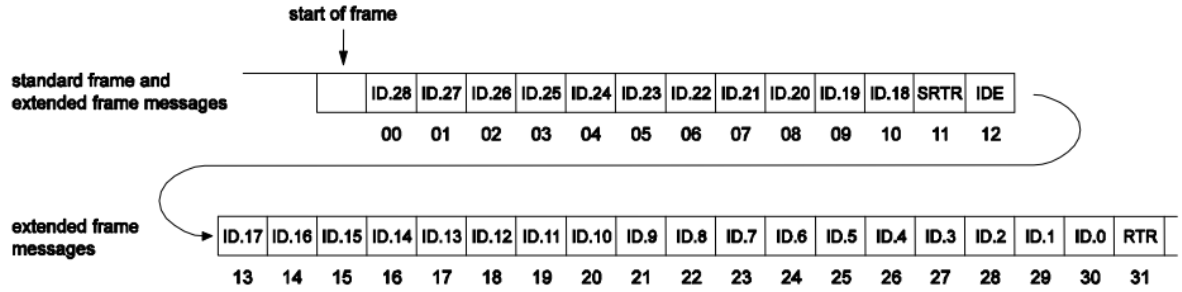

Figure 25-7: Arbitration Lost Bit Interrupt Diagram

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Table 25-5: ALC Register Description

Bits					Decimal Value	Description
ALC4	ALC3	ALC2	ALC1	ALC0		
0	0	0	0	0	00	Arbitration lost in **ID28 / 10**
0	0	0	0	1	01	Arbitration lost in **ID27 / 9**
0	0	0	1	0	02	Arbitration lost in **ID26 / 8**
0	0	0	1	1	03	Arbitration lost in **ID25 / 7**
0	0	1	0	0	04	Arbitration lost in **ID24 / 6**
0	0	1	0	1	05	Arbitration lost in **ID23 / 5**
0	0	1	1	0	06	Arbitration lost in **ID22 / 4**
0	0	1	1	1	07	Arbitration lost in **ID21 / 3**
0	1	0	0	0	08	Arbitration lost in **ID20 / 2**
0	1	0	0	1	09	Arbitration lost in **ID19 / 1**
0	1	0	1	0	10	Arbitration lost in **ID18 / 0**
0	1	0	1	1	11	Arbitration lost in **SRTR / RTR**
0	1	1	0	0	12	Arbitration lost in **IDE** bit
0	1	1	0	1	13	Arbitration lost in **ID17***
0	1	1	1	0	14	Arbitration lost in **ID16***
0	1	1	1	1	15	Arbitration lost in **ID15***
1	0	0	0	0	16	Arbitration lost in **ID14***
1	0	0	0	1	17	Arbitration lost in **ID13***
1	0	0	1	0	18	Arbitration lost in **ID12***
1	0	0	1	1	19	Arbitration lost in **ID11***
1	0	1	0	0	20	Arbitration lost in **ID10***
1	0	1	0	1	21	Arbitration lost in **ID9***
1	0	1	1	0	22	Arbitration lost in **ID8***
1	0	1	1	1	23	Arbitration lost in **ID7***
1	1	0	0	0	24	Arbitration lost in **ID6***
1	1	0	0	1	25	Arbitration lost in **ID5***
1	1	0	1	0	26	Arbitration lost in **ID4***
1	1	0	1	1	27	Arbitration lost in **ID3***
1	1	1	0	0	28	Arbitration lost in **ID2***
1	1	1	0	1	29	Arbitration lost in **ID1***
1	1	1	1	0	30	Arbitration lost in **ID0***
1	1	1	1	1	31	Arbitration lost in **RTR**

25.6 Operation Procedure

25.6.1 Transmitting CAN Data Frame

1. Turn on the CAN clock, release the reset, and multiplex the CAN function pins.
2. Configure the bus timing registers CAN_BTR0/CAN_BTR1.
3. Clear the error flag bit/interrupt flag bit in CAN_ISR register.
4. Configure the interrupt enable register CAN_IMR to enable TI interrupt (optional).
5. Configure the mode register CAN_MR[1] to enter the normal mode.
6. Configure the TX buffer register CAN_TXBUF, write the contents of CAN data frames according to the defined format in sequence, and write 32 bits of data at a time.
7. Configure the command register CAN_CMR[2] to start transmitting.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8. Wait for the status register CAN_SR[5] being set to 1 (if TI interrupt is enabled, here the trigger of TI interrupt can be waited), then the data is transmitted.

25.6.2 Receiving CAN Data Frame

1. Turn on the CAN clock, release the reset, and multiplex the CAN function pins.
2. Configure the bus timing registers CAN_BTR0/CAN_BTR1.
3. Clear the error flag bit/interrupt flag bit in CAN_ISR register.
4. Configure the interrupt enable register CAN_IMR to enable RI interrupt (optional).
5. Configure the RX filter and set CAN_MR[0] to 1 if a single filter is used. The CAN_ACR register configures what the user needs to filter, and the CAN_AMR register selects the bits that need to be compared with those of the CAN_ACR register. If no comparison is required, all bits of the CAN_AMR register shall be set to 1.
6. Configure the mode register CAN_MR[1] to enter the normal mode.
7. Wait for the status register CAN_SR[7] being set to 1 (if RI interrupt is enabled, here the trigger of RI interrupt can be waited), read the data in RX buffer register CAN_RXBUF several times until all the data are retrieved.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26 Inter-integrated Circuit (I2C) Interface

26.1 Overview

The purpose of the I2C module is to facilitate the read and write operations by CPU to the slave devices connected on the I2C bus. When the CPU performs a write operation to the slave device, it configures the configuration registers of the I2C module through the bus, then sends control information and operands to the data communication registers of the I2C module. After parsing the command, the I2C module transmits the data from its data channel registers to the slave device via the I2C bus. Once the transmission is completed, the final status is fed back to the CPU via an interrupt. The process for the CPU to read data from the slave device is similar to that of the write operation.

26.2 Main Features

- Dual-line I2C serial interface
- Standard mode (100 Kb/s), fast mode (400 Kb/s) and fast mode+ (1 Mb/s)
- Master or slave mode
- 7-bit or 10-bit addressing mode
- Transmission in 7-bit or 10-bit combined addressing modes
- Bulk transfer mode
- 8-byte transmit and receive buffers

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Interrupt and polling operations
- Bit and byte waiting at all speeds
- Programmable SDA hold time
- Bus clearing
- DMA operation
- SMBus (system management bus) / PMBus (power management bus)
- SMBus slave detection and response to ARP commands
- Address resolution protocol (ARP)

26.3 Pin Description

Table 26-1: I2C Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
I2C1_SDA	PB5, PB7, PB9	Input/output	Data
I2C1_SCL	PB6, PB8	Input/output	Clock
I2C2_SDA	PB11, PB12, PB15	Input/output	Data
I2C2_SCL	PB10, PB14	Input/output	Clock
I2C3_SDA	PA9, PC9, PD2	Input/output	Data
I2C3_SCL	PA8, PC12	Input/output	Clock

26.4 Functional Description

26.4.1 SMBus / PMBus

SMBus is used to provide predictable communication lines between the system and its devices, describing the device timeout definition and conditions thereof.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.4.1.1 Bus Protocol

A typical SMBus features a set of commands by which data can be read and written. All commands are one byte in length, but the length of their parameters and return values can vary. According to the SMBus specification, the most significant bit (MSB) is transmitted first. For a given device, there are 11 command protocols, including: Quick Command, Send Byte, Receive Byte, Write Byte, Write Word, Read Byte, Read Word, Process Call, Block Read, Block Write, and Block Write-Block Read Process Call.

The SMBus protocol for message transactions is usually different from the I2C data transfer commands, but it is still possible to program the SMBus master for I2C data transfer. The following table describes the SMBus protocol generated by the TX FIFO command of I2C.

In SMBus master mode, all received data is available in RX FIFO. In SMBus slave mode, all bus protocol command codes and data bytes will be placed in RX FIFO, and the data bytes of read request must be sent using TX FIFO, similar to I2C mode.

Table 26-2: SMBus Protocol Usage Table

Protocol	Required TXFIFO Commands	DATA (I2C_DATA_CMD[7:0])	CMD (I2C_DATA_CMD[8])	STOP (I2C_DATA_CMD[9])	Remarks
Quick Command	1	Not applicable	Set R/W.	Set to 1.	Set bit 11 and bit 16 of I2C_TAR to 1.
Send Byte	1	Data byte	Set to 0.	Set to 1.	
Receive Byte	1	Not applicable	Set to 1.	Set to 1.	
Write Byte	2	Command code	Set to 0.	Set to 0.	
		Data byte	Set to 0.	Set to 1.	
Write Word	3	Command code	Set to 0.	Set to 0.	
		Data byte low	Set to 0.	Set to 0.	
		Data byte high	Set to 0.	Set to 1.	

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Protocol	Required TXFIFO Commands	DATA (I2C_DATA_CMD[7:0])	CMD (I2C_DATA_CMD[8])	STOP (I2C_DATA_CMD[9])	Remarks
Read Byte	2	Command code	Set to 0.	Set to 0.	
		Not applicable	Set to 1.	Set to 1.	
Read Word	3	Command code	Set to 0.	Set to 0.	
		Not applicable	Set to 1.	Set to 0.	
		Not applicable	Set to 1.	Set to 1.	
Process Call	5	Command code	Set to 0.	Set to 0.	
		Data byte low	Set to 0.	Set to 0.	
		Data byte high	Set to 0.	Set to 0.	
		Not applicable	Set to 1.	Set to 0.	
		Not applicable	Set to 1.	Set to 1.	
Block Write	N + 1	Command code	Set to 0.	Set to 0.	
		Data byte	Set to 0.	Set to 0.	
		N + 1) Data byte N	Set to 0.	Set to 1.	
Block Read	N+1	Command code	Set to 0.	Set to 0.	
		Not applicable	Set to 0.	Set to 0.	
		N + 1) Not applicable	Set to 0.	Set to 1.	
Block Write-Block Read Process Call	M + N + 1	Command code	Set to 0.	Set to 0.	
		Data byte 1	Set to 0.	Set to 0.	
		M + 1) Data byte M	Set to 0.	Set to 0.	
		M + 2) Not applicable	Set to 1.	Set to 0.	
		M + 3) Not applicable	Set to 1.	Set to 0.	
		M + N + 1) Not applicable	Set to 1.	Set to 1.	

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Protocol	Required TXFIFO Commands	DATA (I2C_DATA_CMD[7:0])	CMD (I2C_DATA_CMD[8])	STOP (I2C_DATA_CMD[9])	Remarks
SMBUS Host Notify Protocol	3	Data byte low	Set to 0.	Set to 0.	I2C_TAR is set to SMBus master address (0001 000).

The I2C slave receives via the Quick command only if SMBUS_SLAVE_QUICK_CMD_EN of I2C_CR is enabled. Whenever this bit is selected, the slave receives only the Quick command and no other bus protocols. The I2C slave issues a SMBUS_QUICK_DET interrupt upon receipt of the Quick command.

SMBus introduces a packet error checking mechanism by appending a PEC byte to the end of the bus protocol. This can be accomplished by adding additional commands (PEC bytes) while transmitting and by decoding them in software when receiving.

26.4.1.2 SMBus Address Resolution Protocol

SMBus slave address conflicts can be resolved by the master dynamically assigning a new unique address to each slave device. This feature allows devices to be “hot-swapped”.

SMBus introduces a 128-bit unique device ID (UDID) for each device in the system to isolate each device for address assignment. The high 96 bits of the UDID are set to 0, while the low 32 bits are controlled by the I2C_SMBUS_ARP_UDID_LSB register.

The SMUBS_PERSISTANT_SLV_ADDR_EN bit in the IC_CR register is used to indicate whether I2C supports persistent slave addresses.

The I2C master can issue generic and directed address resolution protocol (ARP) commands to assign dynamic addresses to slaves in the SMBus system.

Table 26-3: Table of SMBus ARP Commands Derived via TxFIFO Commands in I2C

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

ARP Command	Required TXFIFO Commands	Command/Data (I2C_DATA_CMD [7:0])	CMD Bit (I2C_DATA_CMD[8])	STOP Bit (I2C_DATA_CMD[9])	Remarks
Prepare for ARP	2	Command = '0000 0001'	Set to 0.	Set to 0.	I2C_TAR[6:0] is set to SMBus default address (1100 001).
		PEC byte	Set to 0.	Set to 1.	
Reset device (general)	2	Command = '0000 0010'	Set to 0.	Set to 0.	I2C_TAR[6:0] is set to SMBus default address (1100 001).
		PEC byte	Set to 0.	Set to 1.	
Get UDID (general)	20	Command = '0000 0011'	Set to 0.	Set to 0.	1. I2C_TAR[6:0] is set to SMBus default address (1100 001). 2. Perform 16 reads of the 128 UDID bytes. 3. The last read command accesses the slave address.
		Not applicable	Set to 1.	Set to 0.	
		Not applicable	Set to 1.	Set to 0.	
		Not applicable	Set to 1.	Set to 0.	
		PEC byte	Set to 1.	Set to 1.	
Assign address	20	Command = '0000 0100'	Set to 0.	Set to 0.	1. I2C_TAR[6:0] is set to SMBus default address (1100 001). 2. Perform 16 writes of the 128 UDID bytes. 3. The last write command accesses the slave address.
		Byte count = 17	Set to 0.	Set to 0.	-
		UDID byte 15	Set to 0.	Set to 0.	
		UDID byte 14	Set to 0.	Set to 0.	
		Assigned address	Set to 0.	Set to 0.	
		PEC byte	Set to 01.	Set to 1.	
Get UDID (directed)	19	Command = '0000 0011'	Set to 0.	Set to 0.	1. I2C_TAR[6:0] is set to SMBus default address (1100 001).
		{Slave address[6:0],1}	Set to 1.	Set to 0.	

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

ARP Command	Required TXFIFO Commands	Command/Data (I2C_DATA_CMD [7:0])	CMD Bit (I2C_DATA_CMD[8])	STOP Bit (I2C_DATA_CMD[9])	Remarks
		Not applicable	Set to 1.	Set to 0.	2. Perform 16 reads of the 128 UDID bytes. 3. The last read command accesses the slave address.
		Not applicable	Set to 1.	Set to 0.	
		PEC byte	Set to 1.	Set to 1.	
Reset device (directed)	2	Command = {slave address[6:0],0}	Set to 0.	Set to 0.	I2C_TAR[6:0] is set to SMBus default address (1100 001).
		PEC byte	Set to 0.	Set to 1.	

26.4.1.3 Performing ARP in Master Mode

When I2C is used as a SMBus master, the following steps shall be performed to assign a unique address to each slave device in order to resolve address conflicts:

1. After a reset or cold start, the SMBus master issues a "Prepare for ARP" command, indicating that the master will perform ARP to allocate dynamic addresses for all devices. The slave must refresh any pending master notification commands.
2. Upon receiving an ACK for the "Prepare to ARP" command, it indicates that there are devices in the system that support ARP. The master then issues a "Get UDID" command. A NACK signifies that either there are no devices supporting ARP or that all slaves currently have resolved addresses. In this case, the master must complete the operations outlined in step 8.
3. The I2C master issues a "Get UDID" command to receive the UDID information from the slave for dynamic address allocation.
4. If the first three bytes of the "Get UDID" command are ACKed and the received byte count is 0x11, the master issues an "Assign Address" command. Otherwise, the master must

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

complete the operations described in step 8 to indicate that ARP has been completed.

5. The master issues an “Assign Address” command to dynamically assign an address to the slave corresponding to the UDID received via the “Get UDID” command.
6. If the assigned address packet is ACKed, the master will remove the assigned address from the address pool and jump to step 3 to obtain the UDID of another slave device. If the address packet is NACKed, the master will not remove this address from the address pool and will then jump to step 3 to obtain the UDID of the same slave or another slave.
7. If the assigned address packet is ACKed, the master will store the assigned address along with the UDID feature of the device in the used address pool.
8. The master jumps to step 3 to issue the “Get UDID” command again to receive the UDIDs of other slaves. If a NACK is received for the “Get UDID,” the master jumps to step 9.
9. The I2C can switch to slave mode to detect requests from devices for master notification protocols.
10. If the I2C switches to slave mode and detects the master notification protocol, indicating that the slave is requesting a dynamic address, the master must perform ARP as described in step 11.
11. If in master mode, jump to step 3 to perform ARP; otherwise, jump to step 12.
12. The I2C switches to master mode and jumps to step 3 to perform ARP.

26.4.1.4 Performing ARP in Slave Mode

As a SMBus slave, the I2C performs the following tasks:

- Decode ARP commands and respond based on the internal status flags: SMBUS_SLAVE_ADDR_VALID and SMBUS_SLAVE_ADDR_RESOLVED of the I2C register.
- Generate and verify the PEC byte for ARP commands.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- An ACK will be generated for the PEC byte only if it matches the CRC value calculated from the received data. If not, NACK the PEC byte.

When another SMBus master on the bus issues an ARP command and the I2C device has participated in ARP, the I2C as a slave performs the following operations:

1. After a reset or cold start, the I2C slave checks if it supports a persistent slave address (PSA).
2. If the I2C has a PSA, it is indicated by the set address valid flag and set in the slave address register (I2C_SAR). If the flag is not set, proceed to step 4.
3. The I2C persistent slave stores the persistent address in I2C_SAR and sets the address valid flag to 1, with the address resolved flag set to 0.
4. The I2C non-persistent slave (non-PSA) clears the address valid and address resolved flags.
5. The I2C checks the slave address field in the received packet for the ARP default address to determine if it is an ARP command or a normal command. If it matches, proceed to step 6; otherwise, go to step 25.
6. If the I2C detects a packet sent to the SMBus device default address, it checks the command field to determine if it is the “Prepare to ARP” command. If it is, proceed to step 7; otherwise, go to step 8.
7. Upon receiving the “Prepare to ARP” command, the I2C acknowledges the packet and clears the address resolved flag to participate in the ARP process. The I2C then continues with step 5 and waits for another SMBus packet.
8. The I2C checks the command field to verify if the “Reset Device” command has been issued. If it is, proceed to step 9; otherwise, go to step 10.
9. Upon receiving the “Reset Device” command, the I2C acknowledges the packet and clears the address and address valid flags (if it is non-PSA, I2C_CR[19]=0). The I2C then proceeds to step 5 and waits for another SMBus packet.
10. The device checks the command to verify if the “Assign Address” command has been issued.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If it is, proceed to step 11; otherwise, go to step 13.

11. Upon receiving the "Assign Address" command, the I2C compares its UDID with the received bytes. If any byte does not match, the I2C will not acknowledge that byte and subsequent bytes. If all bytes in the UDID match, the device proceeds to step 12; otherwise, it goes to step 5 and waits for another SMBus packet.
12. After matching the UDID in step 11, DW_apb_i2c receives the slave address and uses it to set the I2C_SAR register. The I2C sets its address valid and address resolved flags, indicating it has received a dynamic address and will no longer respond to the "Get UDID" command unless it receives a "Prepare to ARP" or "Reset Device" command. The I2C now proceeds to step 5 and waits for another SMBus packet.
13. The I2C checks the command field to verify if the "Get UDID" command has been issued. If it is, proceed to step 14; otherwise, go to step 19.
14. Upon receiving the "Get UDID" command, the I2C checks its address resolved flag to determine if it must participate in the ARP process. If it is set, its address has been resolved by the ARP master, so the device proceeds to step 5 and waits for another SMBus packet. If the ARP flag has been cleared, proceed to step 15.
15. The I2C returns its UDID and monitors the SMBus data line for conflicts. If a conflict is detected at any time, the I2C sets the SLV_ARB_LOST bit and stops the transmission. It then continues with step 5 and waits for another SMBus packet. If no conflict is detected, the I2C proceeds to step 16.
16. The I2C checks its address valid (AV) flag to determine the value to return to the device slave address field. If the AV flag is set, proceed to step 17; otherwise, go to step 18.
17. When the AV flag is set, the current I2C_SAR is valid, so the device returns it to the device slave address field while monitoring the SMBus data line for conflicts. The I2C continues with step 5 and waits for another SMBus packet.
18. If the AV flag is not set, the current slave address (I2C_SAR) is invalid. The I2C returns a

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

value of FFh and monitors the SMBus data line for conflicts. If the ARP master receives a value of FFh, the device requires address allocation. The I2C then continues with step 5 and waits for another SMBus packet.

19. The I2C may be receiving direct commands. If the address valid flag is set and the address matches the one in I2C_SAR, proceed to step 20; otherwise, continue with step 5 to wait for another SMBus packet.
20. If the address valid flag is set, check if the command is a “Reset Device” direct command. If it is, proceed to step 21; otherwise, go to step 22.
21. Upon receiving the “Reset Device” command, the I2C acknowledges the packet and clears the address resolved and address valid flags (if it is non-PSA, I2C_CR[19]=0). The I2C then proceeds to step 5 and waits for another SMBus packet.
22. The I2C checks if the received command is a “Directed Get UDID” command. If so, proceed to step 23 to return the UDID information; otherwise, go to step 24.
23. If the received command is a “Directed Get UDID” command, return the UDID information along with the current slave address, then go to step 5 and wait for another SMBus packet.
24. If the received command is not a “Directed Get UDID” command, the I2C has not received a valid ARP command, so it will NACK the command and continue with step 5 to wait for another SMBus packet.
25. If the address valid bit is set, proceed to step 26; otherwise, go to step 5 and wait for another SMBus packet. The received address is not the SMBus device default address, and the packet may be the address for I2C core functions. The device checks its address valid bit to determine whether to respond.
26. When the address valid bit is set, the I2C has a valid slave address. It compares the received slave address with its own slave address; if they match, the I2C proceeds to step 27; otherwise, it goes to step 5 and waits for another SMBus packet.
27. The I2C receives the packet addressed to its core functions, acknowledges the packet, and

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

processes it accordingly. The I2C then continues with step 5 and waits for another SMBus packet.

26.4.2 Bus Clearing

The I2C supports a bus clearing function that provides recovery of the data (SDA) and clock (SCL) lines when the clock or data lines are pulled low abnormally.

26.4.2.1 SDA Pull-down Recovery

If the SDA line is pulled down, the master will perform the following operations, as shown in FiguresFigure 26-1 andFigure 26-2.

1. The master will send a maximum of 9 clock pulses to recover the low level of the bus within these 9 clock cycles.
2. If the SDA line recovers within these 9 clock pulses, the master will send a STOP condition to release the bus.
3. If the SDA line has not recovered by the end of the 9th clock pulse, the system will require a hardware reset.

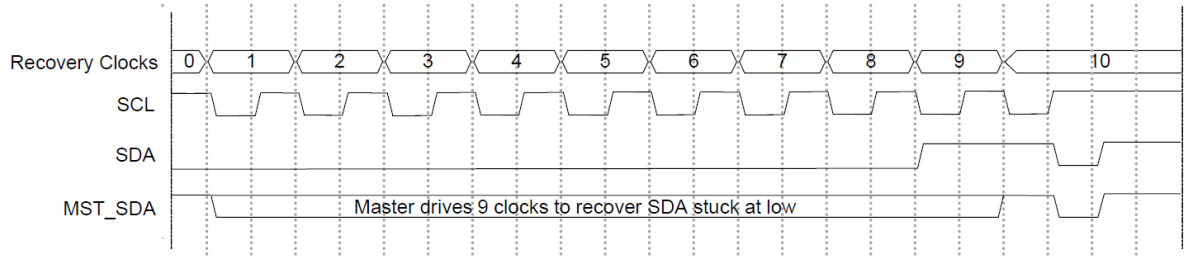

Figure 26-1 : Recovery of SDA Using 9 SCL Clock Pulses

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

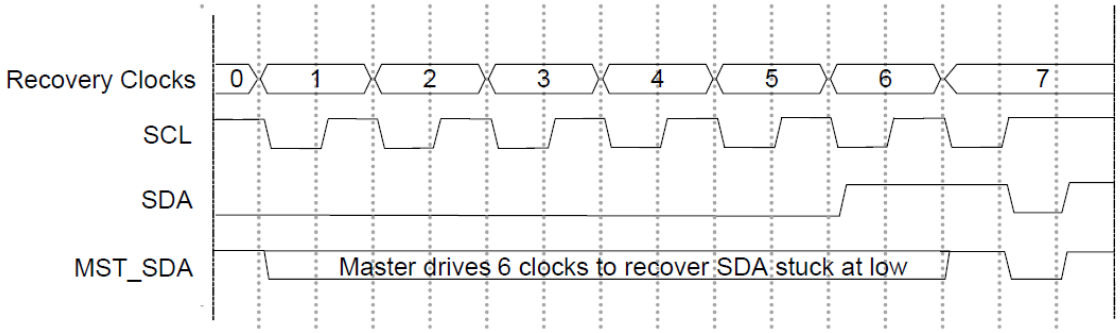

Figure 26-2 : Recovery of SDA Using 6 SCL Clock Pulses

26.4.3 Configuring SCLHCNT and SCLLCNT for I2C

When I2C operates as a master in standard mode (100 Kb/s), fast mode (400 Kb/s), or fast mode+ (1 Mb/s), the *CNT registers must be set up before I2C transmission to ensure I/O timing. The *CNT registers are as follows:

- I2C_SSSCLHCNT
- I2C_SSSCLLCNT
- I2C_FSSCLHCNT
- I2C_FSSCLLCNT

26.4.3.1 Minimum Setting Values for High and Low Levels

- The I2C_SSSCLLCNT and I2C_FSSCLLCNT registers must be greater than I2C_FSSPKLEN + 7.
- The I2C_SSSCLHCNT and I2C_FSSCLHCNT registers must be greater than I2C_FSSPKLEN + 5.

26.4.3.2 SCLHCNT and SCLLCNT Calculation Methods

The following describes how to calculate I2C_FSSCLHCNT and I2C_FSSCLLCNT in fast mode. Similarly, values for standard mode and fast mode+ can also be calculated.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- In fast mode (400 Kb/s), the SCL period is 2.5 μs.
- For example, if the APB clock of I2C is 48 MHz, then the I2C_CLK period is 20.8 ns.
- According to the protocol, the minimum time for high and low levels of SCL is as follows:

  MIN_SCL_LOWtime_FS = 1300 ns

  MIN_SCL_HIGHtime_FS = 600 ns

  MIN_SCL_LOWtime_SS = 4700 ns

  MIN_SCL_HIGHtime_SS = 4000 ns

  MIN_SCL_LOWtime_FS+ = 500 ns

  MIN_SCL_HIGHtime_FS+ = 260 ns

$$\frac{SCL_PERIOD_FS}{I2C_FSSCLHCNT + I2C_FSSCLLCNT} = I2C_CLK_PERIOD$$

I2C_FSSCLLCNT x I2C_CLK_PERIOD = MIN_SCL_LOWtime_FS

Substituting the SCL and I2C clock frequencies into the equation yields:

$$\frac{2500}{I2C_FSSCLHCNT + I2C_FSSCLLCNT} = 20.8$$

I2C_FSSCLLCNT × 20.8 = 1300

Solving for I2C_FSSCLHCNT and I2C_FSSCLLCNT:

I2C_FSSCLHCNT + I2C_FSSCLLCNT = 120.9

I2C_FSSCLLCNT = 62.5

Rounding I2C_FSSCLLCNT up gives I2C_FSSCLLCNT = 63. Consequently, I2C_FSSCLHCNT = 58.

26.4.4 SDA Hold Time

The I2C protocol specification requires that the minimum hold time for the SDA signal is 300 ns in standard mode and fast mode, and 0 ns in fast mode+.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Due to varying line delays encountered by each application, I2C includes a software-configurable I2C_SDA_HOLD register for dynamically adjusting the SDA hold time.

Bit[15:0] is used for the SDA hold time during master and slave transmission (from high to low).

Bit[23:16] is used for extended SDA transitions (if any), as long as the SCL on the receiving side is high (in either master mode or slave mode).

If different speed modes require different SDA hold times, the I2C_SDA_HOLD register must be rewritten when the speed mode changes. The I2C_SDA_HOLD register can only be written when I2C is disabled (I2C_EN[0] = 0).

26.4.4.1 SDA Timing During Reception

Speed Mode	Max. Value of RX_HOLD
Standard mode	I2C_SS_SCL_HCNT – I2C_FS_SPKLEN - 3
Standard mode or fast mode+	I2C_SS_SCL_HCNT – I2C_FS_SPKLEN - 3

26.4.4.2 SDA Timing During Transmission

The TX_HOLD register can be used to modify the timing of the SDA signal generated by I2C (ic_sda_oe). Each value in the TX_HOLD register represents one I2C clock cycle.

When I2C operates in master mode, the minimum hold time is one I2C clock cycle. Therefore, when TX_HOLD is set to 0, I2C will continue to drive SDA for one I2C clock cycle after SCL becomes 0. For other values in TX_HOLD:

- After SCL becomes 0, I2C will drive SDA for TX_HOLD * I2C clock cycles.

  When I2C operates in slave mode, the minimum hold time is (I2C_FS_SPKLEN + 7) * I2C clock cycles. Therefore, when the value of TX_HOLD is less than I2C_FS_SPKLEN + 7, I2C will continue to drive SDA for (I2C_FS_SPKLEN + 7) * I2C clock cycles after SCL becomes 0.
- After SCL becomes 0, I2C will drive SDA for TX_HOLD * I2C clock cycles.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

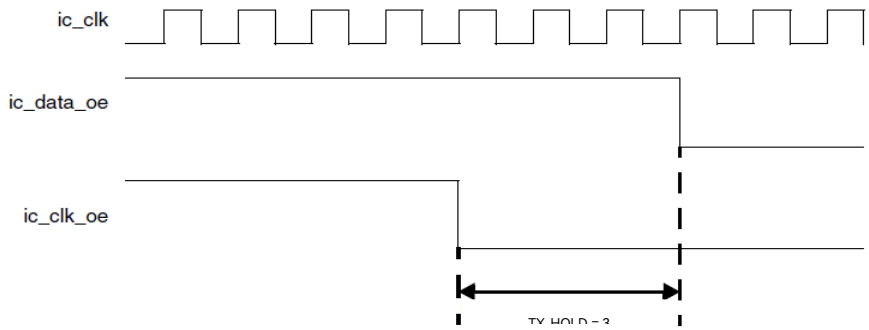

Figure 26-3: Timing Diagram for I2C Master Mode TX_HOLD = 3

26.4.5 Communication Using DMA

The I2C interface supports transmitting and receiving data using DMA. DMA transmission or DMA reception can be enabled individually by setting the corresponding bits in the DMA register. A DMA request is generated when the data register becomes empty during transmission or becomes full during reception. The DMA request must be responded to before the current byte transmission is complete.

- **DMA transmission**

  The DMA transmission mode can be activated by setting the TXEN bit in DMA register. After allocating a DMA channel for I2C, the DMA controller will load the data from the preset memory into the DR register when transmitting data.

- **DMA reception**

  The DMA reception mode can be activated by setting the RDMAE bit in DMA register. After allocating a DMA channel for I2C, the DMA controller will transfer the data from the DR register to the preset memory each time a data byte is received.

26.5 Register Description

I2C1 register base address: 0x4600_7000

I2C2 register base address: 0x4600_8000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

I2C3 register base address: 0x4600_9000

The registers are listed below:

Table 26-4: List of I2C Registers

Offset Address	Name	Description
0x00	I2C_CR	I2C control register
0x04	I2C_TAR	I2C slave address access register
0x08	I2C_SAR	I2C slave address register
0x10	I2C_DATACMD	I2C data command register
0x14	I2C_SSSCLHCNT	I2C standard mode SCL high-level configuration register
0x18	I2C_SSSCLLCNT	I2C standard mode SCL low-level configuration register
0x1C	I2C_FSSCLHCNT	I2C fast mode SCL high-level configuration register
0x20	I2C_FSSCLLCNT	I2C fast mode SCL low-level configuration register
0x2C	I2C_ISR	I2C interrupt status register
0x30	I2C_INTMASK	I2C interrupt mask register
0x34	I2C_RAWISR	I2C raw interrupt status register
0x38	I2C_RXTL	I2C RX FIFO threshold register
0x3C	I2C_TXTL	I2C TX FIFO threshold register
0x40	I2C_CLR	I2C combined and independent interrupt clear register
0x44	I2C_CLRRXUNDER	I2C RX_UNDER interrupt clear register
0x48	I2C_CLRRXOVER	I2C RX_OVER interrupt clear register
0x4C	I2C_CLRTXOVER	I2C TX_OVER interrupt clear register
0x50	I2C_CLRRDREQ	I2C RD_REQ interrupt clear register
0x54	I2C_CLRTXABRT	I2C TX_ABRT interrupt clear register
0x58	I2C_CLRRXDONE	I2C RX_DONE interrupt clear register
0x5C	I2C_CLRACTIVITY	I2C ACTIVITY interrupt clear register
0x60	I2C_CLRSTOPDET	I2C STOP_DET interrupt clear register
0x64	I2C_CLRSTARTDET	I2C START_DET interrupt clear register
0x68	I2C_CLRGENCALL	I2C GEN_CALL interrupt clear register
0x6C	I2C_EN	I2C enable register
0x70	I2C_SR	I2C status register
0x74	I2C_TXFLR	I2C transmit buffer depth register

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x78	I2C_RXFLR	I2C receive buffer depth register
0x7C	I2C_SDAHOLD	I2C SDA hold time register
0x84	I2C_SLVDATANACKONLY	I2C slave data NACK generation register
0x88	I2C_DMACR	I2C DMA control register
0x8C	I2C_DMATDLR	I2C DMA transmit data level register
0x90	I2C_DMARDLR	I2C DMA receive data level register
0x94	I2C_SDASETUP	I2C SDA setup time register
0x98	I2C_ACKGENERALCALL	I2C broadcast call ACK register
0x9C	I2C_ENSR	I2C enable status register
0xA0	I2C_FSSPKLEN	I2C spike suppression limit register
0xBC	I2C_SMBUSCLOCKLOWSEXT	I2C SMBUS slave clock extension timeout register
0xC0	I2C_SMBUSCLOCKLOWMEXT	I2C SMBUS master clock extension timeout register
0xC4	I2C_SMBUSTHIGHMAXIDLECOUNT	I2C SMBUS maximum bus idle count register
0xC8	I2C_SMBUSISR	I2C SMBUS interrupt status register
0xCC	I2C_SMBUSINTMASK	I2C SMBUS interrupt mask register
0xD0	I2C_SMBUSRAWISR	I2C SMBUS raw interrupt status register
0xD4	I2C_CLRSMBUSISR	I2C SMBUS interrupt status clear register
0xDC	I2C_SMBUSUDIDLSB	I2C SMBUS ARP UDID LSB register

26.5.1 I2C Control Register (I2C_CR)

Offset address: 0x00

Reset value: 0x0000 0065

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	Reserved

Unicmicro

Bit	Name	Attribute	Reset Value	Description
19	SMBUS_PERSISTANT_SLV_ADDR_EN	R/W	0x0	Enable the I2C as a persistent or non-persistent slave.
18	SMBUS_ARP_EN	R/W	0x0	Only for slave mode: Control whether I2C enables address resolution logic in SMBus mode.
17	SMBUS_SLAVE_QUICK_CMD_EN	R/W	0x0	Only for slave mode: 0: I2C receives all protocols except the QUICK command. 1: In SMBus mode, I2C receives only the QUICK command.
16:12	RSV	-	-	Reserved
11	BUS_CLEAR_FEATURE_CTRL	R/W	0x0	Only for master mode: 0: Bus clearing function enabled 1: Bus clearing function disabled
10:9	RSV	-	-	Reserved
8	TX_EMPTY_CTRL	R/W	0x0	This bit controls the generation of the TX_EMPTY interrupt; please refer to the I2C_RAW_INTR_SR register for details.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7	STOP_DET_IFADDRESSED	R/W	0x0	Generation of STOP_DET interrupt in slave mode: 1: STOP_DET interrupt is generated only when there is an address match. 0: STOP_DET interrupt is generated regardless of whether there is an address match. This bit is applicable only in slave mode. Note: When the broadcast address is addressed, if this bit is set, the slave will not generate a STOP_DET interrupt. The STOP_DET interrupt is generated only when the sent address matches the slave address.
6	SLAVE_DISABLE	R/W	0x1	I2C slave enable: 0: Slave enabled 1: Slave disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	RESTART_EN	R/W	0x1	In master mode, this bit controls whether to send a RESTART condition: 0: Disabled 1: Enabled When RESTART is disabled, the I2C interface as a master cannot perform the following functions: ● Sending a start byte ● Changing the transfer direction in combined format mode ● Read operation in 10-bit addressing format After replacing the RESTART condition, sending a stop condition followed by a start condition will result in splitting into multiple I2C transfers. If any of the above operations are executed, it will set bit 6 (TX_ABRT) of the IC_RAW_INTR_STAT register.
4	10ADDR_MASTER	R	0x0	Addressing mode when I2C operates as a master: 0: 7-bit addressing mode 1: 10-bit addressing mode
3	10ADDR_SLAVE	R/W	0x0	In slave mode, this bit controls the response to 10-bit or 7-bit addressing: 0: 7-bit addressing; The I2C ignores the 10-bit addressing. For 7-bit addressing, only the lower 7 bits of the IC_SAR register are compared. 1: 10-bit addressing; The I2C only responds to 10-bit addressing, comparing the received address with the 10 bits of IC_SAR.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2:1	SPEED	R/W	0x2	These two bits control the operating speed mode of the I2C. This setting is only valid when the I2C is in master mode. 1: Standard mode (0–100 Kb/s) 2: Fast mode (400 Kb/s) or fast mode+ (0–1000 Kb/s)
0	MASTER_MODE	R/W	0x1	Master mode enable: 0: Disabled 1: Enabled

SLAVE_DISABLE (I2C_CR[6]) and MASTER_MODE (I2C_CR[0]) are configured as show in the following table.

Table 26-5: I2C Slave and Master Configuration Table

SLAVE_DISABLE (I2C_CR[6])	MASTER_MODE (I2C_CR[0])	State
0	0	Slave
0	1	Setting error
1	0	Setting error
1	1	Master

26.5.2 I2C Slave Address Access Register (I2C_TAR)

Offset address: 0x04

Reset value: 0x0000 0055

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	SMBUS_QUICK_CMD	R/W	0x0	When bit 11 (SPECIAL) is set to 1, this bit indicates whether the I2C executes the QUICK command.
15:12	RSV	-	-	Reserved
11	SPECIAL	R/W	0x0	This bit indicates whether the software is executing a special

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				command (broadcast call or start byte command): 0: Ignore the 10th bit GC, normally use the ADDR bit. 1: Execute special I2C commands as described by the GC bit.
10	GC_OR_START	R/W	0x0	If bit 11 is set, this bit indicates whether the I2C is executing a broadcast call or a start byte command: 0: Broadcast call address; only write operations can be performed when sending a broadcast call address. The I2C interface remains in broadcast addressing mode until the value of SPECIAL (bit 11) is cleared. 1: Start byte command
9:0	ADDR	R/W	0x55	Destination address of the main operation: These bits can be ignored when sending a broadcast address. To generate a start byte command, the CPU only needs to write to these bits once.

26.5.3 I2C Slave Address Register (I2C_SAR)

Offset address: 0x08

Reset value: 0x0000 0055

Bit	Name	Attribute	Reset Value	Description
31:10	RSV	-	-	Reserved
9:0	ADDR	R/W	0x55	When the I2C is in slave mode, the slave address is saved. For 7-bit addressing mode, ADDR[6:0] is valid.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.4 I2C Data Command Register (I2C_DATACMD)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved
10	RESTART	W	0x0	Generation of RESTART signal before transmitting or receiving data: 1: If the RESTART_EN signal is “1”, a RESTART signal is generated before data is received or transmitted (depending on the value of CMD), regardless of whether the previous command changes the direction of data transmission. If the RESTART_EN signal is “0”, a START signal immediately follows the STOP signal. 0: If the RESTART_EN signal is “1”, a RESTART signal is generated only when the previous command changes the direction of data transmission. If the RESTART_EN signal is “0”, a START signal immediately follows the STOP signal.
9	STOP	W	0x0	Generation of STOP signal after transmitting or receiving data: 1: A STOP signal is generated after the current byte, regardless of whether the TX FIFO is empty. If the TX FIFO is not empty, the master immediately issues a new transfer and bus arbitration signal. 0: A STOP signal is not generated after the current byte, regardless of whether the TX FIFO is empty. The master continues the current transmission (transmitting or receiving data based on the value of CMD). If the TX FIFO is empty, the master will pull SCL low to

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				suspend the bus until TX FIFO receives new data.
8	CMD	W	0x0	Performing read or write operation in master mode: 1: Read 0: Write This bit is used to distinguish between read and write commands when a command is entered into TXFIFO. In RX mode, the bit write operation is ignored. In TX mode, 0 indicates that the data in DATA has been transmitted.
7:0	DATA	R/W	0x0	Data to be transmitted or received on the I2C bus: If a write operation is performed on DATA followed by a read, the read operation is invalid. However, when reading this register, the returned value is the data received by the I2C.

26.5.5 I2C Standard Mode SCL High-level Configuration Register (I2C_SSSCLHCNT)

Offset address: 0x14

Reset value: 0x0000 0028

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SS_SCL_HCNT	R/W	0x28	SCL clock high-level cycle in I2C standard mode Note: This register is configurable between 6 and 65525, because the I2C interface uses a 16-bit counter whose value is equal to I2C_SSSCLHCNT + 10, indicating that the I2C bus is in an idle state.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.6 I2C Standard Mode SCL Low-level Configuration Register (I2C_SSSCLLCNT)

Offset address: 0x18

Reset value: 0x0000 002F

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SS_SCL_LCNT	R/W	0x2F	SCL clock low-level cycle in I2C standard mode The minimum value is 8.

26.5.7 I2C Fast Mode SCL High-level Configuration Register (I2C_FSSCLHCNT)

Offset address: 0x1C

Reset value: 0x0000_0006

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	FS_SCL_HCNT	R/W	0x6	SCL clock high-level cycle in I2C fast mode The minimum value is 6.

26.5.8 I2C Fast Mode SCL Low level Configuration Register (I2C_FSSCLLCNT)

Offset address: 0x20

Reset value: 0x0000 000D

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	FS_SCL_LCNT	R/W	0xD	SCL clock low-level cycle in I2C fast mode The minimum value is 8.

26.5.9 I2C Interrupt Status Register (I2C_ISR)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11	GEN_CALL	R	0x0	General call request received interrupt status: 0: No interrupt 1: Interrupt occurred
10	START_DET	R	0x0	Start detection interrupt status, indicating whether the I2C bus interface has detected a START or RESTART condition: 0: No interrupt 1: Interrupt occurred
9	STOP_DET	R	0x0	Stop detection interrupt status, indicating whether the I2C bus interface has detected a STOP condition: 0: No interrupt 1: Interrupt occurred
8	ACTIVITY	R	0x0	Activity interrupt status, recording the I2C activity status until cleared: 0: No interrupt 1: Interrupt occurred
7	RX_DONE	R	0x0	Reception complete interrupt status: 0: No interrupt 1: Interrupt occurred
6	TX_ABRT	R	0x0	Transmission abort interrupt status: 0: No interrupt 1: Interrupt occurred

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	RD_REQ	R	0x0	In slave mode, interrupt status for other masters attempting to read data from the I2C interface: 0: No interrupt 1: Interrupt occurred
4	TX_EMPTY	R	0x0	Interrupt status of TX FIFO reaching or falling below the threshold: 0: No interrupt 1: Interrupt occurred
3	TX_OVER	R	0x0	TX FIFO overflow interrupt status: 0: No interrupt 1: Interrupt occurred
2	RX_FULL	R	0x0	Interrupt status of RX FIFO reaching or exceeding the threshold: 0: No interrupt 1: Interrupt occurred
1	RX_OVER	R	0x0	RX FIFO overflow interrupt status: 0: No interrupt 1: Interrupt occurred
0	RX_UNDER	R	0x0	Data read overflow (i.e., CPU reading from FIFO when RX_FIFO is empty) interrupt status: 0: No interrupt 1: Interrupt occurred

26.5.10 I2C Interrupt Mask Register (I2C_INTMASK)

Offset address: 0x30

Reset value: 0x0000 08FF

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11	GEN_CALL	R/W	0x1	General call request received interrupt mask: 0: Masked 1: Unmasked

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	START_DET	R/W	0x0	Start detection interrupt mask: 0: Masked 1: Unmasked
9	STOP_DET	R/W	0x0	Stop detection interrupt mask: 0: Masked 1: Unmasked
8	ACTIVITY	R/W	0x0	Activity interrupt mask: 0: Masked 1: Unmasked
7	RX_DONE	R/W	0x1	Reception complete interrupt mask: 0: Masked 1: Unmasked
6	TX_ABRT	R/W	0x1	Transmission abort interrupt mask: 0: Masked 1: Unmasked
5	RD_REQ	R/W	0x1	As a slave, interrupt mask of another I2C master requesting data reading: 0: Masked 1: Unmasked
4	TX_EMPTY	R/W	0x1	Interrupt mask of TX FIFO reaching or falling below the threshold: 0: Masked 1: Unmasked
3	TX_OVER	R/W	0x1	TX FIFO overflow interrupt mask: 0: Masked 1: Unmasked
2	RX_FULL	R/W	0x1	Interrupt mask of RX FIFO reaching or exceeding the threshold: 0: Masked 1: Unmasked
1	RX_OVER	R/W	0x1	RX FIFO overflow interrupt mask: 0: Masked 1: Unmasked

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	RX_UNDER	R/W	0x1	Data read overflow (i.e., CPU reading from FIFO when RX FIFO is empty) interrupt mask: 0: Masked 1: Unmasked

26.5.11 I2C Raw Interrupt Register (I2C_RAWISR)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11	GEN_CALL	R	0x0	Broadcast call: This bit is set when broadcast call address is received. It is cleared when the I2C interface is disabled or when the CPU reads the GC register. I2C will store the received data in receive buffer.
10	START_DET	R	0x0	Start condition detection: Regardless of whether the I2C interface is operating in master or slave mode, this bit is set once a start or repeated start condition is detected on the I2C interface.
9	STOP_DET	R	0x0	Stop condition detection: The status of this bit is based on the status of STOP INT in the CR register. When STOP INT = 0, this bit is set once a stop condition is detected on the I2C interface, regardless of whether the I2C interface is operating in master or slave mode. In slave mode, a STOP interrupt

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				will occur regardless of address matching when STOPINT = 1. In master mode (MASTER = 1), this bit indicates whether a stop condition has occurred on the I2C interface. In slave mode (MASTER = 0), a STOP interrupt is generated only when the slave address matches successfully.
8	ACTIVITY	R	0x0	I2C interface activation: this bit is used to capture the active state of the I2C module. Once set, it can only be cleared by one of the following four methods: Disabling the I2C interface Reading the I2C_CLR_ACTIVITY register Reading the I2C_CLR register System reset Once set, it can only be cleared by the above methods, even if the I2C is idle, this bit will remain high until cleared.
7	RX_DONE	R	0x0	Slave transmission complete: When the I2C operates as a slave transmitter, this bit will be set if the master does not respond after transmitting a byte of data. This happens after the last byte of transmission, indicating the end of transmission.
6	TX_ABRT	R	0x0	Transmission abort: This bit is set when the I2C interface operates as a transmitter and the data in the buffer cannot be transmitted completely. Note: Transmission abort will clear both the receive and transmit buffers in the I2C interface. The transmit buffer will remain refreshed until the TX_ABRT register is read. Once the read operation

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				is performed, the transmitter can receive new data on the APB bus.
5	RD_REQ	R	0x0	Read request: When the I2C operates as a slave, this bit is set if another master attempts to read data from the I2C interface. The I2C interface will keep the bus waiting (SCL = 0) until the interrupt is processed. This means that the I2C interface as a slave has been successfully addressed by another master and is required to transmit data. The processor must respond to this interrupt and then write data to the I2C_DATA_CMD register. This bit is set when the processor reads the RD_REQ register.
4	TX_EMPTY	R	0x0	Transmit buffer empty: The status of this bit depends on the EMPINT status in the CR register. When EMPINT = 0, the bit is set when the transmit buffer is empty. When EMPINT = 1, the bit is set when the transmit buffer is empty and the internal shift register has completed. This bit is automatically cleared by hardware when the transmit buffer is not empty.
3	TX_OVER	R	0x0	Transmit buffer overrun: This bit is set when the transmit buffer is full and the processor writes new data resulting in an overflow.
2	RX_FULL	R	0x0	Receive buffer not empty: This bit is set when the receive buffer is not empty. This bit is cleared by hardware when the receive buffer is below the threshold.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	RX_OVER	R	0x0	Receive buffer overrun: This bit is set when the receive buffer is full and new data is received. In this case, the I2C interface will respond, but the new data will be lost.
0	RX_UNDER	R	0x0	Receive buffer underrun: This bit is set when the processor reads the DR register while the RX FIFO is empty.

26.5.12 I2C RX FIFO Threshold Register (I2C_RXTL)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	RX_TL	R/W	0x0	RX FIFO threshold: Control the trigger of RX_FULL interrupt.

26.5.13 I2C TX FIFO Threshold Register (I2C_TXTL)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	TX_TL	R/W	0x0	TX FIFO threshold: Control the trigger of TX_EMPTY interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.14 I2C Combined and Independent Interrupt Clear Register (I2C_CLR)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_INTR	R	0x0	Reading this register will clear all combined interrupts and independent interrupts. This bit does not clear the interrupts that can be cleared automatically by hardware, but clear only the interrupts that can be cleared by software.

26.5.15 I2C RX_UNDER Interrupt Clear Register (I2C_CLRRXUNDER)

Offset address: 0x44

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_RX_UNDER	R	0x0	Reading this register clears the RX_UNDER interrupt (I2C_RAW_ISR[0]).

26.5.16 I2C RX_OVER Interrupt Clear Register (I2C_CLRRXOVER)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_RX_OVER	R	0x0	Reading this register clears the RX_OVER interrupt (I2C_RAW_ISR[1]).

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.17 I2C TX_OVER Interrupt Clear Register (I2C_CLRTXOVER)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_TX_OVER	R	0x0	Reading this register clears the RX_OVER interrupt (I2C_RAW_ISR[3]).

26.5.18 I2C RD_REQ Interrupt Clear Register (I2C_CLRRDREQ)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_RD_REQ	R	0x0	Reading this register clears the RD_REQ interrupt (I2C_RAW_ISR[5]).

26.5.19 I2C TX_ABRT Interrupt Clear Register (I2C_CLRTXABRT)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_TX_ABRT	R	0x0	Reading this register clears the TX_ABRT interrupt (I2C_RAW_ISR6]). It also releases the TX FIFO from the refresh/reset state in order to receive the written data.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.20 I2C RX_DONE Interrupt Clear Register (I2C_CLRRXDONE)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_RX_DONE	R	0x0	Reading this register clears the RX_DONE interrupt (I2C_RAW_ISR[7]).

26.5.21 I2C ACTIVITY Interrupt Clear Register (I2C_CLRACTIVITY)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_ACTIVITY	R	0x0	If the I2C bus is inactive, read this register to clear the ACTIVITY interrupt (I2C_RAW_ISR[8]). If the I2C is still active, then the ACTIV interrupt will remain set. This bit is cleared by hardware when the I2C module is disabled or when the I2C bus is no longer active. The status of ACTIVITY(8) in I2C_RAW_ISR can be obtained by reading this register.

26.5.22 I2C STOP_DET Interrupt Clear Register (I2C_CLRSTOPDET)

Offset address: 0x60

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_STOP_DET	R	0x0	Reading this register clears the STOP interrupt (I2C_RAW_ISR[9]).

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.23 I2C START_DET Interrupt Clear Register (I2C_CLRSTARTDET)

Offset address: 0x64

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_START_DET	R	0x0	Reading this register clears the START interrupt (I2C_RAW_ISR[10]).

26.5.24 I2C GEN_CALL Interrupt Clear Register (I2C_CLRGENCALL)

Offset address: 0x68

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CLR_GEN_CALL	R	0x0	Reading this register clears the GC interrupt (I2C_RAW_ISR[11]).

26.5.25 I2C Enable Register (I2C_EN)

Offset address: 0x6C

Reset value: 0x0000 0004

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	SMBUS_CLK_RESET	R/W	0x0	Release SMBus master clock reset in SMBus master mode
15:4	RSV	-	-	Reserved
3	SDA_STUCK_RECOVERY_ENABLE	R/W	0x0	This bit is used as part of the SDA recovery mechanism if SDA is pulled low at the TX_ABORT

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				interrupt. This bit will be automatically cleared.
2	TX_CMD_BLOCK	R/W	0x1	In master mode: 0: Once the first data in the TX FIFO is available, I2C will automatically start transmitting data. 1: Even if there is data to be transmitted in the TX FIFO, the I2C data transmission will be prevented.
1	ABORT	R/W	0x0	I2C transmission abort: 0: Abort did not occur or has already ended. 1: Abort operation is in progress. When the I2C module is set to operate as a master, the I2C transmission can be aborted via software. Once set, it cannot be cleared immediately. After being set, the I2C module control logic will generate a STOP condition and clear the transmit buffer after completing the current transmission, and a TX_ABRT interrupt will be generated after the abort operation. The ABORT bit will be cleared automatically after the abort operation.
0	ENABLE	R/W	0x0	I2C module enable: 0: I2C module disabled (transmit and receive buffer remain empty) 1: I2C module enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.26 I2C Status Register (I2C_SR)

Offset address: 0x70

Reset value: 0x0000 0006

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	SMBUS_SLAVE_ADDR_RESOLVED	R	0x0	This bit is set when the slave address (I2C_SAR[6:0]) is resolved by the ARP master.
17	SMBUS_SLAVE_ADDR_VALID	R	0x0	This bit is set when the slave address (I2C_SAR[6:0]) is valid.
16	SMBUS_QUICK_CMD	R	0x0	This bit is set when the R/W bit of the QUICK command is received.
15:12	RSV	-	-	Reserved
11	SDA_STUCK_NOT_RECOVERED	R	0x0	This bit is set when SDA is still pulled low after the recovery mechanism.
10:7	RSV	-	-	Reserved
6	SLV_ACTIVITY	R	0x0	Slave state machine active status bit: 0: The slave state machine is in IDLE state, so the I2C slave part is inactive. 1: The slave state machine is not in IDLE state, so the I2C slave part is active.
5	MST_ACTIVITY	R	0x0	Master state machine active status bit: 0: The master state machine is in IDLE state, so the I2C master part is inactive. 1: The master state machine is not in IDLE state, so the I2C master part is active.
4	RFF	R	0x0	Receive buffer full:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Receive buffer not full 1: Receive buffer full
3	RFNE	R	0x0	Receive buffer not empty: 0: Receive buffer empty 1: Receive buffer not empty
2	TFE	R	0x1	Transmit buffer empty: 0: Transmit buffer not empty 1: Transmit buffer empty
1	TFNF	R	0x1	Transmit buffer not full: 0: Transmit buffer not full 1: Transmit buffer full
0	ACTIVITY	R	0x0	I2C activity status: the result of the logical OR operation between the MST_ACTIVITY bit and the SLV_ACTIVITY bit.

26.5.27 I2C Transmit Buffer Depth Register (I2C_TXFLR)

Offset address: 0x74

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
15:3	RSV	-	-	Reserved
2:0	CNT	R	0x0	Number of valid data in transmit buffer (0–7)

26.5.28 I2C Receive Buffer Depth Register (I2C_RXFLR)

Offset address: 0x78

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
15:3	RSV	-	-	Reserved
2:0	CNT	R	0x0	Number of valid data in receive buffer (0–7)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.29 I2C SDA Hold Time Register (I2C_SDAHOLD)

Offset address: 0x7C

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	-	-	Reserved
23:16	RX_HOLD	R/W	0x0	When I2C is in receive mode, the SDA hold time is measured in units of APB1 clock cycles.
15:0	TX_HOLD	R/W	0x1	When I2C is in transmit mode, the SDA hold time is measured in units of APB1 clock cycles.

26.5.30 I2C Slave Data NACK Generation Register (I2C_SLVDATANACKONLY)

Offset address: 0x84

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	NACK	R/W	0x0	I2C as slave receiver: 0: Normally generate NACK and ACK 1: Generate NACK after receiving data

26.5.31 I2C DMA Control Register (I2C_DMACR)

Offset address: 0x88

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
15:2	RSV	-	-	Reserved

Unicmicro

Bit	Name	Attribute	Reset Value	Description
1	TDMAE	R/W	0x0	Transmit DMA enable: 0: Transmit DMA disabled 1: Transmit DMA enabled
0	RDMAE	R/W	0x0	Receive DMA enable: 0: Receive DMA disabled 1: Receive DMA enabled

26.5.32 I2C DMA Transmit Data Level Register (I2C_DMATDLR)

Offset address: 0x8C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
15:2	RSV	-	-	Reserved
1:0	DMATDL	R/W	0x0	When TDMAE = 1, dma_tx_req is generated when the data in TX FIFO is equal to or less than the value of DMATDL.

26.5.33 I2C DMA Receive Data Level Register (I2C_DMARDLR)

Offset address: 0x90

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
15:2	RSV	-	-	Reserved
1:0	DMARDL	R/W	0x0	When RDMAE = 1, dma_rx_req is generated when the data in RX FIFO is equal to or greater than (DMARDL + 1).

26.5.34 I2C SDA Setup Time Register (I2C_SDASETUP)

Offset address: 0x94

Reset value: 0x0000 0064

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	SDA_SETUP	R/W	0x64	SDA setup time: If the recommended setup delay time is 1000 ns, and the APB1 clock frequency is 10 MHz, it is recommended to set this register to 11. The minimum value of this register is 2.

26.5.35 I2C Broadcast Call ACK Register (I2C_ACKGENERALCALL)

Offset address: 0x98

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	ACK_GEN_CALL	R/W	0x1	Broadcast call ACK: 1: Respond with ACK after receiving a broadcast call. 0: Does not respond after receiving a broadcast call, and does not generate an interrupt.

26.5.36 I2C Enable Status Register (I2C_ENSR)

Offset address: 0x9C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2	SLV_RX_DATA_LOST	R	0x0	Slave data loss: 1: The I2C transmission has been terminated. Although the data byte responded with NACK, it has entered the I2C data transmission phase. 0: The I2C has been disabled, and the slave reception has not entered the data transfer phase.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	SLV_DISABLED_WHILE_BUSY	R	0x0	The slave is disabled while busy.
0	IC_EN	R	0x0	I2C enable status: 0: Disabled 1: Enabled

26.5.37 I2C Spike Suppression Limit Register (2C_FSSPKLEN)

Offset address: 0xA0

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	I2C_FS_SPKLEN	R/W	0x1	Before performing any I2C bus transactions, this register must be configured to ensure stable operation. Writing is valid only when I2C_CR[0] = 0.

26.5.38 I2C SMBUS Slave Clock Extension Timeout Register (I2C_SMBUSCLOCKLOWSEXT)

Offset address: 0xBC

Reset value: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	SMBUS_CLK_LOW_SEXT_TIMEOUT	R/W	0xFFFFFFFF	Used to detect the slave clock extension timeout in master mode, where the slave extends the time from start to stop.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.39 I2C SMBUS Master Clock Extension Timeout Register (I2C_SMBUSCLOCKLOWMEXT)

Offset address: 0xC0

Reset value: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	SMBUS_CLK_LOW_MEXT_TIMEOUT	R/W	0xFFFFFFFF	Used to detect the timeout for the extension of SMBus clock (SCL) from start to ACK, ACK to ACK, or ACK to stop in master mode.

26.5.40 I2C SMBUS Maximum Bus Idle Count Register (I2C_SMBUSTHIGHMAXIDLECOUNT)

Offset address: 0xC4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SMBUS_THIGH_MAX_BUS_IDLE_CNT	R/W	0xFFFF	Used to set the required bus idle time period when a master device has been dynamically added to the bus and the state transitions on the SMBCLK or SMBDAT lines may not have been detected yet.

26.5.41 I2C SMBus Interrupt Status Register (I2C_SMBUSISR)

Offset address: 0xC8

Reset value: 0x0000 0000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	R_SLV_RX_PEC_NACK	R	0x0	Interrupt status for the slave generating NACK for the PEC byte from the slave ARP command: 0: No interrupt 1: Interrupt occurred
7	R_ARP_ASSGN_ADDR_CMD_DET	R	0x0	Interrupt status for receiving an Assign Address ARP: 0: No interrupt 1: Interrupt occurred
6	R_ARP_GET_UDID_CMD_DET	R	0x0	Interrupt status for receiving a general or directed Get UDID ARP command: 0: No interrupt 1: Interrupt occurred
5	R_ARP_RST_CMD_DET	R	0x0	Interrupt status for receiving a general or directed Reset ARP command: 0: No interrupt 1: Interrupt occurred
4	R_ARP_PREPARE_CMD_DET	R	0x0	Interrupt status for receiving a Prepare to ARP command: 0: No interrupt 1: Interrupt occurred
3	R_HOST_NOTIFY_MST_DET	R	0x0	Interrupt status for receiving a Host Notify command: 0: No interrupt 1: Interrupt occurred
2	R_QUICK_CMD_DET	R	0x0	Interrupt status for receiving a Quick command: 0: No interrupt 1: Interrupt occurred
1	R_MST_CLOCK_EXTND_TIMEOUT	R	0x0	Interrupt status for master transactions from start to stop (START to ACK, ACK to ACK, or ACK to STOP) exceeding

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				I2C_SMBUS_CLOCK_LOW_MEXT time for each byte in the message: 0: No interrupt 1: Interrupt occurred
0	R_SLV_CLOCK_EXTND_TIMEOUT	R	0x0	Interrupt status for slave transactions (start to stop) exceeding IC_SMBUS_CLOCK_LOW_SEXT time: 0: No interrupt 1: Interrupt occurred

26.5.42 I2C SMBus Interrupt Mask Register (I2C_SMBUSINTMASK)

Offset address: 0xCC

Reset value: 0x0000 01FF

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	M_SLV_RX_PEC_NACK	R/W	0x1	Interrupt mask for the slave generating NACK for the PEC byte from the slave ARP command: 0: Masked 1: Unmasked
7	M_ARP_ASSGN_ADDM_CMD_DET	R/W	0x1	Interrupt mask for receiving an Assign Address ARP: 0: Masked 1: Unmasked
6	M_ARP_GET_UDID_CMD_DET	R/W	0x1	Interrupt mask for receiving a general or directed Get UDID ARP command: 0: Masked 1: Unmasked

Unicmicro

Bit	Name	Attribute	Reset Value	Description
5	M_ARP_RST_CMD_DET	R/W	0x1	Interrupt mask for receiving a general or directed Reset ARP command: 0: Masked 1: Unmasked
4	M_ARP_PREPARE_CMD_DET	R/W	0x1	Interrupt mask for receiving a Prepare to ARP command: 0: Masked 1: Unmasked
3	M_HOST_NOTIFY_MST_DET	R/W	0x1	Interrupt mask for receiving a Host Notify command: 0: Masked 1: Unmasked
2	M_QUICK_CMD_DET	R/W	0x1	Interrupt mask for receiving a Quick command: 0: Masked 1: Unmasked
1	M_MST_CLOCK_EXTND_TIMEOUT	R/W	0x1	Interrupt mask for master transactions from start to stop (START to ACK, ACK to ACK, or ACK to STOP) exceeding I2C_SMBUS_CLOCK_LOW_MEXT time for each byte in the message: 0: Masked 1: Unmasked
0	M_SLV_CLOCK_EXTND_TIMEOUT	R/W	0x1	Interrupt mask for slave transactions (start to stop) exceeding IC_SMBUS_CLOCK_LOW_SEXT time: 0: Masked 1: Unmasked

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.43 I2C SMBus RAW Interrupt Status Register (I2C_SMBUSRAWISR)

Offset address: 0xD0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	SLV_RX_PEC_NACK	R	0x0	The slave generates a NACK for the PEC byte from the slave ARP command.
7	ARP_ASSGN_ADDCMD_DET	R	0x0	An Assign Address ARP command is received.
6	ARP_GET_UDID_CMD_DET	R	0x0	A general or directed Get UDID ARP command is received.
5	ARP_RST_CMD_DET	R	0x0	A general or directed Reset ARP command is received.
4	ARP_PREPARE_CMD_DET	R	0x0	A Prepare to ARP command is received.
3	HOST_NOTIFY_MST_DET	R	0x0	A Host Notify command is received.
2	QUICK_CMD_DET	R	0x0	A Quick command is received.
1	MST_CLOCK_EXTND_TIMEOUT	R	0x0	Master transactions from start to stop (START to ACK, ACK to ACK, or ACK to STOP) exceed the I2C_SMBUS_CLOCK_LOW_MEXT time for each byte in the message. This bit is valid only when I2C_CR[0] = 1.
0	SLV_CLOCK_EXTND_TIMEOUT	R	0x0	Slave transactions (start to stop) exceed the IC_SMBUS_CLOCK_LOW_SEXT time. This bit is valid only when I2C_CR[0] = 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.5.44 I2C SMBus Interrupt Status Clear Register (I2C_CLRSMBUSISR)

Offset address: 0xD4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	CLR_SLV_RX_PEC_NACK	W	0x0	Writing 1 to this bit clears the interrupt status for the slave generating NACK for the PEC byte from the slave ARP command.
7	CLR_ARP_ASSGN_ADDCLR_CMD_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving an Assign Address ARP.
6	CLR_ARP_GET_UDID_CMD_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving a general or directed Get UDID ARP command.
5	CLR_ARP_RST_CMD_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving a general or directed Reset ARP command.
4	CLR_ARP_PREPARE_CMD_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving a Prepare to ARP command.
3	CLR_HOST_NOTIFY_MST_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving a Host Notify command.
2	CLR_QUICK_CMD_DET	W	0x0	Writing 1 to this bit clears the interrupt status for receiving a Quick command.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	CLR_MST_CLOCK_EXTND_TIMEOUT	W	0x0	Writing 1 to this bit clears the interrupt status for master transactions from start to stop (START to ACK, ACK to ACK, or ACK to STOP) exceeding I2C_SMBUS_CLOCK_LOW_MEXT time for each byte in the message.
0	CLR_SLV_CLOCK_EXTND_TIMEOUT	W	0x0	Writing 1 to this bit clears the interrupt status for slave transactions (start to stop) exceeding IC_SMBUS_CLOCK_LOW_SEXT time.

26.5.45 I2C SMBus ARP UDID LSB register (I2C_ SMBUSUDIDLSB)

Offset address: 0xDC

Reset value: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	IC_SMBUS_ARP_UDID_LSB	R/W	0xFFFFFFFF	Under the address resolution protocol, the unique device identifier of LSB 32-bit slave is stored.

26.6 Operation Procedure

The interface can operate in one of the four following modes:

- Slave transmitter
- Slave receiver
- Master transmitter
- Master receiver

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Note: The I2C interface module can only operate in either master mode or slave mode, but cannot work in both modes simultaneously. Therefore, it is important to ensure that the registers I2C_CR[6] and I2C_CR[0] cannot be set to 0 and 1 (or 1 and 0) respectively at the same time.

26.6.1 Master and Slave Initialization Configuration

Master initialization:

1. Configure the GPIO alternate function to enable the I2Cx clock in RCM module.
2. Set I2C_EN[0] to 0 to disable the I2C module.
3. Set I2C_CR[6] to 1 to disable the slave.
4. Configure I2C_CR[4] to set the master address format to 7-bit / 10-bit.
5. Configure I2C_CR[2:1] to set the I2C operating speed.
6. To ensure communication timing, the value of CNT shall be configured before I2C transmission. In standard mode, configure I2C_SSSCLHCNT[15:0] and I2C_SSSCLLCNT[15:0]. In fast mode, configure I2C_FSSCLHCNT[15:0] and I2C_FSSCLLCNT[15:0].
7. Configure I2C_RXTL[7:0] and I2C_TXTL[7:0] to set the thresholds of RX FIFO and TX FIFO, which are set to 0 here.
8. Set I2C_CR[0] to 1 to enable the master.
9. Set I2C_EN[0] to 1 to enable the I2C module.

Slave initialization:

1. Configure the GPIO alternate function to enable the I2Cx clock in RCM module.
2. Set I2C_EN[0] to 0 to disable the I2C module.
3. Configure I2C_CR[3] to set the slave address format to 7-bit / 10-bit.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Configure I2C_SAR[9:0] and write the slave address according to the set slave address format.
5. Set I2C_CR[6] to 0 to enable the slave.
6. Set I2C_CR[7] to 1 to generate a STOP_DET interrupt only when the address matches.
7. Set I2C_CR[0] to 0 to disable the master.
8. Configure I2C_RXTL[7:0] and I2C_TXTL[7:0] to set the thresholds of RX FIFO and TX FIFO, which are set to 0 here.
9. Set I2C_EN[0] to 1 to enable the I2C module.

26.6.2 Master Transmitter

1. Configure I2C_TAR[9:0] to set the destination address of the slave to be operated on.
2. Set I2C_DATACMD[8] to 0 to specify a write operation. At the same time, write the data to be transmitted into I2C_DATACMD[7:0].
3. After each operation on the I2C_DATACMD register, read I2C_RAWISR[4] until the read status value is 1, indicating that the data in the transmit buffer has been transmitted (the condition for setting I2C_RAWISR[4] is determined by I2C_TXTL[7:0]).
4. To transmit a single data byte, it is also required to set I2C_DATACMD[9] to 1 in step 2 to generate a STOP signal after transmitting the current byte.
5. To transmit multiple data bytes, repeat steps 2 and 3. When transmitting the last data byte, set I2C_DATACMD[8] to 0 to set the write operation, and set I2C_DATACMD[9] to 1 to generate a STOP signal after transmitting the current byte. At the same time, write the data to be transmitted into I2C_DATACMD[7:0].

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.6.3 Master Receiver

1. Configure I2C_TAR[9:0] to set the destination address of the slave to be operated on.
2. Set I2C_DATACMD[8] to 1 to specify a read operation.
3. Read I2C_RAWISR[2] until the read status value is 1, indicating that the receive buffer has received one data byte (the condition for setting I2C_RAWISR[2] is determined by I2C_RXTL[7:0]). At this point, read the data from I2C_DATACMD[7:0].
4. To receive a single data byte, it is also required to set I2C_DATACMD[9] to 1 in step 2 to generate a STOP signal after reading the current byte.
5. To transmit multiple data bytes, repeat steps 2 and 3. When receiving the last data byte, set I2C_DATACMD[8] to 1 to set the read operation, and set I2C_DATACMD[9] to 1 to generate a STOP signal after reading the current byte.

26.6.4 Slave Transmitter

1. Read I2C_RAWISR[5] until the read status value is 1, indicating that the master is attempting to read data from this slave.
2. Set I2C_DATACMD[8] to 0 to specify a write operation. At the same time, write the data to be transmitted into I2C_DATACMD[7:0].
3. Read I2C_RAWISR[4] until the read status value is 1, indicating that the data in the transmit buffer has been transmitted (the condition for setting I2C_RAWISR[4] is determined by I2C_TXTL[7:0]).
4. To transmit multiple data bytes, repeat steps 2 and 3.
5. After transmitting all data, the RD_REQ state shall be cleared by setting I2C_CLRRDREQ[0].

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

26.6.5 Slave Receiver

1. Read I2C_RAWISR[2] until the read status value is 1, indicating that the receive buffer has received one data byte (the condition for setting I2C_RAWISR[2] is determined by I2C_RXTL[7:0]).
2. At this point, read the data from I2C_DATACMD[7:0].
3. To read multiple data bytes, repeat steps 1 and 2.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27 Inter-IC Sound (I2S) Interface

27.1 Overview

The Inter-IC Sound Bus (I^2S) is a standardized communication interface developed by Philips for use in many (super) large-scale IC-based systems, especially in many digital stereo audio systems.

Main features:

- Master and slave modes
- Simplex transmitting, simplex receiving, duplex transceiving
- Philips standard, left-justified and right-justified standards, PCM (with short and long frame) standard
- Configurable audio channel length (bits): 16 or 32
- Configurable audio data length (bits): 8, 16, 24 or 32
- Stereo / mono audio data available in non-PCM mode
- The stereo audio data can be transmitted first via left channel or first via right channel.
- The mono audio data can be transmitted via left channel or right channel.
- In non-PCM mode, the polarity of WS is optional.
- In non-PCM mode, the moment of switching between SD and WS is optional on the rising or falling edge of SCK; while in PCM mode, it is fixed on the rising edge of SCK.
- Built-in 8-word TX FIFO and RX FIFO (1 word = 32 bits), each can store 16 pieces of 16-bit data or 8 pieces of 32-bit data.
- MCLK generated by PLL is required when audio block is defined as Master at 256 times the audio sampling rate (Fs), which is typically 8 / 11.025 / 16 / 22.05 / 24 / 32 / 44.1 / 48 / 96 / 192 kHz.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- If interrupt is enabled, the following conditions will trigger an interrupt:
  - TX FIFO with enough space
  - RX FIFO with enough data
  - TX FIFO under-load
  - RX FIFO overflow
- DMA operation

27.2 Pin Description

Table 27-1: I2S Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
I2S1_MCLK	PC6	Input/output	The audio master clock can be generated either from an internal audio PLL or from an external input.
I2S1_WS	PB9, PB12	Input/output	Word select
I2S1_CK	PB10, PB13	Input/output	Bit clock
I2S1_SD	PC3, PB15	Input/output	Audio data input/output
I2S1_EXTSD	PC2, PB14	Input	Full-duplex audio data input
I2S2_MCLK	PC7	Input/output	The audio master clock can be generated either from an internal audio PLL or from an external input.
I2S2_WS	PA4, PA15	Input/output	Word select
I2S2_CK	PB3, PC10	Input/output	Bit clock
I2S2_SD	PB5, PC12	Input/output	Audio data input/output
I2S2_EXTSD	PB4, PC11	Input	Full-duplex audio data input

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.3 Functional Description

27.3.1 TX FIFO

The TX FIFO is an 8-word deep FIFO used to store the data written by CPU or DMA into the "Write Data Register" (I2S_WR).

27.3.2 RX FIFO

The RX FIFO is an 8-word deep FIFO used to store the data received from the external I2S module. The CPU or DMA reads data in RX FIFO through the "Read Data Register" (I2S_RD).

27.3.3 16-bit Channel Control

This module is used to control the 16-bit long audio channels. It sends audio data from the TX FIFO to the external I2S device through a port and receives audio data from the external I2S device to store it in the RX FIFO via another port.

In the I2S protocol, the SCK (serial clock, also known as bit clock) is used to guide the shift register to shift data by one bit during each SCK cycle. The WS signal (word select, also known as left-right clock or LRCLK) indicates which channel of data is being transmitted.

In master mode, the SCK is generated by dividing the MCLK (master clock) from the PLL module. This control module counts the edges of the synchronized SCK and generates the WS signal.

In slave mode, both SCK and WS come from the external I2S master device. This control module counts the edges of the synchronized SCK and adjusts the SCK count based on the edges of the synchronized WS.

When I2S_GCR[9] is set to 1, the transmit operation begins. When the "Transmitter Shift Register" is empty, it fetches data from the TX FIFO and shifts the data to the serial port.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When I2S_GCR[8] is set to 1, the receive operation begins. Input data from the serial port is placed in the "Receiver Shift Register". Once all bits of the data are shifted in, the data is transferred to the RX FIFO.

27.3.4 32-bit Channel Control

This module is used to control the 32-bit long audio channel. It functions similarly to the control logic for 16-bit channels, and the logic is also analogous. The main difference lies in the SCK frequency and some control signals that manage the transmission timing of dual-channel data from TX FIFO or to RX FIFO.

27.3.5 Port Synchronization

In master mode, this module generates the SCK by dividing MCLK from the PLL. To maintain the sampling rate as 1/256 of the MCLK frequency, the SCK frequency is 1/8 of MCLK for 16-bit channels and 1/4 of MCLK for 32-bit channels. For 16-bit channels, the frequency of PCLK must be greater than 3/4 times MCLK. For 32-bit channels, the frequency of PCLK must be greater than 3/2 times MCLK. In slave mode, SCK comes directly from the external I2S device. The frequency of PCLK must be greater than three times SCK.

The following table summarizes the requirements for PCLK frequency in different modes.

Table 27-2: PCLK Frequency Requirements for I2S in Different Modes

Master Mode		
16-bit channel	$f_{SCK} = 1/8 \times f_{MCLK}$	$f_{PCLK} > 3/4 \times f_{MCLK}$
32-bit channel	$f_{SCK} = 1/4 \times f_{MCLK}$	$f_{PCLK} > 3/2 \times f_{MCLK}$
Slave Mode		
$f_{PCLK} > 3 \times f_{SCK}$		

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.3.6 Interface Timing

I2S (Inter-IC Sound) is a bus standard developed by Philips. According to the Philips standard, SD (serial audio data) is delayed by one SCK (serial clock or bit clock) cycle compared to WS (word select, or left-right clock). If SD is shorter than the channel, zeros are padded after SD in the channel. By default, SD toggles on the falling edge of SCK and samples on the rising edge of SCK.

The MSB-justified (left-justified) and LSB-justified (right-justified) standards were developed later. In these standards, SD and WS are transmitted synchronously. If SD is shorter than the channel, in the MSB-justified standard, zeros are padded after SD in the channel; while in the LSB-justified standard, zeros are padded before SD in the channel. By default, SD toggles on the falling edge of SCK and samples on the rising edge of SCK.

In the PCM standard, SD is delayed by one SCK cycle compared to WS. There are two frame synchronization formats in the PCM standard: short frame synchronization and long frame synchronization. In the short frame synchronization format, when the least significant bit of SD (including the zeros padded after the valid bits of SD) is being transmitted, WS remains high for one SCK cycle. In the long frame synchronization format, when the high 13 bits of SD are being transmitted, WS remains high for 13 SCK cycles. Note that WS does not represent word select (left/right clock), because there is no distinction between the left channel and the right channel in the PCM standard. If SD is shorter than the channel, zeros are padded after SD in the channel. Specifically, SD toggles on the rising edge of SCK and samples on the falling edge of SCK, which is not configurable.

Figure 27-1 shows the I2S interface timing for a 16-bit channel, and Figure 27-2 shows the I2S interface timing for a 32-bit channel. Note that “sck_cnt” is an inaccessible internal counter and is not a signal on the I2S port.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

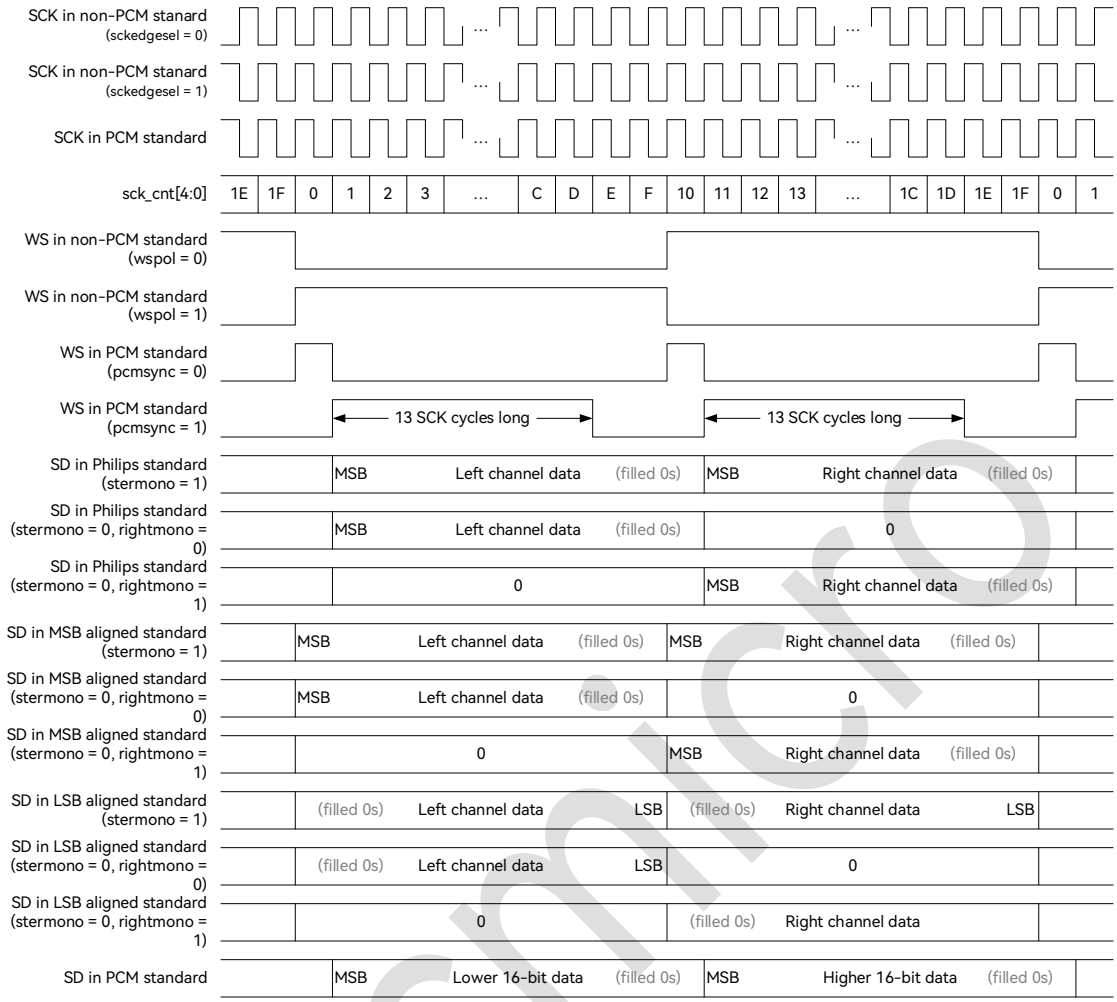

Figure 27-1: I2S Signal Waveform for 16-bit Channel

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

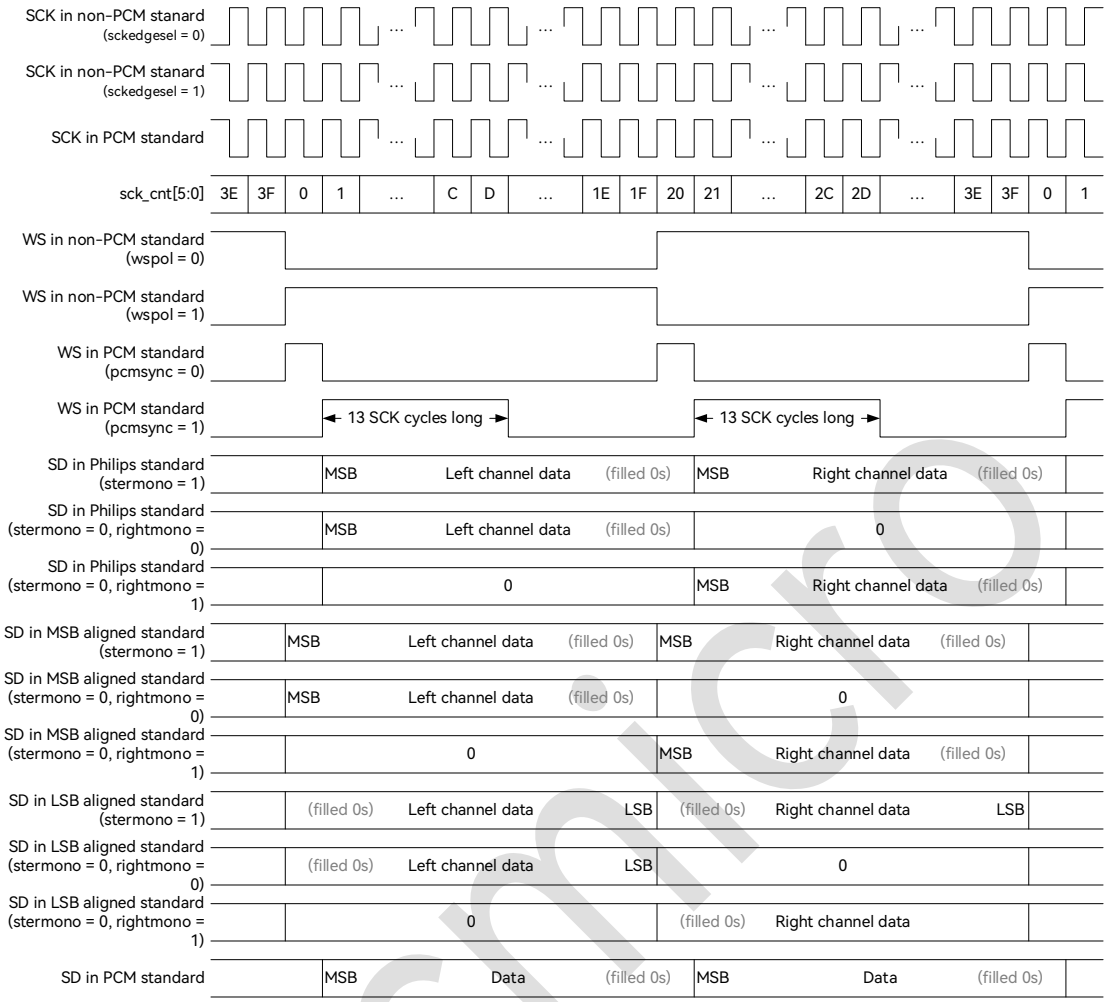

Figure 27-2: I2S Signal Waveform for 32-bit Channel

27.4 Register Description

I2S1 register base address: 0x4700_D000

I2S2 register base address: 0x4700_E000

The registers are listed below:

Table 27-3: List of I2S Registers

Offset Address	Name	Description
0x00	I2S_WR	I2S write data register
0x04	I2S_RD	I2S read data register
0x08	I2S_CSR	I2S current status register
0x0C	I2S_GCR	I2S global control register
0x10	I2S_DFR	I2S data format register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x14	I2S_ISR	I2S interrupt status register
0x18	I2S_IER	I2S interrupt enable register
0x1C	I2S_ICR	I2S interrupt clear register

27.4.1 I2S Write Data Register (I2S_WR)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	I2S_WR	W/R	0x0	Audio data written to TX FIFO: ● For data in a 16-bit channel: One word of data contains two 16-bit mono data (with the 16 low bits transmitted first) or a pair of stereo data (with the 16 low bits transmitted first; by default, the left channel data is transmitted first, but it is possible to configure which data is transferred first via I2S_DFR[8]). ● For data in a 32-bit channel: One word of data contains either a single 32-bit mono data or stereo data.

Note: The data in TXFIFO must be MSB-justified.

1. For data in a 16-bit channel:

   If the data is 8 bits long, it must be stored in bits [31:24] or [15:8], while bits [23:16] and [7:0] are ignored.

2. For data in a 32-bit channel:

   - If the data is 8 bits long, it must be stored in bits [31:24], while bits [23:0] are ignored.
   - If the data is 16 bits long, it must be stored in bits [31:16], while bits [15:0] are ignored.
   - If the data is 24 bits long, it must be stored in bits [31:8], while bits [7:0] are ignored.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.4.2 I2S Read Data Register (I2S_RD)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	I2S_RD	R	0x0	Audio data read from RX FIFO: For data in a 16-bit channel: One word of data contains two 16-bit mono data (with the 16 low bits transmitted first) or a pair of stereo data (with the 16 low bits transmitted first; by default, the left channel data is transmitted first, but it is possible to configure which data is transferred first via I2S_DFR[8] "Right_first"). For data in a 32-bit channel: One word of data contains either a single 32-bit mono data or stereo data.

Note: The data in RX FIFO is always MSB-justified.

1. For data in a 16-bit channel:

   If the data is 8 bits long, it will be stored in bits [31:24] or [15:8], while bits [23:16] and [7:0] will remain zero.

2. For data in a 32-bit channel:

   - If the data is 8 bits long, it will be stored in bits [31:24], while bits [23:0] will remain zero.
   - If the data is 16 bits long, it will be stored in bits [31:16], while bits [15:0] will remain zero.
   - If the data is 24 bits long, it will be stored in bits [31:8], while bits [7:0] will remain zero.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.4.3 I2S Current Status Register (I2S_CSR)

Offset address: 0x08

Reset value: 0x0000 0105

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	RXFIFO_LEVEL	R	0x0	RX FIFO data count: These bits indicate how many words of data are in the RX FIFO. The range is from 0x0 to 0x8.
11	RSV	-	-	Reserved
10	RXFIFO_TRIG	R	0x0	RX FIFO data available status: This bit is set when the RX FIFO has received enough data (reaching the data amount specified by I2S_GCR[1:0]). 0: RX FIFO without enough data 1: RX FIFO with enough data
9	RXFIFO_FULL	R	0x0	RX FIFO full status: 0: RX FIFO not full 1: RX FIFO full
8	RXFIFO_EMPTY	R	0x1	RX FIFO empty status: 0: RX FIFO not empty 1: RX FIFO empty
7:4	TXFIFO_LEVEL	R	0x0	TX FIFO data count: These bits indicate how many words of data are in the TX FIFO. The range is from 0x0 to 0x8.
3	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	TXFIFO_TRIG	R	0x1	TX FIFO availability: This bit is set when the TX FIFO has sufficient empty space (reaching the number of vacancies specified by I2S_GCR[5:4]). 0: TX FIFO has insufficient empty space. 1: TX FIFO has sufficient empty space.
1	TXFIFO_FULL	R	0x0	TX FIFO full status: 0: TX FIFO not full 1: TX FIFO full
0	TXFIFO_EMPTY	R	0x1	TX FIFO empty status: 0: TX FIFO not empty 1: TX FIFO empty

27.4.4 I2S Global Control Register (I2S_GCR)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:15	RSV	-	-	Reserved
15	SD_DIR	W/R	0x0	SD pin direction selection: 0: Input 1: Output In full-duplex mode, the SD pin must be set to output.
14	I2S_EN	W/R	0x0	I2S enable: 0: I2S disabled 1: I2S enabled
13	DMA_MODE	W/R	0x0	DMA access enable: 0: CPU reads/writes TX FIFO and RX FIFO 1: DMA reads/writes TX FIFO and RX FIFO
12	INTEN	W/R	0x0	I2S interrupt enable:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: I2S interrupt disabled 1: I2S interrupt enabled
11	MST_MODE	W/R	0x0	Master/slave mode selection: In master mode, I2S outputs SCK and WS; in slave mode, I2S inputs SCK and WS. 0: Slave mode 1: Master mode
10	FILLDATASEL	W/R	0x0	Data padding selection when TX FIFO is underrun: 0: Transmit empty packet data (all zeros) 1: Transmit the previous data
9	TXEN	W/R	0x0	Transmit control logic and TX FIFO enable: 0: Transmit disabled 1: Transmit enabled
8	RXEN	W/R	0x0	Receive control logic and RX FIFO enable: 0: Receive disabled 1: Receive enabled
7:5	RSV	-	-	Reserved
4	TXFIFO_WTMK	W/R	0x0	TX FIFO watermark (number of triggers) selection: 0: TX FIFO not full (There are 1 words or more empty bits in TX FIFO.) 1: 4 words or less data in TX FIFO (There are 4 words or more empty bits in TX FIFO.)
3:1	RSV	-	-	Reserved
0	RXFIFO_WTMK	W/R	0x0	RX FIFO watermark (number of triggers) selection: 0: RX FIFO not empty 1: 4 words or more data in RX FIFO

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.4.5 I2S Data Format Register (I2S_DFR)

Offset address: 0x10

Reset value: 0x0000 0024

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved
10	PCM_SYNC	W/R	0x0	PCM frame synchronization format selection: This bit is valid only in PCM standard. 0: Short frame synchronization (WS stays high for 1 SCK cycle) 1: Long frame synchronization (WS stays high for 13 SCK cycles)
9	RIGHT_MONO	W/R	0x0	Left/right mono selection: This bit is valid only for mono data. 0: Select left channel data for transmission or reception 1: Select right channel data for transmission or reception
8	RIGHT_FIRST	W/R	0x0	Stereo audio priority channel selection: This bit is used to select which channel of data is transmitted or received first and is valid only for stereo data. 0: Left channel data first 1: Right channel data first

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7	SCK_EDGESEL	W/R	0x0	Selection of SCK edge where data switching occurs: This bit is not valid for the PCM standard. Note: In the PCM standard, SD and WS switch on the rising edge of SCK. 1: SD and WS switch on the rising edge of SCK 0: SD and WS switch on the falling edge of SCK (recommended)
6	WSPOL	W/R	0x0	WS polarity selection: This bit is not valid for the PCM standard, as it does not concern left or right channels; WS is used for frame synchronization. 0: WS low indicates left channel (used in Philips standard) 1: WS high indicates left channel (used in MSB- and LSB-justified standards)
5	STER_MONO	W/R	0x1	Stereo or mono data selection: This bit is not valid for the PCM standard, as it does not concern left or right channels. In fact, the control logic of the PCM standard is designed for stereo data. 0: Mono data 1: Stereo data
4	CHANLEN32	W/R	0x0	Channel length selection: 0: 16 bits 1: 32 bits

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3:2	DATA_LENSEL	R	0x1	Audio data length selection: A 16-bit channel supports 8-bit and 16-bit data, while a 32-bit channel supports all four data types. 0x0: 8 bits 0x1: 16 bits (default) 0x2: 24 bits 0x3: 32 bits
1:0	I2S_STD	W/R	0x0	I2S standard selection: 0X0: Philips standard 0x1: MSB-justified (left-justified) standard 0x2: LSB-justified (right-justified) standard 0x3: PCM standard

27.4.6 I2S Interrupt Status Register (I2S_ISR)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	FRAME_ERR_INTF	R	0x0	Frame error interrupt flag: In slave mode, this bit is set when WS switches at unexpected time. Frame errors may lead to errors and omissions in SD, as if a frame error occurs, the SCK counter will automatically correct itself. However, this sudden jump in the SCK counter may cause many control signal errors. 0: No frame error has occurred. 1: Frame error has occurred.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	UNDERRUN_INTF	R	0x0	TX FIFO underrun error interrupt flag: This bit is set when the TX FIFO is empty after transmit is enabled. The data transmitted can be 0x0 or previously transmitted data. 0: No TX FIFO underrun error has occurred. 1: TX FIFO underrun error has occurred.
2	RXOERR_INTF	R	0x0	RX FIFO overflow error interrupt flag: This bit is set when an attempt is made to write a newly arrived data word into a full RX FIFO. In this case, the newly arrived data will be lost, and the receiving process will continue. 0: No RX FIFO overflow error has occurred. 1: RX FIFO overflow error has occurred.
1	RX_INTF	R	0x0	RX FIFO data sufficient interrupt flag: This bit is set when the RX FIFO has received enough data (reaching the data amount specified by I2S_GCR[1:0]). 0: RX FIFO has insufficient data. 1: RX FIFO has sufficient data.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	TX_INTF	R	0x0	TX FIFO space sufficient interrupt flag: This bit is set when the TX FIFO has sufficient empty space (reaching the number of vacancies specified by I2S_GCR[5:4]). 0: TX FIFO has insufficient empty space. 1: TX FIFO has sufficient empty space.

Note: This interrupt status register is the raw interrupt status register, which cannot be masked by the interrupt enable register but can be cleared by the interrupt clear register.

27.4.7 I2S Interrupt Enable Register (I2S_IER)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	FRAME_ERR_INTEN	W/R	0x0	Frame error interrupt enable: 0: Disabled 1: Enabled
3	UNDERRUN_INTEN	W/R	0x0	TX FIFO underrun error interrupt enable: 0: Disabled 1: Enabled
2	RXOERR_INTEN	W/R	0x0	RX FIFO overflow error interrupt enable: 0: Disabled 1: Enabled
1	RX_INTEN	W/R	0x0	RX FIFO data sufficient interrupt enable: 0: Disabled 1: Enabled

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	TX_INTEN	W/R	0x0	TX FIFO space sufficient interrupt enable: 0: Disabled 1: Enabled

27.4.8 I2S Interrupt Clear Register (I2S_ICR)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	FRAME_ERR_INTCLR	W	0x0	Frame error interrupt clear: 0: Not cleared 1: Cleared
3	UNDERRUN_INTCLR	W	0x0	TX FIFO underrun error interrupt clear: 0: Not cleared 1: Cleared
2	RXOERR_INTCLR	W	0x0	RX FIFO overflow error interrupt clear: 0: Not cleared 1: Cleared
1	RX_INTCLR	W	0x0	RX FIFO data sufficient interrupt clear: 0: Not cleared 1: Cleared
0	TX_INTCLR	W	0x0	RX FIFO space sufficient interrupt clear: 0: Not cleared 1: Cleared

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

27.5 Operation Procedure

1. Configure the I2S_DFR register to select the I2S standard and set the channel length, data format, etc.
2. Configure the I2S_IER register to enable interrupts.
3. Configure the I2S_GCR register to enable I2S and interrupts, but do not enable transmission (I2S_GCR[9]) or reception (I2S_GCR[8]) yet. Also select master/slave mode, CPU or DMA access mode, and FIFO watermark, etc.
4. If DMA access mode is selected, please configure the DMA controller correctly.
5. Set I2S_GCR[15] to choose the SD pin as input or output (must be set as output for full-duplex transmission).
6. Set I2S_GCR[9] / I2S_GCR[8] to start data transmission.
7. If CPU access mode is selected, write the data to be transmitted to the I2S_WR register whenever there is sufficient space in the TX FIFO, and/or read the received data from the I2S_RD register whenever there is enough data in the RXFIFO. The status of TX FIFO and RX FIFO can be checked through the I2S_CSR and I2S_ISR registers, or can be obtained via interrupt requests.
8. If DMA access mode is selected, the DMA controller will write the data to be transmitted to the I2S_WR register via DMA handshake signals, and/or read the received data from the I2S_RD register. The status of TXFIFO and RXFIFO can be checked at any time through the I2S_CSR and I2S_ISR registers.
9. Regardless of whether CPU access mode or DMA access mode is selected, disable I2S by clearing I2S_GCR[14] after the transmission is complete.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28 Universal Asynchronous Receiver Transmitter (UART)

28.1 Overview

Universal asynchronous receiver/transmitter (hereinafter referred to as UART) is a widely used serial communication interface that supports full duplex communication. UART is to send the data transmitted in parallel in memory or processor to the UART receiver of peripherals in series, or to receive the serial data of UART peripherals and convert them into parallel data for the processor. It supports serial communication with external interface devices.

28.2 Main Features

- 16-byte hardware FIFO
- Baud rate supporting integer-N and fractional-N
- CTS / RTS flow control
- Error start bit detection
- Frame interrupt detection
- Circuit-break detection
- Setting of data bit width (5–9 bits) and number of stop bits (1 bit, 1.5 bits and 2 bits)
- Fixed check, parity check or no parity check for data
- IrDA 1.0 protocol with the baud rate ranging from 9.6 k to 115.2 k
- DMA operation

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.3 Pin Description

Table 28-1: UART Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
UART1_TX	PA9, PB6	Output	Transmitting data
UART1_RX	PA10, PB7	Input	Receiving data
UART1_CTS	PA11	Input	Hardware flow control mode transmitting enable signal
UART1_RTS	PA12	Output	Hardware flow control mode transmitting request signal
UART2_TX	PA2, PD5	Output	Transmitting data
UART2_RX	PA3, PD6	Input	Receiving data
UART2_CTS	PA0, PD3	Input	Hardware flow control mode transmitting enable signal
UART2_RTS	PA1, PD4	Output	Hardware flow control mode transmitting request signal
UART3_TX	PB10, PC10, PD8	Output	Transmitting data
UART3_RX	PB11, PC11, PD9	Input	Receiving data
UART3_CTS	PB13, PD11	Input	Hardware flow control mode transmitting enable signal
UART3_RTS	PB14, PD12	Output	Hardware flow control mode transmitting request signal
UART4_TX	PA0, PC10	Output	Transmitting data
UART4_RX	PA1, PC11	Input	Receiving data
UART4_CTS	PA2	Input	Hardware flow control mode transmitting enable signal
UART4_RTS	PA3	Output	Hardware flow control mode transmitting request signal
UART5_TX	PC12	Output	Transmitting data
UART5_RX	PD2	Input	Receiving data
UART6_TX	PC6	Output	Transmitting data
UART6_RX	PC7	Input	Receiving data
UART6_CTS	PB12	Input	Hardware flow control mode transmitting enable signal
UART6_RTS	PB13	Output	Hardware flow control mode transmitting request signal

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.4 Functional Description

The UART can be configured to transmit or receive data at any desired baud rate based on user requirements.

28.4.1 Configurable Baud Rate

UART supports the configuration of any baud rate for data transmission and reception, which is primarily configured by the DLL register, DLH register and DLF register. The calculation process is as follows:

Baud rate is the desired baud rate, f_{clk} is the clock frequency, A is the integer part of $\frac{fclk}{16 * Baud\ rate}$, B is the fractional part of $\frac{fclk}{16 * Baud\ rate}$, then A is configured by the DLH and DLL registers, and B is configured by the DLF register.

28.4.2 UART Transmission Mode

In UART transmit mode, parallel data can be converted into serial data for transmitting. When using UART for data transmission, it is possible to configure the data size, baud rate, whether parity checking is enabled, and the type of parity.

28.4.3 UART Reception Mode

When using the UART for data reception, any baud rate can be configured for data reception, and parity checking can be used to verify if any errors occurred during data transmission.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.4.4 IrDA Mode

UART is compatible with the IrDA 1.0 physical layer protocol, with a maximum transmission baud rate of 115.2K Baud. The data format is fixed as 1 start bit + 8 data bits + 1 stop bit, with no parity bit.

28.5 Register Description

UART1 register base address: 0x4700_F000

UART2 register base address: 0x40B0_3000

UART3 register base address: 0x4600_3000

UART4 register base address: 0x4600_4000

UART5 register base address: 0x4600_5000

UART6 register base address: 0x4600_6000

Table 28-2: List of UART Registers

Offset Address	Name	Description
0x00	UART_RBR	Receive buffer register
0x00	UART_THR	Transmit buffer register
0x00	UART_DLL	Baud rate division low-order register
0x04	UART_DLH	Baud rate division high-order register
0x04	UART_IER	Interrupt enable register
0x08	UART_IIR	Interrupt status register
0x08	UART_FCR	FIFO control register
0x0C	UART_LCR	LINE control register
0x10	UART_MCR	Flow control register
0x14	UART_LSR	LINE status register
0x18	UART_MSR	Flow status register
0x20	UART_LPDLL	Low-power baud rate division low-order register
0x24	UART_LPDLH	Low-power baud rate division high-order register
0x7C	UART_USR	Status register
0x80	UART_TFL	TX FIFO data count register
0x84	UART_RFL	RX FIFO data count register

Unicmicro

Offset Address	Name	Description
0xC0	UART_DLF	Fractional frequency division register
0xC4	UART_RAR	Receive address matching register
0xC8	UART_TAR	Transmit address matching register
0xCC	UART_LCRE	LINE control extension register

Registers are detailed in the following sections.

28.5.1 Receive Buffer Register (UART_RBR)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8:0	RBR	R	0x0	The receive data register can store data received in UART mode or SIR mode. This field serves as the entry for the RX FIFO and can only be accessed when the DLAB bit of the UART_LCR is 0.

28.5.2 Transmit Buffer Register (UART_THR)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8:0	THR	W	0x0	The transmit data register can store data to be transmitted in UART mode or SIR mode. This field serves as the entry for the TX FIFO and can only be accessed when the DLAB bit of the UART_LCR is 0.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.5.3 Baud Rate Parameter Low-order Register (UART_DLL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	DLL	R/W	0x0	Baud rate configuration register (low): This field is accessible only when the DLAB bit of UART_LCR is set to 1. The calculation formula for the integer part of baud rate is: Baud rate = f_{clk} / (16 * {DLH, DLL}) Note: If there is a fractional division, the DLF must be configured before configuring DLL.

28.5.4 Baud Rate Parameter High-order Register (UART_DLH)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	DLH	R/W	0x0	Baud rate configuration register (high): This field is accessible only when the DLAB bit of UART_LCR is set to 1. The calculation formula for the integer part of baud rate is: Baud rate = f_{clk} / (16 * {DLH, DLL}) Note: If there is a fractional division, the DLF must be configured before configuring DLL.

28.5.5 Interrupt Enable Register (UART_IER)

Offset address: 0x04

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	PTIME	R/W	0x0	THRE interrupt enable: this field is only accessible when the DLAB bit of UART_LCR is 0. 1: THRE interrupt enabled 0: THRE interrupt disabled
6:3	RSV	-	-	Reserved
2	ELSI	R/W	0x0	LINE interrupt enable: this field is only accessible when the DLAB bit of UART_LCR is 0. 1: LINE interrupt enabled 0: LINE interrupt disabled
1	ETBEI	R/W	0x0	TX FIFO empty interrupt enable: this field is only accessible when the DLAB bit of UART_LCR is 0. 1: TX FIFO empty interrupt enabled 0: TX FIFO empty interrupt disabled
0	ERBFI	R/W	0x0	Receiving data interrupt enable: this field is only accessible when the DLAB bit of UART_LCR is 0. 1: RX FIFO non-empty interrupt enabled 0: RX FIFO non-empty interrupt disabled

28.5.6 Interrupt Status Register (UART_IIR)

Offset address: 0x08

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:6	FIFOSE	R	0x0	FIFO enable flag: 11: FIFO enabled 00: FIFO disabled
5:4	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3:0	IID	R	0x01	Status ID: 0000: Reserved 0001: No interrupt 0010: TX FIFO empty 0100: RX FIFO not empty 0110: LINE interrupt status 0111: Busy 1100: Timeout status; after enabling FIFO and RX FIFO non-empty interrupts, if there is at least one data in RX FIFO and the CPU has not read the FIFO within 4 UART frames, this field will be set to the timeout interrupt status. Others: Reserved

Note: The interrupt status in this register will be cleared upon reading.

28.5.7 FIFO Control Register (UART_FCR)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:6	RT	W	0x0	Setting of RX FIFO non-empty interrupt: When the data in FIFO is greater than or equal to the data in FIFO status corresponding to this setting, the RX FIFO non-empty interrupt bit is set as: 00: 1 frame of data 01: 4 frames of data 10: 8 frames of data 11: 14 frames of data

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5:4	TET	W	0x0	Setting of TX FIFO empty interrupt: When the data in FIFO is smaller than or equal to the data in FIFO status corresponding to this setting, the TX FIFO empty interrupt bit is set as: 00: FIFO empty 01: 2 frames of data 10: 4 frames of data 11: 8 frames of data
3	RSV	-	-	Reserved
2	XFIFOR	W	0x0	TX FIFO reset bit, which is fixed at 0: 1: Reset TX FIFO 0: Do not reset TX FIFO
1	RFIFOR	W	0x0	RX FIFO reset bit, which is fixed at 0: 1: Reset RX FIFO 0: Do not reset RX FIFO
0	FIFOE	W	0x0	FIFO enable: 1: FIFO enabled 0: FIFO disabled Changing the value of this bit will reset both the RX FIFO and TX FIFO.

28.5.8 LINE Control Register (UART_LCR)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	DLAB	R/W	0x0	UART_DLL and UART_DLH register access setting bit, writable only when UART is idle: 1: UART_DLL and UART_DLH can be accessed at offset addresses 0x0 and 0x4 respectively. 0: UART_RBR / UART_THR can be accessed at offset address 0x0, and UART_IER can be accessed at offset address 0x4.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	BC	R/W	0x0	Break control bit, used to generate a break condition transmitted to the receiving device. In UART mode, it will hold TX low, while in SIR mode, it will continuously send positive pulses on TX. 1: Break enabled 0: Break disabled Note: When there is still data that has not been transmitted, it will not be triggered until the transmission is completed.
5	SEPS	R/W	0x0	Forced setting of parity bit, writable only when UART is idle: 1: When PEN and EPS are both set to 1, the parity bits checked for transmitting and receiving are 0; when PEN is 1 and EPS is 0, the parity bits checked for transmitting and receiving are 1; when PEN is 0, there are no parity bits for transmitting and receiving. 0: The forced setting function of parity bit is disabled.
4	EPS	R/W	0x0	Parity selection bit, writable only when UART is idle: 1: Even parity 0: Odd parity
3	PEN	R/W	0x0	Parity enable bit, writable only when UART is idle: 1: Parity enabled 0: Parity disabled
2	STOP	R/W	0x0	Stop bit length setting, writable only when UART is idle: 1: 1.5 stop bits when DLS = 00; otherwise, 2 stop bits. 0: 1 bit

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1:0	DLS	R/W	0	UART frame data length setting, writable only when UART is idle: 00: 5 bits 01: 6 bits 10: 7 bits 11: 8 bits

28.5.9 Flow Control Register (UART_MCR)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	DMAE	R/W	0	DMA transfer mode selection bit: 0: DMA mode disabled 1: DMA mode enabled Note: To communicate with an external DMA controller, this bit must be set.
6	SIRE	R/W	0	SIR (IrDA) mode selection bit: 0: IrDA SIR mode disabled 1: IrDA SIR mode enabled In this mode, a frame of data consists of 1 start bit, 8 data bits and 1 stop bit, and cannot be modified through the LCR register (when using this mode, the lower four bits of the LCR must not be configured before enabling this bit).
5	AFCE	R/W	0	1: CTS/RTS automatic flow control enabled 0: CTS/RTS automatic flow control disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	LB	R/W	0	In UART mode, the waveform output from the TX pin can be looped back to the same UART RX pin. In SIR mode, the waveform output from the TX pin can be inverted and then looped back to the same UART RX pin. 1: LOOP mode enabled 0: LOOP mode disabled Note: The waveform from TX to RX is internally looped back. In reality, no waveform is actually emitted from the TX pin.
3:2	RSV	-	-	Reserved
1	RTS	R/W	0	RTS interface software control bit: 1: RTS request output being valid 0: RTS request output being invalid Note: When LB mode is enabled (MCR[4] = 1), RTS will become invalid.
0	RSV	-	-	Reserved

28.5.10 LINE Status Register (UART_LSR)

Offset address: 0x14

Reset value: 0x0000 0060

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	ADDR_RCVD	R	0x0	Flag of whether the received data is address or data in 9-bit data mode: 1: The received data is address information. 0: The received data is data information. Read this register and clear it.
7	RFE	R	0x0	RX FIFO error flag: 1: At lease one data in RX FIFO has parity error or UART frame format error 1: No data error in RX FIFO

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				When the erroneous data in RX FIFO is the next data to be read and there are no errors in the other data in RX FIFO, reading this register clears this flag.
6	TEMT	R	0x01	Transmission complete flag: 1: Transmission completed, both TX FIFO and shift register are empty. 0: Transmission not completed
5	THRE	R	0x01	When both Ptime and FIFO are enabled, TX FIFO empty flag: 1: TX FIFO full 0: TX FIFO not full Otherwise, TX FIFO empty flag: 1: TX FIFO empty 0: TX FIFO not empty
4	BI	R	0x0	Break interrupt flag: 1: Break signal received 0: Break signal not received Reading this register or the RBR register will clear this flag.
3	FE	R	0x0	Frame format error flag: 1: Frame format error 0: No frame format error Reading this register or the RBR register will clear this flag.
2	PE	R	0x0	Parity error flag: 1: Parity error 0: No parity error Reading this register or the RBR register will clear this flag.
1	OE	R	0x0	RX FIFO overflow flag: 1: RX FIFO overflow 0: No RX FIFO overflow Reading this register will clear this flag.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	DR	R	0x0	RX FIFO non-empty flag (redundant with the FRNE function in the USR register when FIFO = ENABLE): 1: RX FIFO not empty 0: RX FIFO empty

28.5.11 Flow Status Register (UART_MSR)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	CTS	R	0x0	CTS flag bit: 1: There is CTS request. 0: There is no CTS request.
3:0	RSV	-	-	Reserved

28.5.12 Low-power Baud Rate Division Low-order Register (UART_LPDLL)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	LPDLL	R/W	0x0	In low-power mode, this is the low-order part of the baud rate division register. This field is accessible only when the DLAB bit of UART_LCR is set to 1. This bit is used to configure SIR mode receive detection. The calculation formula is: Low baud rate = f_{clk} / (16 * {LPDLH, LPDLL})

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.5.13 Low-power Baud Rate Division High-order Register (UART_LPDLH)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	LPDLH	R/W	0x0	In low-power mode, this is the high-order part of the baud rate division register. This field is accessible only when the DLAB bit of UART_LCR is set to 1. This bit is used to configure SIR mode receive detection. The calculation formula is: Low baud rate = f_{clk} / (16 * {LPDLH, LPDLL})

28.5.14 Status Register (UART_USR)

Offset address: 0x7C

Reset value: 0x0000 0006

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4	RFF	R	0x0	RX FIFO full flag: 1: RX FIFO full 0: RX FIFO not full
3	RFNE	R	0x0	RX FIFO non-empty flag: 1: RX FIFO not empty 0: RX FIFO empty
2	TFE	R	0x01	TX FIFO empty flag: 1: TX FIFO empty 0: TX FIFO not empty
1	TFNF	R	0x01	TX FIFO non-full flag: 1: TX FIFO not full 0: TX FIFO full

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	BUSY	R	0x0	1: UART is transmitting. 0: UART is idle.

28.5.15 TX FIFO Data Bit Count Register (UART_TFL)

Offset address: 0x80

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4:0	TFL	R	0x0	Data bit count in TX FIFO.

28.5.16 RX FIFO Data Bit Count Register (UART_RFL)

Offset address: 0x84

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4:0	RFL	R	0x0	Data bit count in RX FIFO.

28.5.17 Fractional Frequency Division Register (UART_DLF)

Offset address: 0xC0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:6	RSV	-	-	Reserved
5:0	DLF	R/W	0x0	Fractional frequency division register: the fractional part of baud rate is DLF/64. The calculation formula is: (PCLK% (BAUDRATE * 16)) / BAUDRATE.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

28.5.18 Receive Address Matching Register (UART_RAR)

Offset address: 0xC4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	RAR	R/W	0x0	Receive address matching register

28.5.19 Transmit Address Matching Register (UART_TAR)

Offset address: 0xC8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	TAR	R/W	0x0	Transmit address matching register

28.5.20 LINE Control Extension Register (UART_LCRE)

Offset address: 0xCC

Reset value: 0x00000000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3	TRANSMIT_MODE	R/W	0x0	9-bit transmit mode selection, writable only when UART is idle: 1: TX FIFO is of 9 bits, and the indication flag of address and data come from TX FIFO. 0: TX FIFO is of 8 bits, the transmitted address is determined by the SEND_ADDR and UART_TAR bits, and the transmitted data comes from TX FIFO.

Unicmicro

Bit	Name	Attribute	Reset Value	Description
2	SEND_ADDR	R/W	0x0	Transmit address matching enable bit: 1: In 9-bit mode, when the 9th bit in the UART frame is 1 (address indication), UART will transmit the data in the UART_TAR register. 0: In 9-bit mode, when the 9th bit in the UART frame is 0 (data indication), UART will transmit the data in FIFO. This bit will be automatically cleared after the transmission is completed.
1	ADDR_MATCH	R/W	0x0	Receive data address matching mode enable: 1: Address matching mode enabled 0: Address matching mode disabled This bit is valid only when the DLS_E bit is 1.
0	DLS_E	R/W	0x0	This bit is writable only when UART is idle: 1: 9-bit mode enabled 0: Frame format determined by DLS bit

28.6 Use Process

28.6.1 UART Initialization

1. Enable the clocks for the corresponding GPIO pins and configure the pins for UART_TX and UART_RX alternate functions.
2. Configure the system configuration register for the UART module clock.
3. Configure the UART_DLF register to set the fractional frequency divisor.
4. Set UART_LCR[7] to 1 and configure the UART_DLH and UART_DLL registers to set the UART baud rate.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. Configure the UART_LCR register to set the UART parity, bit length, data frame length, and set Dlab to 0.
6. Configure the UART_FCR register to enable the FIFO.
7. Configure UART_IER register to enable the corresponding UART interrupts.

28.6.2 UART Transmission Process

1. Before transmitting data, the software can configure the baud rate parameters, parity type and data frame format.
2. Set UART_LCR[7] to 0.
3. Write the first data byte to the UART_THR register.
4. Check the transmission complete flag UART_LSR[6]. If UART_LSR[6] = 1, it indicates that the current data has been transmitted successfully.
5. Continue to write the next data byte to USART_THR.

28.6.3 UART Reception Process

1. Before transmitting data, the software can configure the baud rate parameters, parity type and data frame format.
2. For data reception, query the UART_USR flag or wait for an interrupt. If UART_CSR[3] = 1, indicating that the RX FIFO is not empty, then read the data from the UART_RBR register, after which the corresponding flag will be automatically cleared.
3. Error handling during reception: Wait for an interrupt or check the UART_LSR register flags to determine the type of error and perform the corresponding error handling, after which the software clears the error flag.
4. Continue to receive data.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

29 Low-power Universal Asynchronous Receiver Transmitter (LPUART)

29.1 Overview

LPUART is a low-power UART that requires 32-kHz clock to enable UART communications up to 9600 baud/s.

29.2 Main Features

- Asynchronous data transfer
- Standard UART frame format
  - 1-bit start bit
  - Programmable data word length: 7 or 8 bits
  - Odd parity bit, even parity bit or no parity bit
  - Configurable stop bit: 1 bit or 2 bits
- From 300 baud/s to 9600 baud/s using a 32.768 kHz XTL or a 32 kHz RCL clock source
- Programmable data polarity
- Interrupt flags
  - Receive buffer full
  - Receive buffer overflow
  - Receive frame format error
  - Receive parity bit error
  - Start detection

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Data matching
- Flags of transmission completion

- Wakeup on low-power mode
  - Wakeup on RXD falling edge interrupt
  - Wakeup on start bit detection
  - Wakeup on 1-byte data receiving
  - Wakeup on 1-byte data matching

29.3 System Block Diagram

29.3.1 Block Diagram

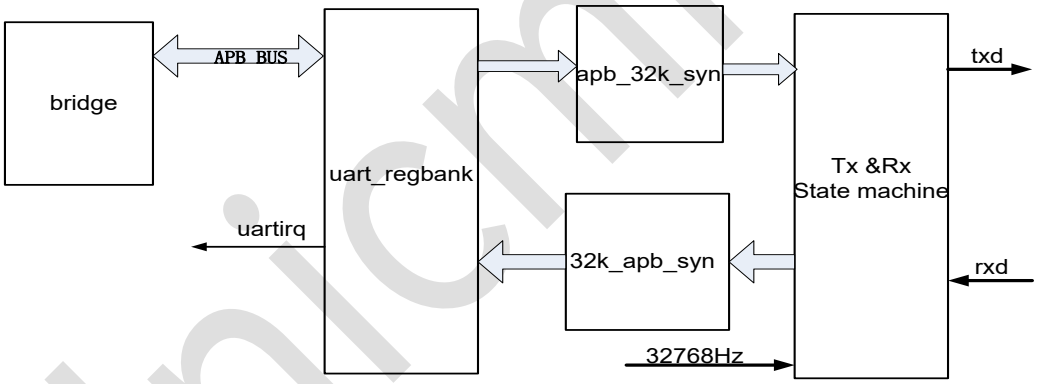

Figure 29-1: LPUART Block Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

29.3.2 Interface Timing

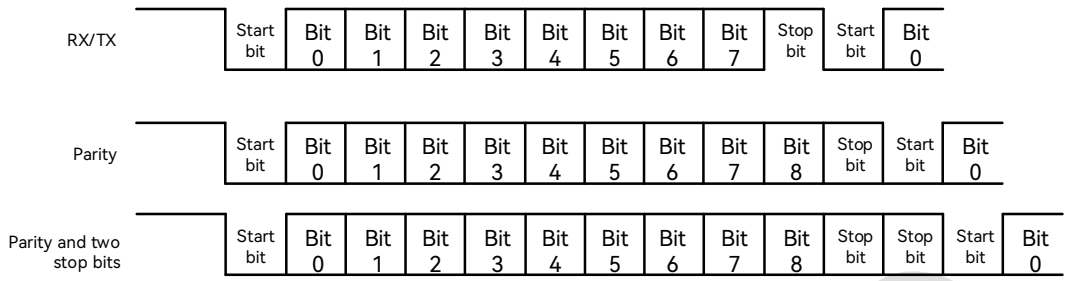

Figure 29-2: LPUART Interface Timing Diagram

29.3.3 Reception Timing

Since the LPUART clock is not an integer multiple of the baud rate, using a fixed division factor would introduce cumulative errors. To ensure accurate sampling, reception alternates between 3 and 4 division factors, ensuring that each bit is sampled once at its midpoint. Whether each bit is divided by 3 or 4 is controlled by the MCTL register. For example:

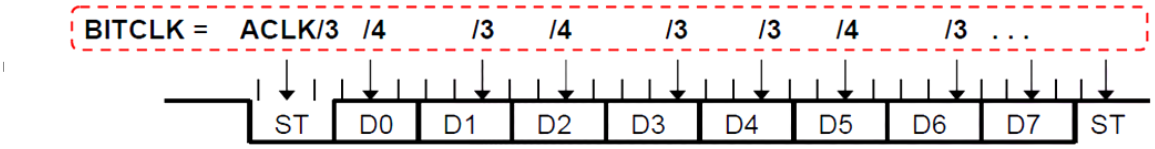

Figure 29-3: LPUART Reception Timing Diagram

29.3.4 Transmission Timing

Similar to LPUART reception, since the LPUART clock is not an integer multiple of the baud rate, using a fixed division factor would also introduce cumulative errors during transmission. Therefore, a 3 and 4 division alternating scheme is used for transmission as well. Whether each bit is divided by 3 or 4 is controlled by the MCTL register. For example:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

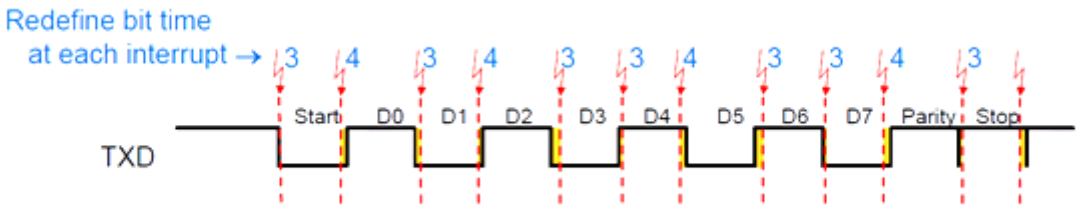

Figure 29-4: LPUART Transmission Timing Diagram

29.4 Pin Description

Table 29-1: LPUART Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
LPUART_TX	PA4, PB11	Output	Transmitting data
LPUART_RX	PC2, PB10	Input	Receiving data

Note: The system can be awakened from standby mode through the PC2 pin. The PB10 pin cannot be used in standby mode.

29.5 Functional Description

29.5.1 Transmission Mode

LPUART can convert parallel data into serial data for transmission. The format of the transmitted data frame can be configured to include 1-bit start bit, 7-bit or 8-bit data, odd parity, even parity, or no parity bit, and 1-bit or 2-bit stop bit. The transmission baud rate ranges from 300 to 9600.

29.5.2 Reception Mode

LPUART can convert serial data into parallel data for reception and supports waking the chip from low-power mode through LPUART receive events. It is capable of receiving data while in Sleep/Stop mode. The transmission baud rate ranges from 300 to 9600 Hz.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

29.6 Register Description

LPUART register base address: 0x40B0_7000

Table 29-2: List of LPUART Registers

Offset Address	Name	Description
0x00	LPUART_RXD	Receive data register
0x04	LPUART_TXD	Transmit data register
0x08	LPUART_STA	Status register
0x0C	LPUART_CON	Control register
0x10	LPUART_IF	Interrupt flag register
0x14	LPUART_BAUD	Baud rate register
0x18	LPUART_EN	RX enable register
0x1C	LPUART_COMPARE	Data matching register
0x20	LPUART_MODU	Baud rate modulation control register

Registers are detailed in the following sections.

29.6.1 Receive Data Register (LPUART_RXD)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	LPURXD	R	0x0	Receive data register

29.6.2 Transmit Data Register (LPUART_TXD)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	LPUTXD	W	0x0	Transmit data register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

29.6.3 Status Register (LPUART_STA)

Offset address: 0x08

Reset value: 0x0000 00C0

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	TC	R	0x1	The transmission complete flag is set when the transmission of a frame of data is completed and the TX buffer is empty. 1: Transmission completed 0: Transmission not completed
6	TXE	R	0x1	TX buffer empty flag, can be set by hardware and automatically cleared by software via writing data to it. 1: Buffer empty 0: Buffer not empty
5	START	R/W	0x0	Start bit detection flag, can be cleared by writing 1.
4	PERR	R/W	0x0	Parity error bit, can be cleared by writing 1.
3	FERR	R/W	0x0	Frame format error bit, can be cleared by writing 1.
2	RXOV	R/W	0x0	Receive buffer overflow bit, can be cleared by writing 1.
1	RXF	R	0x0	Receive buffer full bit, can be cleared by reading the LPUDATA register.
0	MATCH	R/W	0x0	Data matching flag, indicating that the data in the receive buffer matches the compare register, can be cleared by writing 1.

29.6.4 Control Register (LPUART_CON)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	TXPOL	R/W	0x0	Data transmission polarity: 0: Non-inverted 1: Inverted
11	TCIE	R/W	0x0	Transmission complete interrupt enable: 0: Disabled 1: Enabled
10	TXIE	R/W	0x0	TX buffer empty interrupt enable: 0: TX buffer empty interrupt disabled 1: TX buffer empty interrupt enabled
9	RSV	R	0x0	Reserved
8	PAREN	R/W	0x0	Parity bit enable: 0: Data frame without parity bit 1: Data frame with parity bit
7	PTYP	R/W	0x0	Parity type: 0: Even parity 1: Odd parity
6	SL	R/W	0x0	Stop bit length: 0: 1 bit 1: 2 bits
5	DL	R/W	0x0	Data length: 0: 8 bits 1: 7 bits
4	RXPOL	R/W	0x0	Receive polarity: 0: Non-inverted 1: Inverted
3	ERRIE	R/W	0x0	Error interrupt enable: 0: Disabled 1: Enabled
2	RXIE	R/W	0x0	Receive interrupt enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1:0	RXEV	R/W	0x0	Configure to determine under which of the following events to provide the CPU with a receive interrupt: 00: Start bit detects a wake-up 01: 1-byte data reception completed 10: Received data matching succeeded 11: A wake-up detected at falling edge

29.6.5 Interrupt Flag Register (LPUART_IF)

Offset address: 0x10

Reset value: 0x0000 0004

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3	TC_IF	R/W1C	0x0	Transmission complete interrupt flag: 1: Interrupt generated upon transmitting a frame of data 0: No interrupt generated
2	TX_IF	R/W1C	0x1	Transmit buffer empty interrupt flag: 1: Interrupt generated after transmit buffer is empty 0: No interrupt generated
1	RXNEG_IF	R/W1C	0x0	RXD falling edge interrupt flag: 1: Interrupt generated 0: No interrupt generated
0	RX_IF	R/W1C	0x0	Reception complete interrupt flag: 1: Interrupt generated upon receiving a frame of data 0: No interrupt generated

29.6.6 Baud Rate Register (LPUART_BAUD)

Offset address: 0x14

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2:0	BAUD	R/W	0x0	Baud rate (bps): 000: 9600 001: 4800 010: 2400 011: 1200 100: 600 101/110/111: 300

29.6.7 Transmit / Receive Enable Register (LPUART_EN)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1	TXEN	R/W	0x0	Transmit enable: 0: LPUART transmission disabled 1: LPUART transmission enabled After the CPU writes 1 to enable, this register shall be read repeatedly until 1 is read before the next operation.
0	RXEN	R/W	0x0	Receive enable: 0: LPUART reception disabled 1: LPUART reception enabled After the CPU writes 1 to enable, this register shall be read repeatedly until 1 is read before the next operation.

29.6.8 Data Matching Register (LPUART_CMPARE)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	COMPARE	R/W	0x0	Compare the data, if RXEV = 10, the reception complete interrupt will be triggered when the data in the receive buffer matches COMPARE.

29.6.9 Modulation Control Register (LPUART_MODU)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	MCTL	R/W	0x0	Modulation control signal for each bit of LPUART

29.7 Operation Procedure

29.7.1 Initialization

1. Configure the corresponding GPIO pins to be multiplexed as LPUART_TX and LPUART_RX.
2. Configure the PMU function clock control register PMU_FCCR[0] to enable LPUART clock.
3. Configure the baud rate register (LPUBAUD) to set the baud rate.
4. Configure the modulation control register (MCTL) to select the appropriate modulation parameters according to the baud rate.
5. Configure the control register (LPUCON) to set the data frame format, parity, and stop bit length.

29.7.2 Reception Process

1. Configure the receive enable register LPUART_EN[0] = 1 to enable data reception.
2. Configure the control register LPUART_CON[1:0] to enable the receive interrupt.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3. Wait for the receive interrupt to trigger, then read the data from the receive data register (LPURXD).
4. Write 1 to the interrupt flag register LPUART_IF[0] to clear the receive complete interrupt flag.
5. Continue receiving data by repeating steps 2, 3 and 4.

29.7.3 Transmission Process

1. Configure the transmit/receive enable register LPUART_EN[1] = 1 to enable data transmission.
2. Write the data to be transmitted into the transmit data register (LPUTXD).
3. Wait for the status flag register LPUART_STA[6] to be set to 1.
4. Continue transmitting data by repeating steps 2 and 3.

29.7.4 Suggested Configuration for Modulation Control Register

The modulation control register MCTL shall be reasonably configured according to the different communication baud rates. The suggested configuration parameters are as follows:

Table 29-3: MCTL Configuration Parameter Table for LPUART_MODU

Baud	MCTL											
	Bit0 (Start)	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10	Bit 11
9600	0	1	0	0	1	0	1	0	1	0	0	1
4800	1	1	0	1	1	1	1	1	0	1	1	1
2400	1	1	0	1	1	0	1	1	0	1	1	0
1200	0	1	0	0	1	0	0	1	0	0	1	0
600	0	1	1	0	1	0	1	1	0	1	1	0
300	0	1	0	0	0	0	1	0	0	0	0	1

The above parameter table assumes that LPUART operates at an exact 32.768 kHz clock working with RCLP will introduce additional errors and may require fine-tuning the baud rate modulation scheme to obtain better communication results.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30 Universal Synchronous / Asynchronous Receiver Transmitter (USART)

30.1 Overview

USART provides a complete full-duplex universal synchronous / asynchronous serial link. To ensure the highest standard, the data frame format can be programmed in a wide range (data length, parity, number of stop bit, etc.) This receiver implements detection of parity error, frame error and overflow error Receiver timeout allows handling of frames with variable lengths, and transmitter time protection facilitates communication with slow remote devices. Multiprocessor communication can also be supported by address bit processing in reception and transmission.

30.2 Main Features

- Programmable baud rate generator
- 5-bit to 9-bit full-duplex synchronous or asynchronous serial communication
  - Configurable stop bits: 1, 1.5 or 2 in asynchronous mode and 1 or 2 in synchronous mode
  - Parity generation and error detection
  - Detection flags of frame error and overflow error
  - MSB first or LSB first for data transfer
  - Optional open-circuit mark generation and detection

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- 8 or 16 times of oversampling receiver frequency
- Optional hardware handshake RTS – CTS
- Receiver timeout and transmitter time protection
- Optional multiprocessor communication mode with address generation and detection

- IrDA modulation / demodulation
  - Communication rate up to 115.2 kbps
- SPI mode
  - Master or slave
  - Serial clock phase and polarity are software programmable.
  - The frequency of SPI serial clock (SCK) is up to MCK/6 of internal clock frequency.
- LIN mode
  - Complying with LIN1.3 and LIN2.0 protocols
  - Master or slave
  - Handling frames of up to 256 data bytes
  - The response data length can be configured by identifier or defined automatically.
  - Self-synchronization in slave node configuration
  - Automatic processing and verification of “Synch Break” and “Synch Field”
  - “Synch Break” will be detected even if it is partially overlapped with data bytes.
  - Automatic identifier parity calculation, sending and verification
  - Parity sending and verification can be disabled
  - Automatic checksum calculation, sending and verification
  - Checksum sending and verification can be disabled
  - Supporting both “Classic” and “Enhanced” checksum types
  - Complete LIN error checking and reporting
- DMA operation

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.3 Pin Description

Table 30-1: Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
USART7_CTS	PA0, PC0	Input	Hardware flow control mode transmitting enable signal
USART7_RTS	PA1, PC1	Output	Hardware flow control mode transmitting request signal
USART7_CK	PA4, PC4	Input/output	Clock signal
USART7_TX	PA2, PC2	Input/output	Transmitting data
USART7_RX	PA3, PC3	Input/output	Receiving data
USART8_CTS	PB6, PC5	Input	Hardware flow control mode transmitting enable signal
USART8_RTS	PB7, PC9	Output	Hardware flow control mode transmitting request signal
USART8_CK	PB3, PC8	Input/output	Clock signal
USART8_TX	PB4, PC6	Input/output	Transmitting data
USART8_RX	PB5, PC7	Input/output	Receiving data

30.4 Functional Description

30.4.1 Baud Rate Generator

The clock source for the baud rate generator can be selected by configuring the USCLKS field in the mode register (USART_MR):

- Master clock
- Master clock divided by a factor, which is usually 8 unless specified by the product
- External clock, valid on the SCK pin

The baud rate generator is based on a 16 prescaler and is programmed using the CD field in the baud rate generator register (USART_BRGR). If 0 is written to CD, the baud rate generator does not generate any clock. If 1 is written to CD, the prescaler is bypassed.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If the external SCK clock is selected, the duration of the low and high levels of the provided signal must be greater than the master clock (MCK) period. In USART mode, the signal provided on SCK must be at least 4.53 times lower than MCK, or 6 times lower in SPI mode.

30.4.1.1 Asynchronous Mode

In asynchronous mode, the clock is first divided by the CD bits in USART_BRGR. The generated clock is provided as the sampling clock to the receiver, and then divided by 16 or 8 according to the OVER bit in the USART_MR register.

The baud rate is calculated using the following formula:

$$\text{Baud rate} = \frac{\text{Selected clock source}}{(8\ \ (2-Over)\ \ CD)}$$

30.4.1.2 Fractional Baud Rate in Asynchronous Mode

The previously defined baud rate generator is limited by the fact that the output frequency varies only in integer multiples of the reference frequency. One solution to this problem is to integrate a high-resolution fractional N clock generator, where the fractional part is programmed by the FP bit in USART_BRGR. If FP is not 0, the fractional part is valid. This feature is only effective in standard USART mode. The fractional baud rate is calculated using the following formula:

$$\text{Baud rate} = \frac{\text{Selected clock source}}{(8\ \ (2-Over)\quad (CD+\frac{FP}{8})\ \ )}$$

30.4.1.3 Synchronous Mode or SPI Mode

In synchronous mode, the clock is set by the CD bits in the USART_BRGR register:

$$\text{Baud rate} = \frac{\text{Selected clock source}}{CD}$$

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

In synchronous mode, if the external clock (USCLKS = 3) is selected, the clock is provided by the USART SCK pin signal; therefore, no prescaling is needed, and the value in USART_BRGR is invalid. The external clock frequency must be less than 1/3 of the system frequency. For the master in synchronous mode (USCLKS = 0 or 1, CLK0 = 1), the maximum frequency limit for the receiver SCK is MCK / 3.

Whether the external clock or the internal clock divider (MCK/DIV) is selected, if the user wants to guarantee a 50:50 duty cycle for the signal on the SCK pin, the value of the CD field must be set to an even number. If the internal clock MCK is selected, even if the value of the CD field is odd, the baud rate generator will ensure a 50:50 duty cycle on the SCK pin.

30.4.2 Receiver and Transmitter Control

After a reset, the receiver is disabled. The user must enable the receiver by setting the control register USART_CR[4]. However, the receive register can be programmed before the receiver clock is enabled.

After a reset, the transmitter is also disabled. The user must enable the transmitter by setting the control register USART_CR[6]. However, the transmit register can be programmed before the transmitter clock is enabled.

The receiver and transmitter can be enabled together or separately.

At any time, the software can set USART_CR[2] and USART_CR[3] to reset the receiver or transmitter of USART. A software reset clears the status flags and resets the internal state machine, but the user-configured registers remain unchanged. Communication will immediately stop whether it is receiving or transmitting.

The user can also individually disable receiving or transmitting by setting USART_CR[5] and USART_CR[6]. If the receiver is disabled during character reception, the USART will wait until

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the current character reception is complete before stopping reception. If it is disabled during transmission, the USART will wait until the current character and the character stored in the USART_THR register have been transmitted. If time protection is set, it can be processed properly.

30.4.3 Synchronous and Asynchronous Modes

30.4.3.1 Transmitter Operation

The transmitter performs the same operations in both synchronous and asynchronous modes (synchronous = 0 or synchronous = 1). A start bit, up to 9 data bits, an optional parity bit, and up to two stop bits are sequentially output on the TXD pin at the falling edge of the serial clock. The number of data bits is determined by the CHRL field and the MODE9 bit in the USART_MR register. If the MODE9 bit is set, the data bits will be 9 bits regardless of the CHRL field settings. The parity bit is configured by the PAR field in USART_MR and can be set for odd parity, even parity, space parity, mark parity, or no parity. The MSBF field configures which bit is sent first: Writing 1 transmits the most significant bit first, while writing 0 transmits the least significant bit first. The number of stop bits is determined by the NBSTOP field. In asynchronous mode, 1.5 stop bits are supported.

- **Character transmitting**

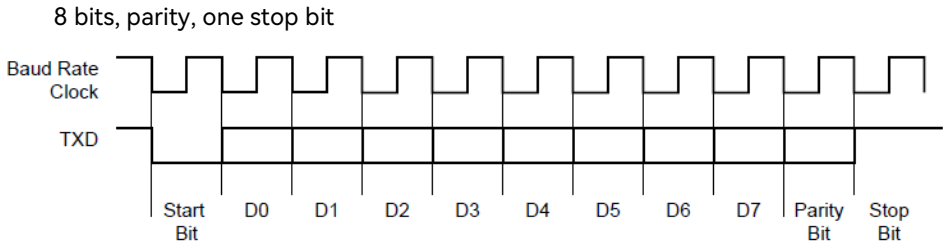

Figure 30-1: USART Character Transmitting

Characters are transmitted by writing to the transmit holding register (USART_THR). The transmitter has two status bits in the channel status register (USART_CSR): TXRDY (transmit

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

ready) indicates that USART_THR is empty, and TXEMPTY indicates that all characters written to USART_THR have been processed. When the current character has been processed, and the last character written to USART_THR is transferred to the transmitter shift register, USART_THR becomes empty, and TXRDY goes high.

When the transmitter is disabled, both TXRDY and TXEMPTY are low. When TXRDY is low, writing characters to USART_THR is invalid, and the data written is lost.

- **Transmitter status**

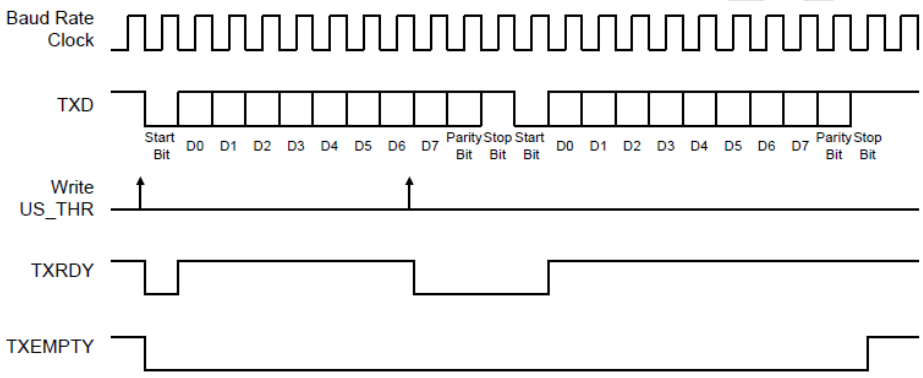

Figure 30-2: USART Transmitter Status

30.4.3.2 Manchester Encoding

When using Manchester encoding, characters transmitted via USART are encoded in the Bi-phase Manchester Encoding II format. To use this mode, the MAN bit field in the USART_MR register must be set to 1. Depending on the polarity configuration, a logic level (0 or 1) is encoded as a transition from 0 to 1 or from 1 to 0 for transmission. Thus, level transitions always occur at the midpoint of each bit time. Although it occupies more (twice as much) bandwidth than the original NRZ signal, it allows for better error control since the expected input must change at half a bit clock time. An example of a Manchester encoding sequence: With the default polarity encoder, the byte 0xB1 or 10110001 would be encoded as 1001101001010110.

- **NRZ code to Manchester code**

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

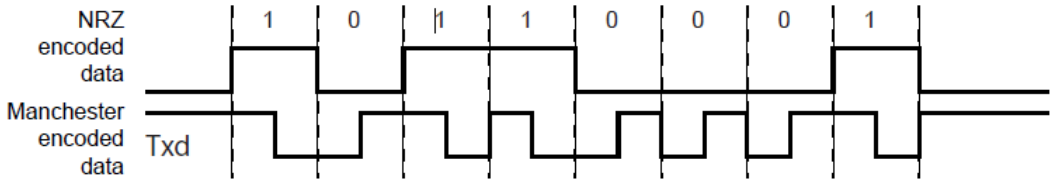

Figure 30-3: NRZ Code to Manchester Code

Manchester encoded characters can also be encapsulated by adding a configurable preamble signal and a frame start delimiter style. Depending on the configuration, the preamble signal is a training sequence composed of a predefined pattern, with a programmable length of 1 to 15 bit times. If the length of the preamble signal is set to 0, no preamble signal waveform will be generated. Various sequences can be selected for the preamble signal mode: ALL_ONE, ALL_ZERO, ONE_ZERO or ZERO_ONE, which are written into the TX_PP field of the USART_MAN register, while TX_PL is used to set the length of the preamble signal. The following figure illustrates and defines the valid modes. For increased flexibility, the encoding scheme can be configured using the TX_MPOL field of the USART_MAN register. If TX_MPOL is set to 0 (default), logic 0 is encoded by a transition from 0 to 1, and logic 1 is encoded by a transition from 1 to 0. If TX_MPOL is set to 1, logic 1 is encoded by a transition from 0 to 1, and logic 0 is encoded by a transition from 1 to 0.

- **Preamble signal mode with default polarity**

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

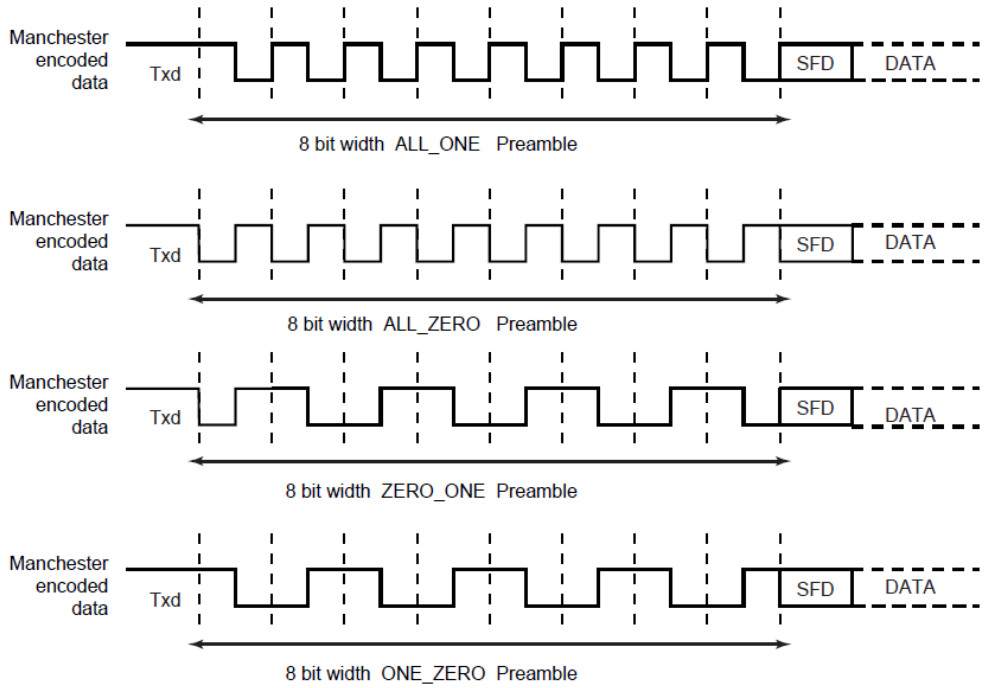

Figure 30-4: Preamble Signal Mode

A frame start delimiter can be configured using the ONEBIT bit in the USART_MR register, consisting of a user-defined pattern to indicate the start of valid data. Figure 30-4 presents these patterns. If the frame start delimiter, i.e., the start bit, is a bit (ONEBIT set to 1), detecting a Manchester encoded logic 0 indicates that a new character is being transmitted on the serial line. If the frame start delimiter is a synchronization pattern, or a synchronization (ONEBIT set to 0) symbol, a new character is considered to start when a sequence of 3 bit times is transmitted serially on the line. When the transition occurs in the middle of the second bit time, the synchronization symbol waveform itself is an invalid Manchester waveform. There are two different synchronization modes: command synchronization symbol and data synchronization symbol. The command synchronization symbol uses a high level to represent a '1' for 1.5 bit times; then it switches to a low level to indicate a second '1' for 1.5 bit times. If the MODSYNC bit in the USART_MR register is set to 1, then the next character will be a command synchronization symbol; if set to 0, the next character will be a data synchronization symbol. When using DMA, the MODSYNC field can be updated by modifying a character in memory. To allow this mode, the VAR_SYNC field in the USART_MR register must be set to 1. This way, the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

MODSYNC in USART_MR is ignored, and the synchronization symbol is configured using the TXSYNH field in USART_THR. The USART character format will be modified to include the synchronization symbol information.

Frame start delimiter

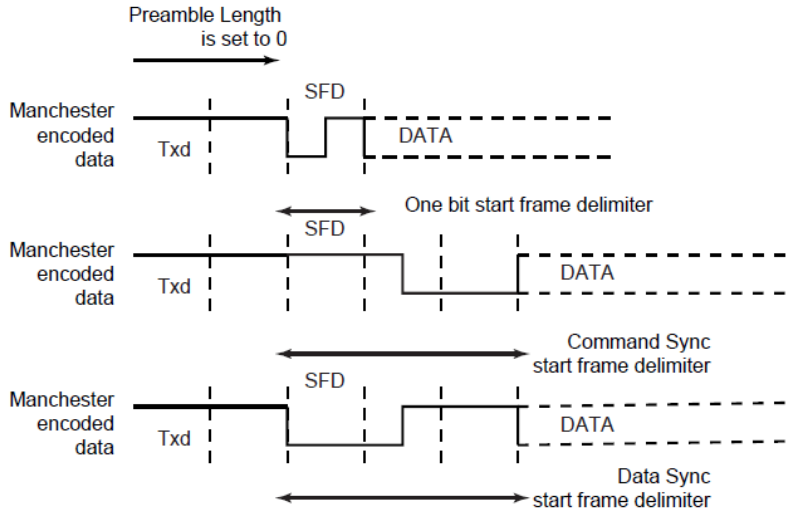

Figure 30-5: Frame Start Delimiter

30.4.3.3 Drift Compensation

Drift compensation is effective only in 16x oversampling mode. A hardware repair system allows greater clock drift. This hardware system can be enabled by setting the USART_MAN bit to 1. If the edge (either rising or falling) of RXD is aligned with the expected edge of the 16x clock cycle, it is considered normal operation with no correction. If a TXD event occurs within 2 to 4 clock cycles before the expected edge, the current cycle is shortened by one clock cycle. If a TXD event occurs within 2 to 3 clock cycles after the expected edge, the current cycle is extended by one clock cycle. These intervals are treated as drift, and corrections are performed automatically.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

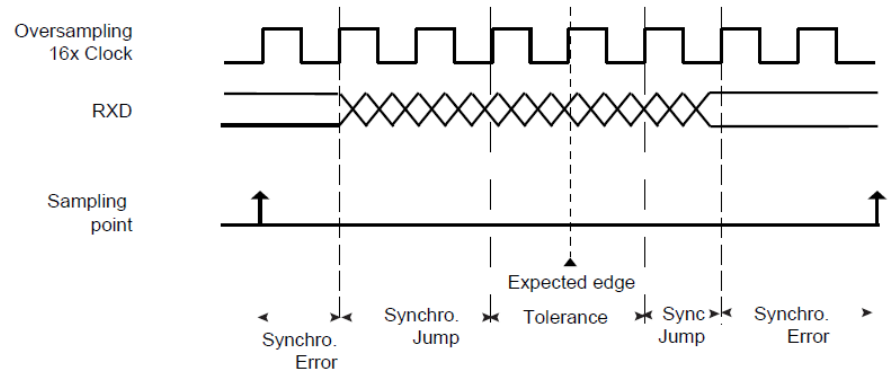

Figure 30-6: Bit Synchronization Diagram

30.4.3.4 Asynchronous Receiver

When the USART operates in asynchronous mode (SYNC = 0), the receiver oversamples the RXD input line. The oversampling frequency is set to either 16 or 8 times the baud rate, as configured by the OVER bit in USART_MR.

The receiver samples the RXD line. If the sample values are all 0 within 1.5 bit time, it indicates that the start bit is detected, after which data bits, parity bits and stop bits are sampled at the bit rate.

If the oversampling frequency is 16 times the baud rate (OVER = 0), a continuous sequence of 8 samples yielding 0 indicates the detection of a start bit. Subsequently, every 16 sampling clock cycles, the subsequent data bits, parity bits and stop bits are sampled in sequence. If the oversampling frequency is 8 times the baud rate (OVER = 1), a continuous sequence of 4 samples yielding 0 indicates the detection of a start bit. Then, the data bits, parity bits and stop bits are sampled in sequence every 8 sampling clock cycles.

The receiver sets the data bit count, first transmitted bit and parity mode fields to be the same as those of the transmitter, specifically CHRL, MODE9, MSBF and PAR. The number of stop bits is irrelevant for the receiver, as it only acknowledges 1 stop bit regardless of the value in the NBSTOP field. Thus, resynchronization can occur between the transmitter and the receiver.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Additionally, after detecting the stop bit, the receiver begins searching for a new start bit, enabling resynchronization even when the transmitter has only 1 stop bit.

The following figures illustrate the start bit detection and character reception when the USART operates in asynchronous mode.

- **Asynchronous start bit detection**

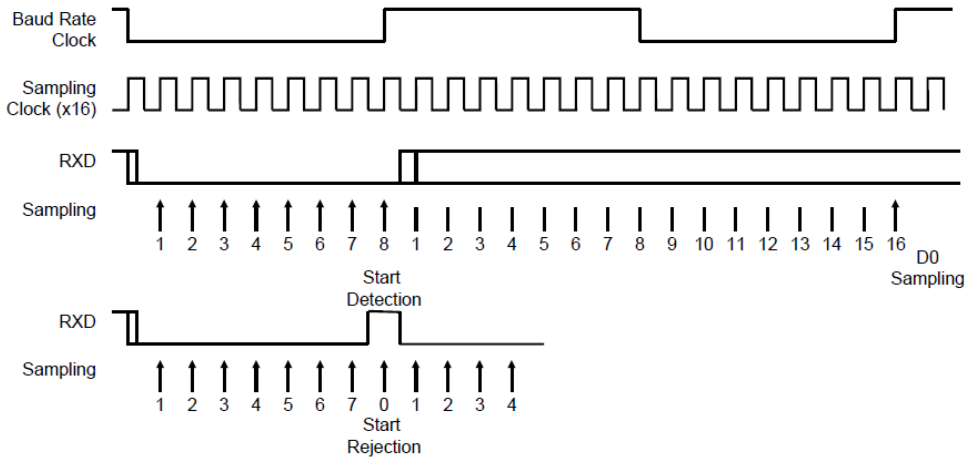

Figure 30-7: Asynchronous Start Bit Detection

- **Character Reception**

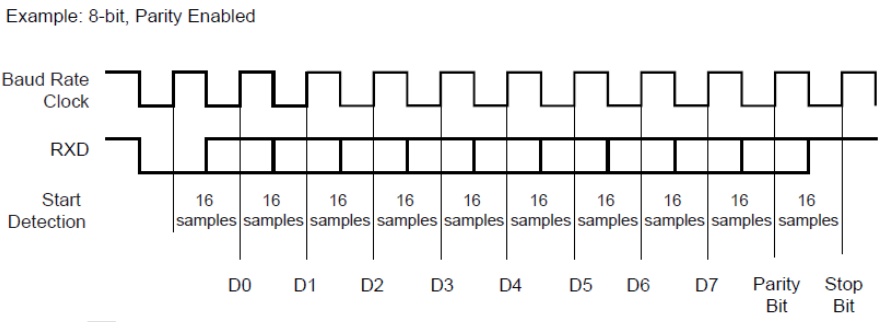

Figure 30-8: Character Reception

30.4.3.5 Manchester Decoder

When the MAN field in the USART_MR register is set to 1, the Manchester decoder is enabled. The decoder performs detection of the preamble signal and frame start delimiter. One input line is specifically designated for the Manchester encoded data input.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

An optional preamble signal sequence can be defined, with its length also being user-defined and completely independent of the transmitter. The length of the preamble signal sequence can be configured by setting the RX_PL bit in the USART_MAN register. If the length is set to 0, preamble signal detection is disabled. Additionally, the input stream polarity can be set through the RX_MPOL field in the USART_MAN register. Depending on application requirements, the preamble signal mode can be defined by the RX_PP field in USART_MAN to match. Unlike the preamble signal, the frame start delimiter is shared in the Manchester encoder/decoder. Therefore, if ONEBIT is set to 1, only 0 Manchester encoding can be detected and treated as a valid frame start delimiter. If ONEBIT is set to 0, only the synchronous mode can be detected and treated as a valid frame start delimiter.

The decoder operates by detecting transitions in the input stream. If RXD is sampled low within 1/4 of a bit time, it is considered the detection of a start bit, as illustrated in the following figure. The sampling pulse rejects device requests.

- **Asynchronous start bit detection**

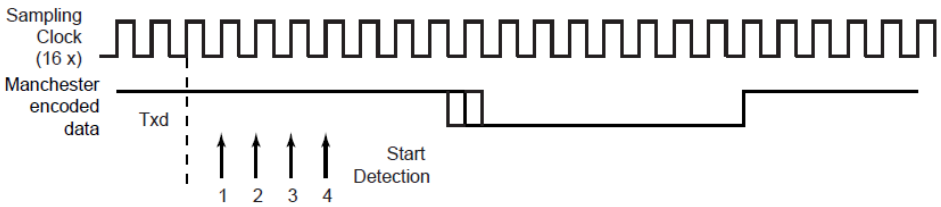

Figure 30-9: Asynchronous Start Bit Detection

The receiver is activated and begins detecting the preamble signal and frame separator, sampling data at 1/4 and 3/4 of the cycle. If a valid preamble signal or frame start delimiter is detected, the receiver will continue decoding using the same synchronization clock. If the data stream does not match a valid mode or valid frame start delimiter, the receiver will resynchronize at the next edge. The minimum time threshold for estimating bit values is 3/4 bit time. If a valid frame start delimiter is detected followed by a valid preamble signal (if used), the input stream will be decoded into NRZ encoded data and sent for USART processing. Figure 30-10 illustrates

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

a case of mismatch in Manchester encoding mode. When the input data stream is sent to the USART, the receiver can also detect violations of Manchester encoding. Violations occur when there is a missing level transition in the bit intervals. In this case, the MANE flag in the USART_CSR register is set to 1. The MANE flag can be cleared by setting the RSTSTA bit in the control register USART_CR to 1.

The following figures depict an example of Manchester error detection during data transmission.

- **Preamble mismatch**

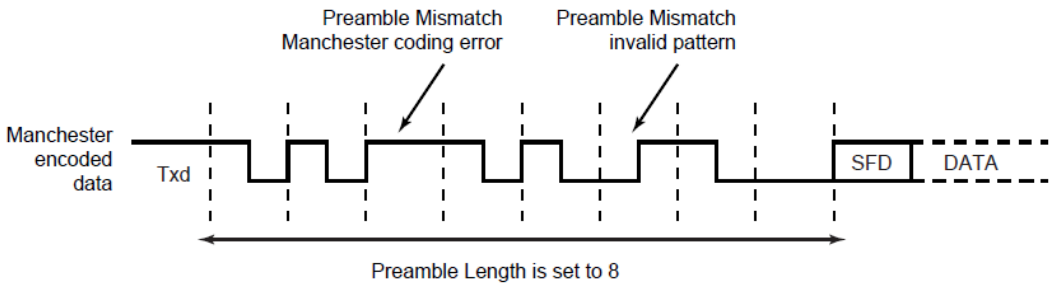

Figure 30-10 : Preamble Mismatch

- **Manchester error flag**

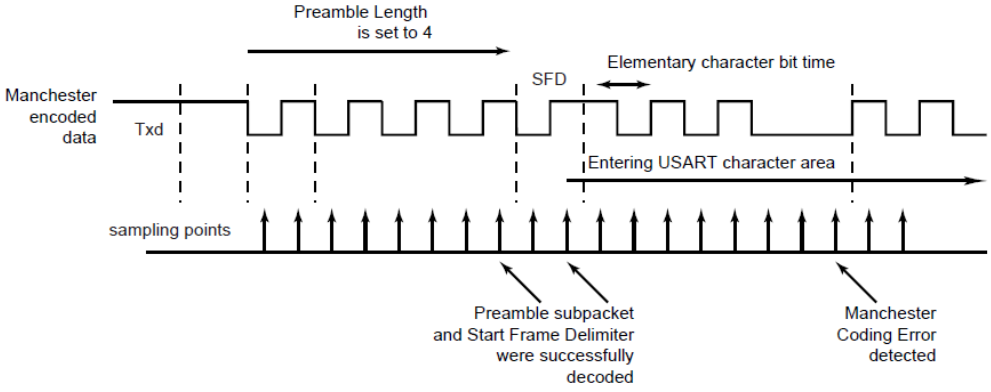

Figure 30-11: Manchester Error Flag

When the frame start delimiter is in synchronous mode (ONEBIT set to 0), command synchronization symbol and data synchronization symbol are supported. If a valid synchronization is detected, the received character is written into the RXCHR field of the USART_RHR register, while RXSYNH is updated. When the received character is a command,

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

RXCHR is set to 1, and when the received character is data, RXCHR is set to 0. This mechanism alleviates and simplifies Direct Memory Access (DMA) since the character already contains its own synchronization field in the same register.

Since the decoder is configured to operate in single-pole mode, the first bit of the frame must be a level transition from 0 to 1.

30.4.3.6 Wireless Interface: Application of Manchester Encoding in USART

This section describes low-data-rate RF transmission systems and their integration with Manchester encoded USART. These systems are based on ICs for transmitters and receivers, supporting ASK and FSK modulation schemes.

The goal of the system is to achieve wireless full-duplex character transmission using two different frequency carriers. All configurations are illustrated in the following figure:

- **Manchester encoded character RF transmission**

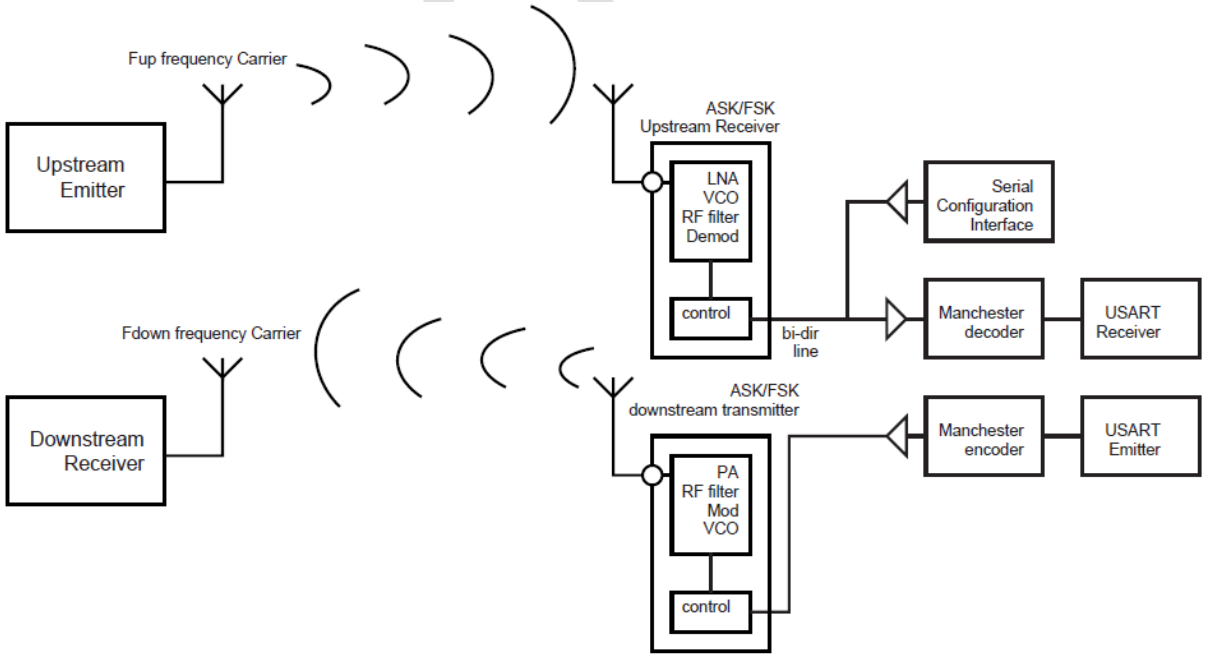

Figure 30-12: Manchester Encoded Character RF Transmission

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The USART module is configured as a Manchester encoder/decoder. In the downlink communication, the Manchester-encoded characters are serially transmitted to the RF transmitter. This may also include user-defined preamble signals and a frame start delimiter. Typically, the preamble signal is used by the RF receiver to distinguish between valid data generated by the transmitter and noise signals. The Manchester data stream is then modulated. The following figure provides an example of an ASK modulation scheme. When the ASK modulator receives a logic high level, the power amplifier (PA) is activated, and an RF signal is transmitted at the downlink frequency. When the ASK modulator receives a logic low (0) level, the RF signal is turned off. If the FSK modulator is activated, it uses two different frequencies to transmit data. When sending a logic 1, the modulator outputs an RF signal at frequency F0; when sending a logic 0, the frequency switches to F1, as shown in the figure below.

For the receiving end, another carrier frequency is used. The RF receiver performs bit detections to detect the demodulated data stream. If a valid pattern is detected, the receiver switches to receiving mode, and the demodulated data stream is sent to the Manchester decoder. Due to bit detection within the RF IC, the amount of data transmitted to the controller can be reduced by a user-defined number of bits. The length of the Manchester preamble signal is defined according to the configuration of the RF IC.

- **ASK modulator output**

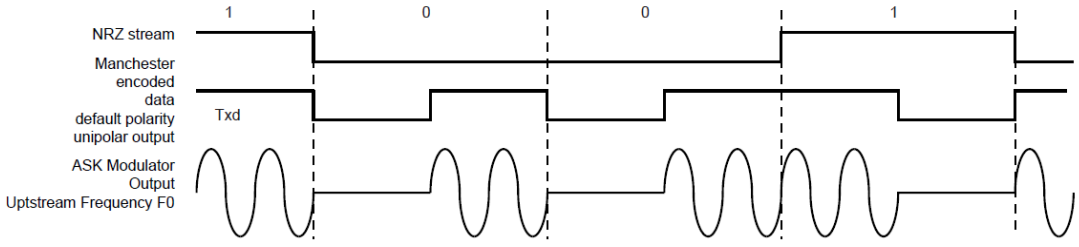

Figure 30-13: ASK Modulator Output

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- **FSK modulator output**

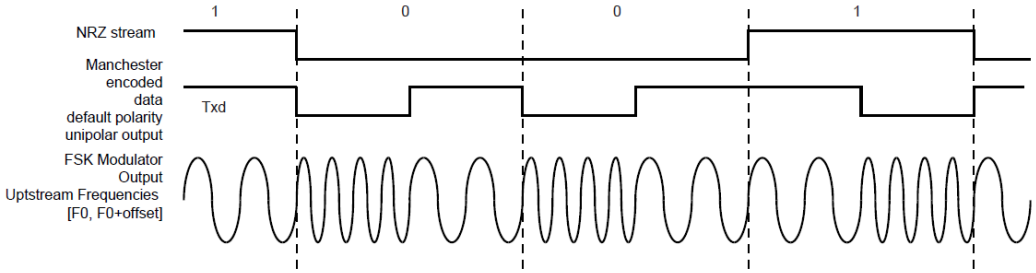

Figure 30-14: FSK Modulator Output

30.4.3.7 Synchronous Receiver

In synchronous mode (SYNC = 1), the receiver samples the RXD signal on the rising edge of each baud rate clock. If a low level is detected, it is determined to be a start bit; subsequently, all data bits, parity bits, and stop bits are sampled, after which the receiver will continue to wait for the next start bit. Synchronous mode provides high-speed transmission capability.

The configuration of fields and bits remains the same as in asynchronous mode. The following figure illustrates the character reception timing in synchronous mode.

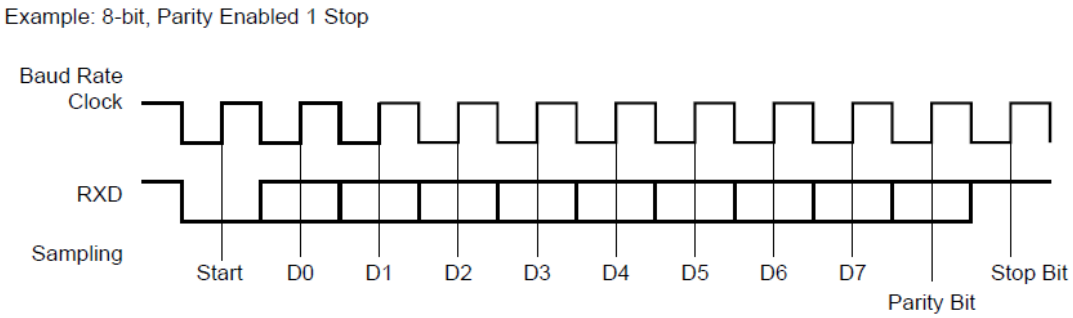

Figure 30-15: Character Reception Timing in Synchronous Mode

30.4.3.8 Receiver Operation

When a character reception is complete, it is transferred to the receive holding register (USART_RHR), and the RXRDY bit in the status register (USART_CSR) is set high. When RXRDY is set, indicating that a character has been received, the OVRE (overflow error) bit is set. The

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

last character is transferred to USART_RHR, overwriting the previous character. Writing a 1 to the RSTSTA (reset status) bit in the control register (USART_CR) can clear the OVRE bit.

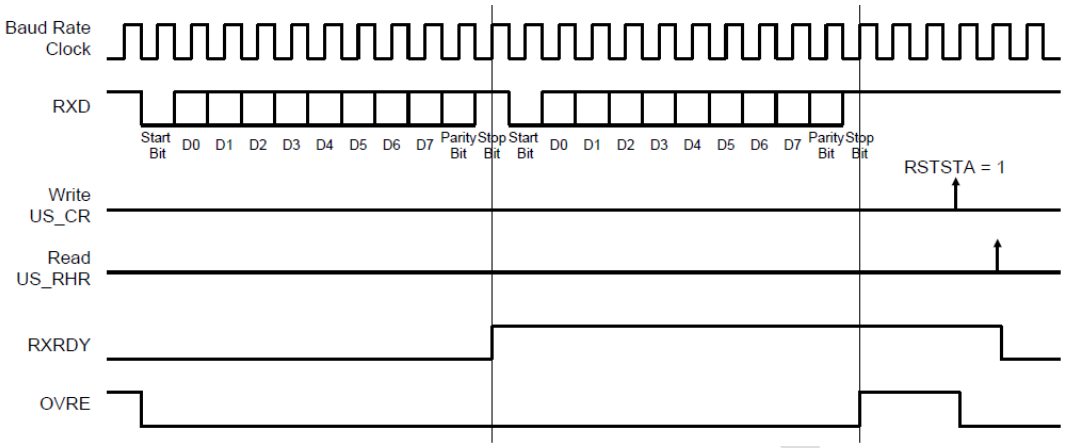

Figure 30-16: Receiver Status

30.4.3.9 Parity

By setting the PAR field in the mode register (USART_MR), the USART can support five different parity modes. The PAR field can also enable multidrop mode, as detailed in Section 30.4.3.10 Multidrop Mode. It supports parity bit generation and error detection.

If even parity is selected, when the number of ones sent by the transmitter is even, the parity bit generated by the parity bit generator is 0; when the number of ones is odd, the parity bit generated is 1. Correspondingly, the receiver parity bit detector counts the received ones, and if the calculated parity bit differs from the sampled parity bit, an even parity error is reported. If odd parity is selected, when the number of ones sent by the transmitter is even, the parity bit generator produces a parity bit of 1; when the number of ones is odd, it produces a parity bit of 0. Correspondingly, the receiver parity bit detector also counts the received ones, and if the calculated parity bit differs from the sampled parity bit, an odd parity error is reported. When using mark parity, the parity bit generated by the parity bit generator is always 1 for all characters; if the sampled parity bit in the receiver is 0, the receiver parity bit detector reports a parity error. When using space parity, the parity bit generated is always 0 for all characters; if

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the sampled parity bit in the receiver is 1, the receiver parity bit detector reports a parity error. If parity is disabled, the transmitter does not generate a parity bit, and the receiver does not report parity errors.

The following table provides examples of the parity bits corresponding to the character 0x41 (ASCII character “A”) under different USART configurations. Since there are two bits set to 1, a “1” is added for odd parity and a “0” for even parity.

Table 30-2: Example of USART Parity Bits

Character	Hexadecimal	Binary	Parity Bit	Parity Mode
A	0x41	0100 0001	1	Odd
A	0x41	0100 0001	0	Even
A	0x41	0100 0001	1	Mark
A	0x41	0100 0001	0	Space
A	0x41	0100 0001	No	No

When the receiver detects a parity error, it sets the PARE (parity error) bit in the channel status register (USART_CSR). Writing a 1 to the RSTSTA bit in the control register (USART_CR) clears the PARE bit. The timing for setting and clearing the parity bit is shown in the diagram below.

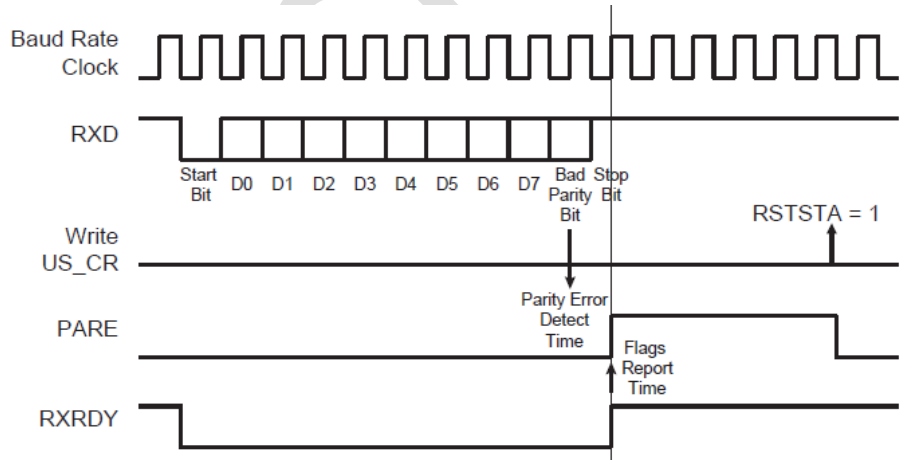

Figure 30-17: Parity Error Timing Diagram

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.4.3.10 Multidrop Mode

If the PAR field in the mode register (USART_MR) is programmed to 0x6 or 0x7, the USART will operate in multidrop mode. This mode distinguishes between data characters and address characters. When the parity bit is 0, data is sent; when the parity bit is 1, an address is sent.

When the USART is configured for multidrop mode, if the parity bit is high, the receiver will set the parity error bit PARE; when the SENDA bit in the control register is set to 1, the transmitter can also send characters when the parity bit is high.

To handle parity errors, writing a 1 to the RSTSTA bit in the control register to clear the PARE bit. When SENDA is written to the USART_CR, the transmitter sends an address byte (parity bit set). In this case, the next byte written to USART_THR will be sent as an address. If the SENDA command is not issued, any character written to USART_THR will be sent normally (with the parity bit set to 0).

30.4.3.11 Transmitter Time Protection

The time protection feature allows USART to connect with slow remote devices.

The time protection function allows an idle state to be inserted between two characters on the transmitter TXD line. This idle state is actually a long stop bit.

The duration of idle state is programmed by the TG field in the transmitter time protection register (USART_TTGR). If the programmed value of this field is zero, no time protection will be generated. Otherwise, after each stop bit is sent, TXD will remain high for the number of bit periods specified in TG.

The behavior of the TXRDY and TXEMPTY status bits can be altered by the time protection, as shown in the diagram below. TXRDY will be set high only after the start bit of the next character is sent. Even if a character has been written to USART_THR, TXRDY will remain 0 during the time

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

protection period. Since the time protection is part of the current transmission, TXEMPTY will remain low until the time protection phase is complete.

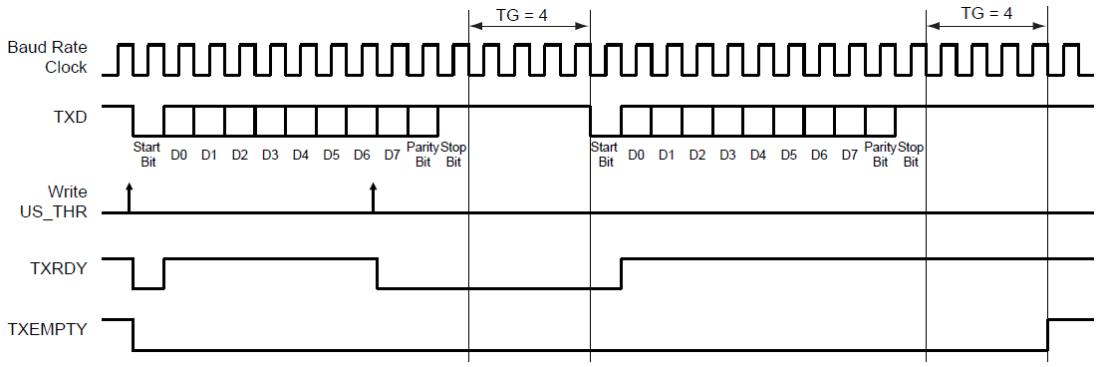

Figure 30-18: Timing Diagram of Time Protection Operation

The maximum time protection periods for the transmitter at different baud rates are shown in the following table.

Table 30-3: Maximum Time Protection Period

Baud Rate (bit/s)	Bit Time (μs)	Time Protection Length
1200	833	212.50
9600	104	26.56
14400	69.4	17.71
19200	52.1	13.28
28800	34.7	8.85
33400	29.9	7.63
56000	17.9	4.55
57600	17.4	4.43
115200	8.7	2.21

30.4.3.12 Receiver Timeout

The receiver timeout supports the handling of variable-length frames. The receiver can detect the idle state on the RXD line; if a timeout is detected, the TIMEOUT bit in the channel status register (USART_CSR) will be set high, generating an interrupt to notify the driver that the frame has ended.

By programming the TO field in the receiver timeout register (USART_RTOR), the timeout delay period (the time the receiver waits for a new character) can be set. If the TO field is set to 0, the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

receiver timeout is disabled, and no timeout will be detected; and the TIMEOUT bit in the USART_CSR register will remain 0. Otherwise, the value of TO is loaded into a 16-bit counter. This counter decrements with each bit period and is reloaded upon the receipt of a new character. If the counter reaches 0, the TIMEOUT bit in the status register will be set high. The user can:

- Stop the counter clock until a new character is received. This can be achieved by writing 1 to the STTTO (start timeout) bit in the control register (USART_CR). In this way, the idle state on the RXD line will not trigger a timeout before the reception of a character, thus avoiding the need to handle an interrupt before receiving a character and allowing waiting for the next idle state on the RXD line after the frame has been received.
- Generate an interrupt if no character is received. This can be accomplished by writing 1 to the RETTO (reload and start timeout) bit in USART_CR. If RETTO is executed, the counter starts counting down from the TO value. The resulting periodic interrupts can be used to handle user-defined timeouts, such as when no key is pressed on the keyboard.

If STTTO is executed, the counter clock stops before the first character is received. The idle state of the RXD before the frame start does not provide a timeout. This prevents periodic interrupts and allows waiting for the frame to end when the RXD line is detected to be idle.

If RETTO is executed, the counter starts counting down from the TO value. The resulting periodic interrupts can be used to handle user-defined timeouts, such as when there is no input from the keyboard.

Table 30-4: Maximum Timeout Periods at Certain Standard Baud Rates

Baud Rate (bit/s)	Bit Time (μs)	Timeout Length (ms)
600	1667	109225
1200	833	54613
2400	417	27306
4800	208	13653
9600	104	6827

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Baud Rate (bit/s)	Bit Time (μs)	Timeout Length (ms)
14400	69	4551
19200	52	3413
28800	35	2276
33400	30	1962
56000	18	1170
57600	17	1138
200000	5	328

30.4.3.13 Frame Error

The receiver can detect frame errors. A frame error is indicated when the stop bit of the received character is detected as 0. Frame errors may occur when the receiver and transmitter are out of synchronization.

Frame errors are represented by the FRAME bit in the USART_CSR register. When a frame error is detected, the FRAME bit is set in the middle of the stop bit time. The error can be cleared by writing 1 to the RSTSTA bit in the USART_CR register.

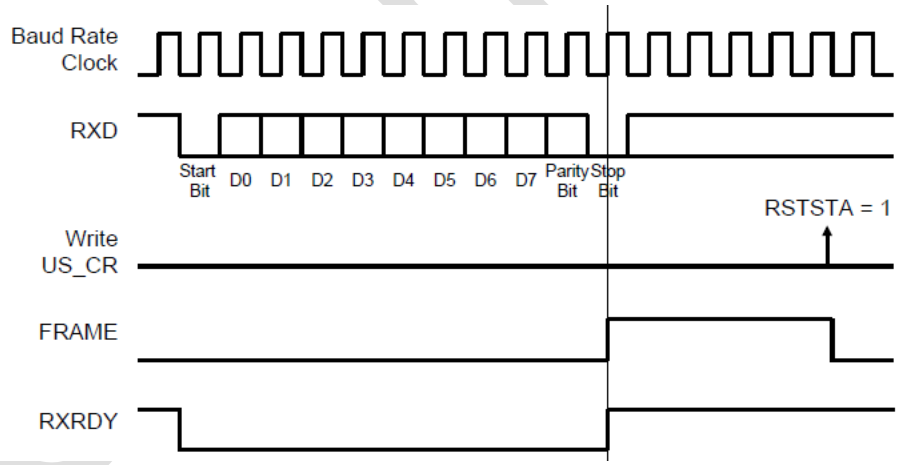

Figure 30-19: Timing Diagram of Frame Error Status

30.4.3.14 Transmission Break

The user can request the transmitter to generate a break condition on the TXD line. This causes the TXD line to be low for at least one complete character time, similar to transmitting a character of 0x00 with both the parity bit and stop bit as 0. Regardless, the transmitter

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

guarantees that the TXD line will be low for one complete character transmission time until the user requests to remove the break condition.

Writing 1 to the STTBRK bit in the USART_CR register will send a break. This can be executed at any time, even if the transmitter is empty (no characters in the shift register or USART_THR) or a character is currently being transmitted. If a break request occurs while a character is being shifted out, the character transmission shall be completed before the TXD line goes low.

Once the STTBRK command is required, other STTBRK commands shall be ignored until the break is completed.

To remove the break condition, write 1 to the STPBRK bit in the USART_CR register. If STPBRK is requested before the end of the minimum break duration (one character, including start, data, parity, and stop bits), the transmitter guarantees that the break condition is completed.

The transmitter treats a break as a character; thus, the STTBRK and STPBRK commands will only be considered when the TXRDY in the USART_CSR register is 1. Similar to processing a character, the TXRDY and TXEMPTY bits will be cleared when the break condition is activated.

Writing 1 to both the STTBRK and STPBRK bits in the USART_CR register will lead to unpredictable results. All STPBRK command requests that do not have preceding STTBRK commands will be ignored. Bytes written to the transmit holding register that have not been initiated will be ignored while a break is pending.

After the break condition, the transmitter will bring the TXD line back high (1) within a maximum of 12 bit times. Therefore, the transmitter ensures that the remote receiver correctly detects the end of the break and the start bit of the next character. If the time guarantee value exceeds 12, the TXD line will remain high during the time guarantee period.

After this period of holding the TXD line high, the transmitter resumes normal operation.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The following figure illustrates the effect of the start break (STTBRK) and stop break (STPBRK) commands on the TXD line.

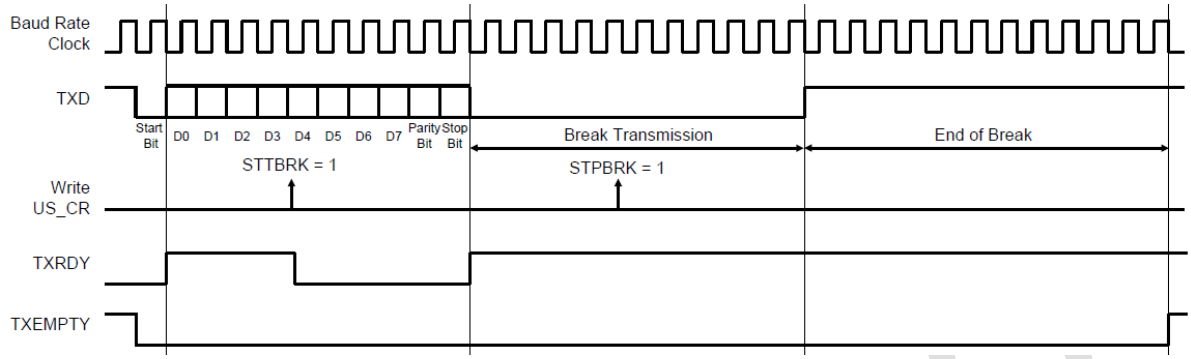

Figure 30-20: Transmission of Break

30.4.3.15 Reception Break

When all received data, parity and stop bits are low, the receiver detects a break condition. This is similar to detecting a frame with data of 0x00 and FRAME as low (indicating a frame error).

Upon detecting a low-level stop bit, the receiver sets the RXBRK bit in the USART_CSR register, which can be cleared by writing 1 to the RSTSTA bit in the USART_CR register.

A break end is detected when at least 2/16 of the bit period is high in asynchronous mode, or when a sample value is high in synchronous mode. The break end can also be achieved by setting the RXBRK bit.

30.4.3.16 Hardware Handshake

The USART can implement out-of-band data flow control through hardware handshake. The RTS and CTS pins are used to connect to remote devices, as shown in the following figure.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

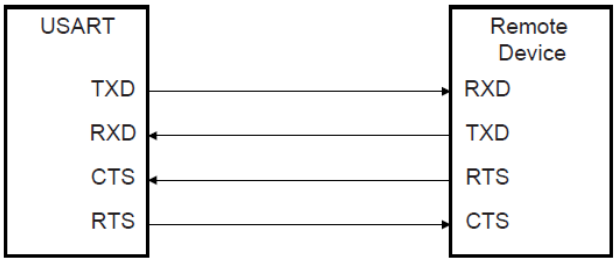

Figure 30-21: Hardware Handshake with Remote Devices

When writing 0x2 to the USART_MODE field in the USART_MR register, the USART will perform hardware handshake.

Except for the changes in the RTS pin level controlled by the receiver and the CTS pin level controlled by the transmitter as described below, the USART operations after enabling hardware handshake remain the same as those in standard synchronous or asynchronous modes. Using this mode requires the use of a PDC channel to receive data. The transmitter can handle hardware handshake under any circumstances.

The following figure illustrates the operation of the receiver when hardware handshake is enabled. If the receiver is disabled and the RXBUFF (receive buffer full) status from the PDC channel is high, the RTS pin is pulled high. Typically, when the CTS pin (driven by RTS) is high, the remote device will not start transmitting. Once the receiver is enabled, RTS goes low, informing the remote device to start transmitting. Defining a new buffer for the PDC and clearing the RXBUFF status bit will cause the RTS pin level to go low.

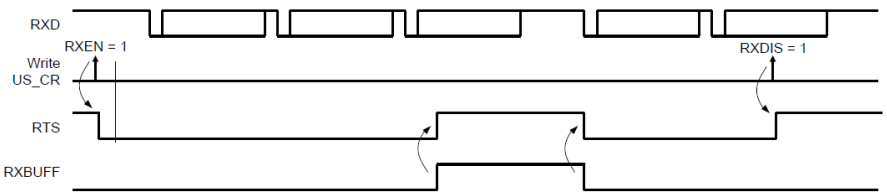

Figure 30-22: Receiver Operation Behavior When Hardware Handshake is Enabled

The following figure illustrates how the transmitter operates when hardware handshake is enabled. The CTS pin disables the transmitter; if a character is being processed, the transmitter

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

will be disabled after the current character is finished; once CTS goes low, the next character transmission begins.

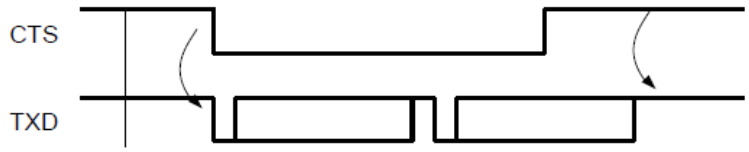

Figure 30-23: Transmitter Operation Behavior When Hardware Handshake is Enabled

30.4.4 IrDA Mode

The IrDA mode of USART supports half-duplex point-to-point wireless communication. It includes a modulator and demodulator for seamless connection with infrared transceivers, as shown in the figure below. The modulator and demodulator are compatible with the IrDA specification version 1.1, supporting data transmission speeds ranging from 2.4 Kb/s to 115.2 Kb/s.

By writing 0x8 to the USART_MODE field in the USART_MR register, the USART IrDA mode can be enabled. The demodulation filter can be configured via the IrDA filter register (USART_IF). The USART transmitter and receiver operate in normal asynchronous mode, and all parameters are accessible. Note that both the modulator and demodulator are activated.

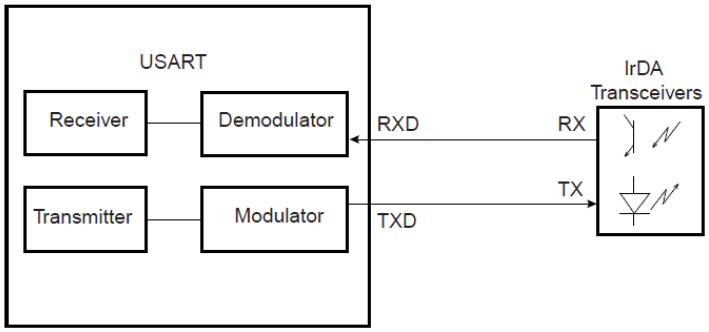

Figure 30-24: Connection with IrDA Transceiver

The receiver and transmitter must be enabled or disabled based on the transmission direction. To receive IrDA signals, the following configurations must be made:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Disable TX and enable RX.
- Configure the TXD port as PIO and set its output to 0 (to avoid LED illumination), disabling the internal pull-up (to reduce power consumption).
- Receive data.

30.4.4.1 IrDA Modulation

When the baud rate is less than or equal to 115.2 Kb/s, the RZI modulation scheme is used. A "0" is represented by a light pulse lasting for 3/16 of a bit period. The table below provides the durations of some signal pulses.

Table 30-5: IrDA Pulse Duration

Baud Rate	Pulse Duration (3/16)
2.4 Kb/s	78.13 μs
9.6 Kb/s	19.53 μs
19.2 Kb/s	9.77 μs
38.4 Kb/s	4.88 μs
57.6 Kb/s	3.26 μs
115.2 Kb/s	1.63 μs

The following figure shows an example of character transmission.

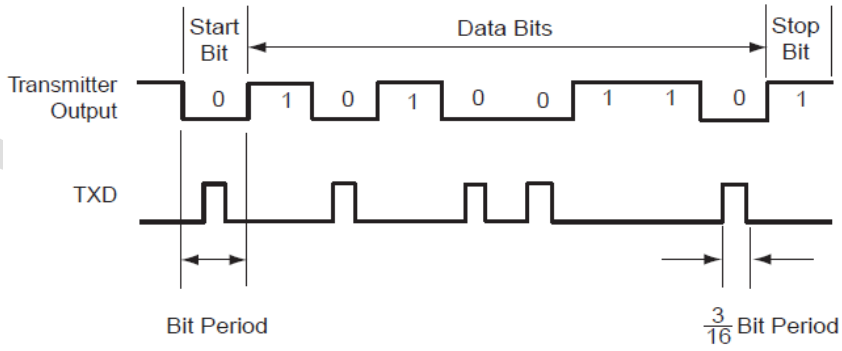

Figure 30-25: IrDA Modulation

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.4.4.2 IrDA Baud Rate

The following table gives some examples of CD values, baud rate errors, and pulse durations. Note that the maximum acceptable error is ±1.87%.

Table 30-6: IrDA Baud Rate Error

Peripheral Clock	Baud Rate	CD	Baud Rate Error	Pulse Time
3686400	115200	2	0.00%	1.63
20000000	115200	11	1.38%	1.63
32768000	115200	18	1.25%	1.63
40000000	115200	22	1.38%	1.63
3686400	57600	4	0.00%	3.26
20000000	57600	22	1.38%	3.26
32768000	57600	36	1.25%	3.26
40000000	57600	43	0.93%	3.26
3686400	38400	6	0.00%	4.88
20000000	38400	33	1.38%	4.88
32768000	38400	53	0.63%	4.88
40000000	38400	65	0.16%	4.88
3686400	19200	12	0.00%	9.77
20000000	19200	65	0.16%	9.77
32768000	19200	107	0.31%	9.77
40000000	19200	130	0.16%	9.77
3686400	9600	24	0.00%	19.53
20000000	9600	130	0.16%	19.53
32768000	9600	213	0.16%	19.53
40000000	9600	260	0.16%	19.53
3686400	2400	96	0.00%	78.13
20000000	2400	521	0.03%	78.13
32768000	2400	853	0.04%	78.13

30.4.4.3 IrDA Demodulator

The demodulator is based on the IrDA receiver filter and contains an 8-bit down-counter whose value is loaded from the USART_IF. When a falling edge is detected on the RXD pin, the filter counter starts counting down at the speed of the master clock (MCK). If a rising edge is detected on the RXD pin, the counter stops and reloads the value from USART_IF. If the counter reaches 0 without detecting a rising edge, the input to the receiver is pulled low within one bit time.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The following figure illustrates the operation of the IrDA demodulator.

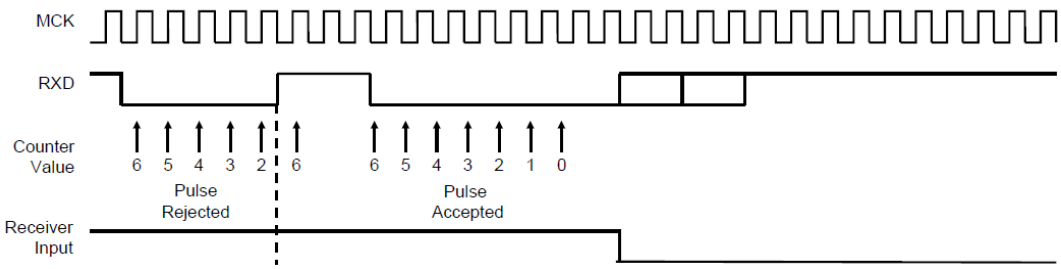

Figure 30-26: IrDA Demodulator Operation

Note that the field value of FI_DI_RATIO in USART_FIDI must be greater than 0 to ensure correct IrDA communication operation.

30.4.5 SPI Mode

The serial peripheral interface (SPI) mode is a synchronous serial data link that allows communication with external devices in either master or slave mode.

It also enables inter-processor communication if an external processor is connected to the system.

The SPI is essentially a shift register that serially transmits data to other SPIs. During data transmission, one SPI system acts as the “master” controlling the data flow, while other SPIs operate as “slaves,” moving data in or out under the control of the master. Different CPUs can take turns acting as the master, and a single master can simultaneously transfer data to multiple slaves (multi-master protocols differ from single-master protocols, where only one CPU remains the master while others act as slaves). However, at any time, only one slave is allowed to write data to the master.

The SPI system selects a slave by the master sending the NSS signal. The USART in master mode of SPI can only connect to one slave because it can only generate one NSS signal.

The SPI system includes two data lines and two control lines:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Master Out Slave In (MOSI): This data line is used for transmitting data from the master to the slave.
- Master In Slave Out (MISO): This data line is used for transmitting data from the slave to the master.
- Serial Clock (SCK): This control line is driven by the master to regulate data flow; the data baud rate can vary during transmission; one bit is transmitted per SCK cycle.
- Slave Select (NSS): This control line is used by the master to select or deselect the slave.

30.4.5.1 Operating Mode

The USART can operate in SPI master mode or SPI slave mode.

To configure the USART to operate in SPI master mode, write 0XE to the USART_MODE bit in the mode register. In this case, the SPI lines must be connected as follows:

- The output pin TXD drives the MOSI line.
- The MISO line drives the input pin RXD.
- The output pin SCK drives the SCK line.
- The output pin RTS drives the NSS line.

To configure the USART to operate in SPI slave mode, write 0XF to the USART_MODE bit in the mode register. In this case, the SPI lines must be connected as follows:

- The MOSI line drives the input pin RXD.
- The output pin TXD drives the MISO line.
- The SCK line drives the input pin SCK.
- The NSS line drives the input pin CTS. To avoid unpredictable behavior, once the SPI mode changes, the transmitter and receiver must be reset in software (in addition to the initialization after a hardware reset).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.4.5.2 Baud Rate

In SPI mode, the baud rate generator operates similarly to that in USART synchronous mode. However, the following constraints must be adhered to:

- SPI master mode:
  - To generate the correct serial clock on the SCK pin, external clock SCK (USCLKS ≠ 0x3) must not be selected, and the CLKO bit in the mode register (USART_MR) must be set to 1.
  - The CD value must be greater than or equal to 4 for the transmitter and receiver to function properly.
  - If the internal clock division (MCK/DIV) is selected, the CD value must be set to an even number to ensure a 50:50 duty cycle on the SCK pin; if the internal clock (MCK) is chosen, the CD value can also be set to an odd number.
- SPI slave mode:
  - External clock (SCK) must be selected; the value of the USCLKS field in the mode register (USART_MR) is invalid; similarly, the value of USART_BRGR is also invalid since the clock is directly provided by the signal on the SCK pin of USART.
  - The frequency of the external clock (SCK) must not exceed 1/4 of the system clock frequency for the transmitter and receiver to operate properly.

30.4.5.3 Data Transmission

Up to 9 bits of data can be continuously transmitted on the TXD pin at the rising or falling edge (depending on CPOL and CPHA settings) of each programmable serial clock, without start bits, parity bits, or stop bits.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The number of data bits can be selected by setting the CHRL bit and the MODE9 bit in the mode register (USART_MR). If 9-bit data is chosen, only the MODE9 bit needs to be set; the CHRL field does not need to be considered.

In SPI mode (whether in master or slave mode), the highest data bit is always transmitted first.

There are four combinations of clock polarity and phase for data transmission. The clock polarity is set by the CPOL bit in the mode register, and the clock phase is set by the CPHA bit. These two parameters determine at which clock edge to drive and sample the data. Each parameter has two states, resulting in four possible combinations. Therefore, a pair of master/slave must use the same parameter pair value to communicate. If multiple slaves are used, and each slave is fixed to a different configuration, the master must be reconfigured when communicating with different slaves.

Table 30-7: SPI Bus Protocol Modes

SPI Bus Protocol Mode	CPOL	CPHA
0	0	1
1	0	0
2	1	1
3	1	0

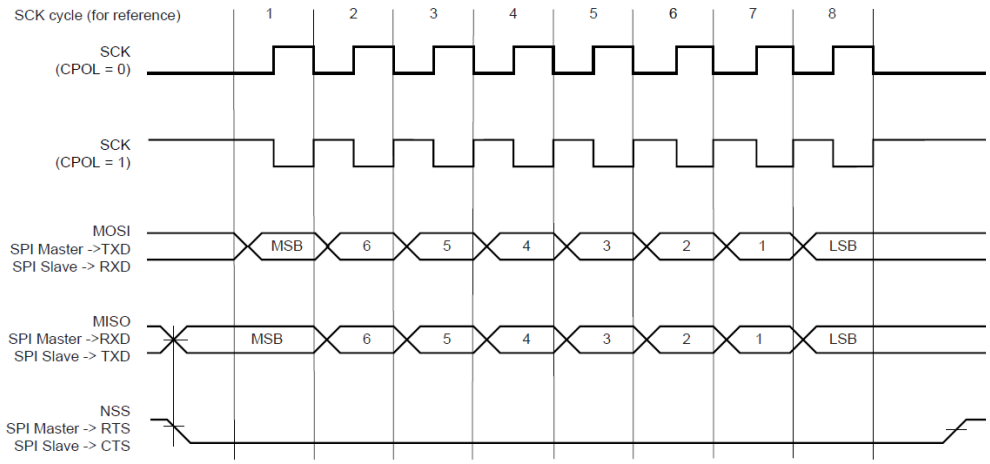

Figure 30-27: SPI Transmission Format (CPHA = 1, 8 bits per transmission)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

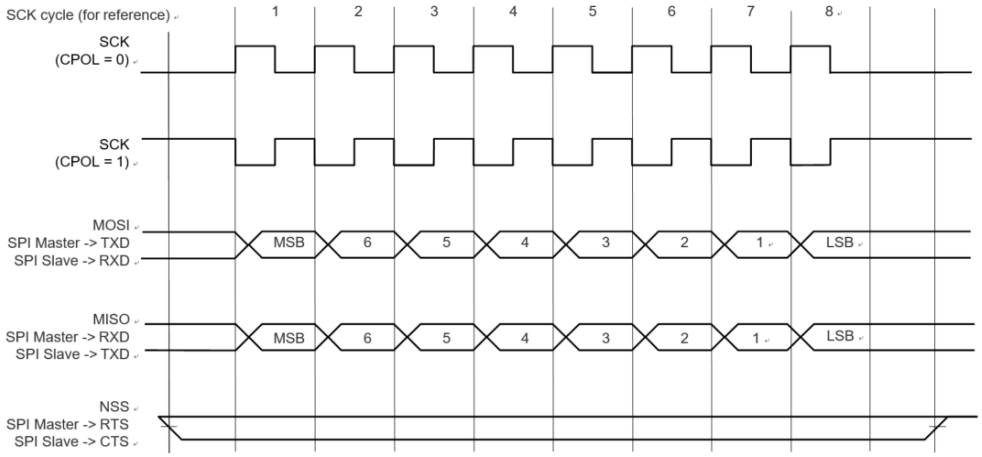

Figure 30-28: SPI Transmission Format (CPHA = 0, 8 bits per transmission)

30.4.5.4 Receiver and Transmitter Control

- **Character Transmission**

Character transmission is performed by writing characters to the transmit holding register (USART_THR). If the USART operates in SPI master mode, additional conditions for transmitting characters can be imposed. When the receiver is not ready (i.e., the character has not been read), setting the INACK bit in the USART mode register (USART_MR) can prevent any character from being transmitted (even if the data has been written to USART_THR). If INACK is set to 0, characters will be sent regardless of the state of the receiver; if INACK is set to 1, the transmitter must wait for the data in the receive holding register to be read before transmitting data (indicated by the RXRDY flag being cleared), thus avoiding any overflow (character loss) in the receiver.

The transmitter has two status bits in the channel status register (USART_CSR): TXRDY (transmit ready) indicates that USART_THR is empty, and TXENPTY indicates that all characters written to USART_THR have been processed. After the current character is processed, the last character written to USART_THR is sent to the shift register of the transmitter, and USART_THR is cleared, then TXRDY is set.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When the transmitter is disabled, both TXRDY and TXENPTY bits are set to 0. When TXRDY is 0, writing characters to USART_THR is ineffective, and the written characters will be lost.

If the USART operates in SPI slave mode, and it is necessary to send a character when the transmit holding register (USART_THR) is empty, the UNRE (buffer data empty error) bit will be set. During this time, the TXD transmission line remains high. The UNRE bit can be cleared by writing 1 to the RSTSTA (reset status) bit in the control register (USART_CR).

In SPI master mode, before transmitting the most significant bit, a low-level signal is issued on the slave select line (NSS) for one bit time; after transmitting the least significant bit, NSS remains high for one bit time. Therefore, the slave select signal is always released between character transmissions, inserting at least a 3-bit time delay. However, to support the CSAAT mode (chip selection activated after transmission) for the slave device, the slave select line (NSS) can be forced low by setting the RTSEN bit in the control register (USART_CR) to 1.

Only by setting the RTSDIS bit in the control register (USART_CR) to 1 can the slave select line (NSS) be released by pulling it high (for example, when all data has been transmitted to the slave device).

In SPI slave mode, the transmitter does not request to initialize character transmission on the falling edge of the slave select line (NSS), but only during the low level. However, the slave select line (NSS) must remain low for at least one bit time before the first serial clock cycle corresponding to the most significant bit.

- **Character Reception**

When a character is fully received, it is transferred to the receive holding register (USART_RHR), and the RXRDY bit in the status register (USART_CSR) is set high.

If a character is received while RXRDY is set, the OVER (overflow error) bit is set. The last character is transferred to USART_RHR, overwriting the current character. Writing 1 to the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

RSTSTA (reset status) bit in the control register (USART_CR) can clear the OVRE bit.

To ensure the normal operation of the receiver in SPI slave mode, the master device must ensure that there is at least a one-bit time delay between transmitting each character in the frame. The receiver does not request to initialize character reception on the falling edge of the slave select line (NSS), but only during the low level. However, the slave select line (NSS) must remain low for at least one bit time before the first serial clock cycle corresponding to the most significant bit.

- **Receiver Timeout**

Since the baud rate clock of the receiver is only available during data transmission in SPI mode, a timeout is not possible in this mode, regardless of the timeout value (TO bit field value) in the timeout register (USART_RTOR).

30.4.6 LIN Mode

The LIN mode provides connections for master nodes and slave nodes on the LIN bus.

LIN (local interconnect network) is a serial communication protocol that effectively supports the control of electromechanical nodes in distributed automotive applications.

The main features of the LIN bus include:

- Single master/multiple slaves concept
- Low-cost implementation based on general UART/SCI interface hardware, equivalent software, or pure state machines
- Slave nodes can achieve self-synchronization without a crystal oscillator or ceramic oscillator.
- Deterministic signal transmission

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Low-cost single-wire implementation
- Transmission rates of up to 20 kbps

LIN provides cost-effective bus communication without the need for CAN bandwidth and multifunctionality.

The LIN mode allows microprocessors to handle LIN frames with minimal actions.

30.4.6.1 Operation Mode

The USART can act as a LIN master node or a LIN slave node.

The node configuration is selected by setting the USART_MODE bit in the USART mode register (USART_MR):

- LIN master node (USART_MODE = 0xA)
- LIN slave node (USART_MODE = 0xB)

To avoid unpredictable behavior, any changes to the LIN node configuration must be made after performing a software reset on the transmitter and receiver (excluding the initialization of node configuration following a hardware reset).

30.4.6.2 Baud Rate Configuration

Refer to the section "30.4.1.1 Asynchronous Mode".

The baud rate is configured in the baud rate generator register (USART_BRGR).

- LIN master node: The baud rate is configured in the baud rate generator register (USART_BRGR).
- LIN slave node: The initial baud rate is configured in the baud rate generator register (USART_BRGR). When writing to USART_BRGR, this configuration is automatically copied

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

to the LIN baud rate register (USART_LINBRR). After the synchronization process is complete, the baud rate is updated in USART_LINBRR.

30.4.6.3 Reception and Transmission Control

Refer to the section "30.4.2 Receiver and Transmitter Control".

30.4.6.4 Character Transmission

Refer to the section "30.4.3.1 Transmitter Operation".

30.4.6.5 Character Reception

Refer to the section "30.4.3.8 Receiver Operation".

30.4.6.6 Message Header Transmission (Master Node Configuration)

All LIN frames start with a message header sent by the master node, which consists of the break field, sync field and identifier field.

Therefore, in the master node configuration, frame processing begins with sending the message header.

Once the identifier is written to the LIN identifier register (USART_LINIR), the message header is transmitted. At the same time, the TXRDY flag goes low.

The break field, sync field and identifier field are automatically sent in sequence.

The break field consists of 13 dominant bits and 1 recessive bit; the sync field is the character 0x55, while the identifier corresponds to the character written to the LIN identifier register (USART_LINIR). The identifier parity bit is automatically calculated and sent.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When the identifier field is transmitted to the shift register of the transmitter, the TXRDY flag goes high.

Once the break field is sent, the LINBK flag in the channel status register (USART_CSR) is set. Similarly, once the identifier field is sent, the LINID flag in USART_CSR is set. These flags can be reset by writing 1 to the RSTSTA bit in the control register (USART_CR).

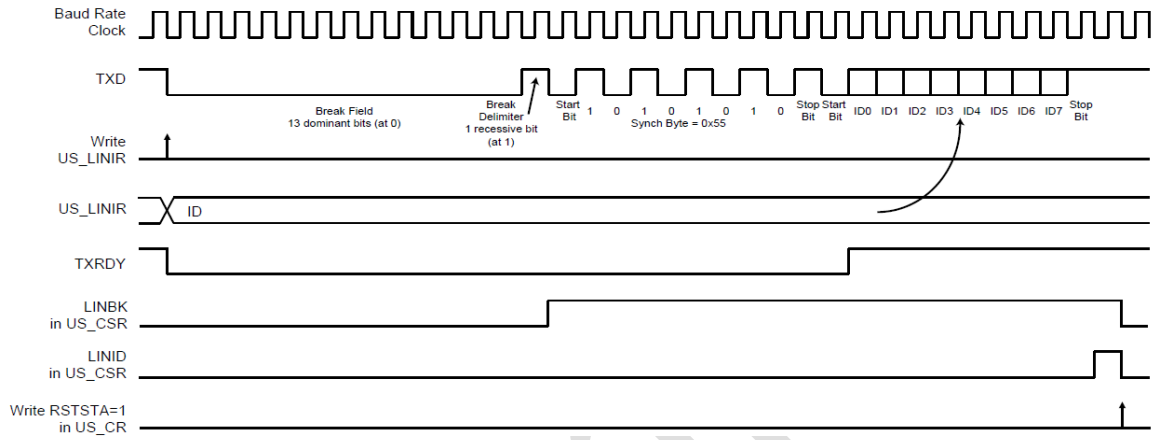

Figure 30-29: Message Transmission

30.4.6.7 Message Header Reception (Slave Node Configuration)

All LIN frames start with a message header sent by the master node, which consists of the break field, sync field and identifier field.

In the slave node configuration, frame processing begins with receiving the message header.

The USART uses an 11-bit break detection threshold at the actual baud rate. At any time, if 11 consecutive recessive bits are detected on the bus, the USART will detect a break field. As long as the break field is not detected, the USART remains idle and does not receive data.

When a break field is detected, the LINBK flag in the channel status register (USART_CSR) is set, and the USART expects the sync field character to be 0x55. This field is used to update the actual baud rate for synchronization. If the received sync character is not 0x55, an inconsistent sync field error is generated.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

After receiving the sync field, the USART expects to receive the identifier field.

When the identifier field is received, the LINID flag in USART_CSR is set. At this point, the IDCHR field in the LIN identifier register (USART_LINIR) is updated with the received character. The identifier parity bit can be automatically calculated and checked.

If the entire message header is not received within the maximum length time specified by $t_{Header_maximum}$, the LINHTE error flag in USART_CSR will be set.

Writing 1 to the RSTSTA bit in the control register (USART_CR) resets the LINID, LINBK and LINHTE flags.

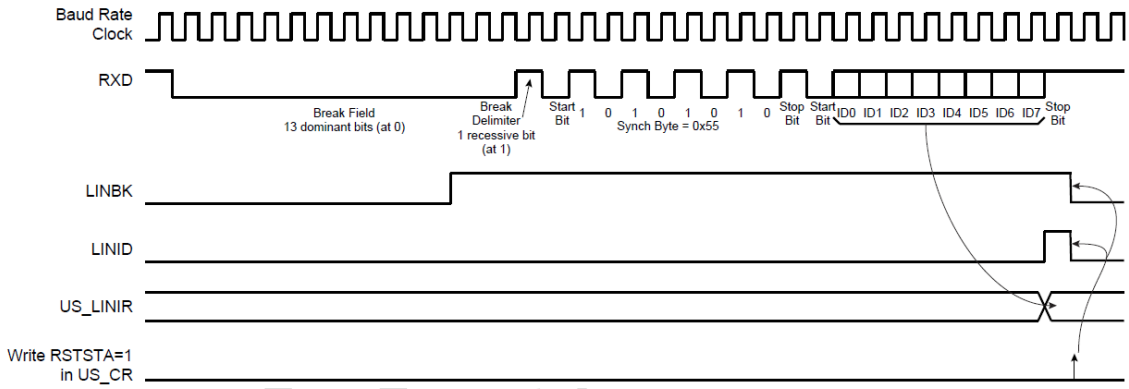

Figure 30-30: Message Reception

30.4.6.8 Slave Node Synchronization

Synchronization is completed solely in the slave node configuration. This step is based on the timing measurement between the falling edges of the sync field. The distances of the falling edges are 2, 4, 6 and 8 bit times.

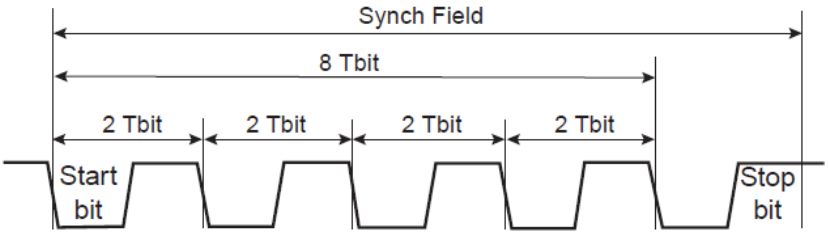

Figure 30-31: Sync Field

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

The timing measurement is performed by a 19-bit counter based on the sampling clock.

When the start bit of the sync field is detected, the counter is reset. The counter then increments during the next 8 Tbits period of the sync field. At the end of these 8 Tbits, the counter stops. At this point, the high 16 bits of the counter (the value divided by 8) are given to the new clock divider (LINCD), and the low 3 bits (the remainder) are provided to the new fractional part (LINFP).

Upon receiving the sync field, the clock divider (CD) and the fractional part (FP) in the baud rate generator register (USART_BRGR) will be updated.

If the sampled synchronization character is not equal to 0x55, the error flag LINISFE in the channel status register (USART_CSR) will be set. Writing 1 to the RSTSTA bit in the control register (USART_CR) to perform a reset.

Once the sync field is fully received, if the SYNCDIS bit in the LIN mode register (USART_LINMR) is not disabled, the clock divider (LINCD) and the fractional part (LINFP) in the LIN baud rate register (USART_LINBRR) will be updated.

After receiving the sync field:

- If the calculated baud rate deviation compared to the initial baud rate is greater than the maximum tolerance FTol_Unsynch (±15%), the clock divider (LINCD) and fractional part (LINFP) will not be updated, and the error flag LINSTE in the channel status register (USART_CSR) will be set.
- If the sampled synchronization character is not equal to 0x55, the clock divider (LINCD) and fractional part (LINFP) will not be updated, and the error flag LINISFE in the channel status register (USART_CSR) will be set.

To reset the LINSTE and LINISFE flags, write 1 to the RSTSTA bit in the control register (USART_CR).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

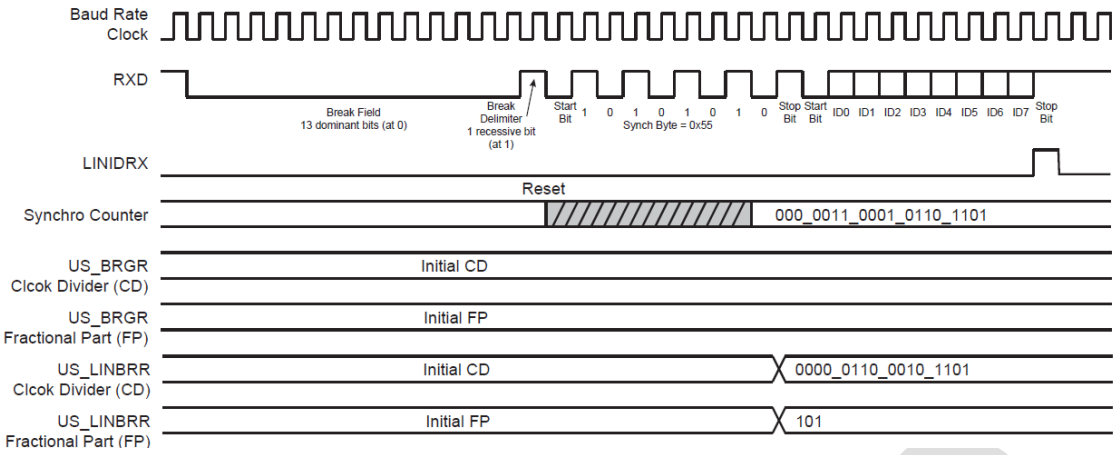

Figure 30-32: Timing Diagram of Slave Node Synchronization

The accuracy of synchronization depends on several parameters:

- Nominal clock frequency (f_{Nom}) (theoretical slave node clock frequency)
- Baud rate
- Oversampling (Over = 0 ≥ 16X or Over = 1 ≥ 8X)

The following formulas are used to calculate the deviation of the slave bit rate relative to the master bit rate (f_{SLAVE} is the actual slave node clock frequency):

$$\text{Baudrate_deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baudrate}}{8 \times f_{SLAVE}}\right)\%$$

$$\text{Baudrate_deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baudrate}}{8 \times \left(\frac{f_{TOL_UNSYNCH}}{100}\right) \times f_{Nom}}\right)\%$$

$f_{TOL_UNSYNCH}$ is the deviation of the actual slave node clock from the nominal clock frequency. The LIN standard specifies that this must not exceed ±15%. It also stipulates that the bit rate difference between two nodes in communication must not exceed ±2%. that the baud rate deviation must not exceed ±1%.

This leads to the minimum value for the nominal clock frequency:

$$f_{Nom}(\text{min}) = \left(100 \times \frac{[0{,}5 \times 8 \times (2 - \text{Over}) + 1] \times \text{Baudrate}}{8 \times \left(\frac{-15}{100} + 1\right) \times 1\%}\right)\text{Hz}$$

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Examples:

- Baud rate = 20 kbps, OVER = 0 (Oversampling 16X) ≥ f_{Nom}(min) = 2.64 MHz
- Baud rate = 20 kbps, OVER = 1 (Oversampling 8X) ≥ f_{Nom}(min) = 1.47 MHz
- Baud rate = 1 kbps, OVER = 0 (Oversampling 16X) ≥ f_{Nom}(min) = 132 kHz
- Baud rate = 1 kbps, OVER = 1 (Oversampling 8X) ≥ f_{Nom}(min) = 74 kHz

30.4.6.9 Identifier Parity Check

The protected identifier consists of two subfields: the identifier and the identifier parity. Bits 5:0 are allocated to the identifier, and bits 7:6 are allocated to the parity.

The USART interface can generate and check these parity bits but can also disable these functions. The user can choose between the two modes using the PARDIS bit in the LIN mode register (USART_LINMR):

- PARDIS = 0:
  - During message header transmission, the parity bit is calculated and sent along with the lower 6 bits of the IDCHR field of the LIN identifier register (USART_LINIR), with bits 6 and 7 discarded.
  - During message header reception, the parity bit of the identifier is checked. If the parity bit is incorrect, an identifier parity error occurs. Only the lower 6 bits of the IDCHR field are updated with the received identifier, while bits 7:6 are fixed at 0.
- PARDIS = 1:
  - During message header transmission, all bits of the IDCHR field in the LIN identifier register (USART_LINIR) are sent on the bus.
  - During message header reception, all bits of the IDCHR field are updated with the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

received identifier.

30.4.6.10 Node Operation

The relevance of the node in the identifier function is related to LIN responses. Therefore, the USART must be configured after sending or receiving an identifier. There are three configurations:

- PUBLISH: The node sends a response.
- SUBSCRIBE: The node receives a response.
- IGNORE: The node is irrelevant to the response; it neither sends nor receives a response.

This is configured by the node operation (NACT) field in the USART_LINMR register. For example, a LIN group contains one master and two slaves:

- Data transmitted from the master to slaves 1 and 2:
  - NACT (master) = PUBLISH
  - NACT (master) = SUBSCRIBE
  - NACT (master) = SUBSCRIBE
- Data transmitted from the master to slave 1:
  - NACT (master) = PUBLISH
  - NACT (master) = SUBSCRIBE
  - NACT (master) = IGNORE
- Data transmitted from slave 1 to the master:
  - NACT (master) = PUBLISH
  - NACT (master) = SUBSCRIBE
  - NACT (master) = IGNORE
- Data transmitted from slave 1 to slave 2:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- NACT (master) = IGNORE
- NACT (master) = PUBLISH
- NACT (master) = SUBSCRIBE

- Data transmitted from slave 2 to the master and slave 1:
  - NACT (master) = SUBSCRIBE
  - NACT (master) = SUBSCRIBE
  - NACT (master) = PUBLISH

30.4.6.11 Response Data Length

The LIN response data length is the number of data field bytes in the response, excluding the checksum.

The response data length can be configured by the user or automatically defined by bits 5:4 of the identifier (compatible with LIN specification 1.1). Users can select between these two modes using the DLM bit in the LIN mode register (USART_LINMR):

- DLM = 0: The response data length is configured by the user through the DLC field of USART_LINMR. The response data length equals (DLC + 1) bytes. The DLC setting range is from 0 to 255, so the response can contain from 1 data byte to 256 data bytes.
- DLM = 1: The response data length is defined by the identifier (IDCHR in USART_LINIR) according to the following table. The DLC field of USART_LINMR is discarded. The response can contain 2, 4 or 8 data bytes.

Table 30-8: Response Data Length with DLM = 1

IDCHR[5]	IDCHR[4]	Response Data Length (Bytes)
0	0	2
0	1	2
1	0	4
1	1	8

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

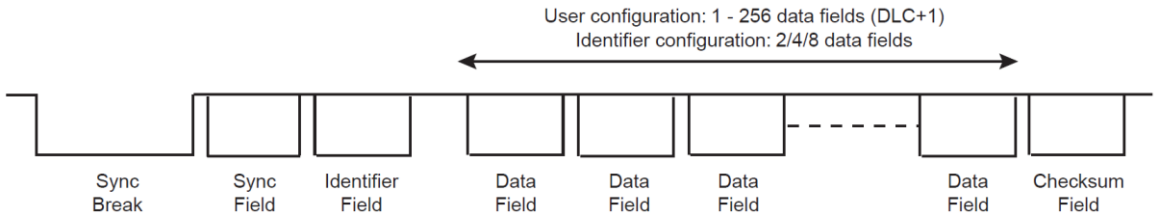

Figure 30-33: Response Data Length Diagram

30.4.6.12 Checksum

The last field of the frame is the checksum. The checksum consists of the carry-inclusive inverted 8-bit sum of either all data bytes or all data bytes along with the protected identifier. The checksum calculated only for the data bytes is referred to as a classic checksum, used for communication with LIN 1.3 slaves. The checksum that includes both data bytes and the protected identifier byte is referred to as an enhanced checksum, used for communication with LIN 2.0 slaves.

The USART can be configured to:

- Automatically send and check the enhanced checksum (CHKDIS = 0 & CHKTYP = 0)
- Automatically send and check the classic checksum (CHKDIS = 0 & CHKTYP = 0)
- Not send and check the checksum (CHKDIS = 1)

This configuration is controlled by the checksum type (CHKTYP) and checksum disable (CHKDIS) fields in the LIN mode register (USART_LINMR).

If the checksum feature is disabled, users can manually send it by treating the checksum as a normal data byte and adding 1 to the response data length.

30.4.6.13 Frame Slot Mode

This mode is only useful for the master node. It adheres to the following rules: each frame slot length must be greater than or equal to $t_{Frame_Maximum}$.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If the frame slot mode is enabled (FSDIS = 0) and the frame transmission is completed, the TXRDY flag will only be reset after $t_{Frame_Maximum}$. Therefore, if the duration of the previous frame slot is less than $t_{Frame_Maximum}$, the master node cannot send a new header.

If the frame slot mode is disabled (FSDIS = 1), the TXRDY flag will be immediately reset upon completion of frame transmission.

The $t_{Frame_Maximum}$ is calculated as follows:

- If the checksum is sent (CHKDIS = 0):
  - $t_{Header_Nominal} = 34 \times t_{bit}$
  - $t_{Response_Nominal} = 10 \times (NData + 1) \times t_{bit}$
  - $t_{Frame_Maximum} = 1.4 \times (t_{Header_Nominal} + t_{Response_Nominal} + 1)$*
  - $t_{Frame_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1 + 1) + 1) \times t_{bit}$
  - $t_{Frame_Maximum} = (77 + 14 \times DLC) \times t_{bit}$
- If the checksum is not sent (CHKDIS = 1):
  - $t_{Header_Nominal} = 34 \times t_{bit}$
  - $t_{Response_Nominal} = 10 \times NData \times t_{bit}$
  - $t_{Frame_Maximum} = 1.4 \times (t_{Header_Nominal} + t_{Response_Nominal} + 1)$*
  - $t_{Frame_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1) + 1) \times t_{bit}$
  - $t_{Frame_Maximum} = (63 + 14 \times DLC) \times t_{bit}$

Note*: The “+1” ensures that the result of $t_{Frame_Maximum}$ is an integer (LIN specification 1.3).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

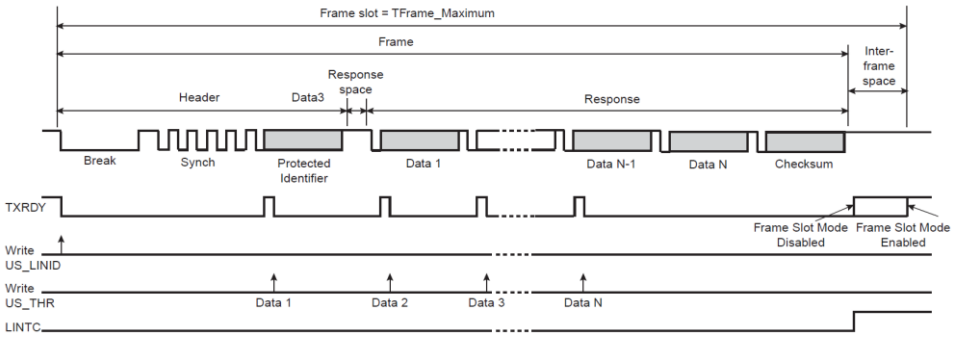

Figure 30-34: USART Frame Slot Mode

30.4.7 LIN Error

30.4.7.1 Bit Error

When the USART is transmitting, if the value transmitted on the TX line differs from the value sampled on the RX line, a bit error will be generated in the slave node configuration. If a bit error is detected, the transmission is aborted at the next byte boundary.

This error is indicated by the LINBE flag in the channel status register (USART_CSR).

30.4.7.2 Sync Field Inconsistency Error

If the received sync field character is not 0x55, this error will be generated in the slave node configuration.

This error is indicated by the LINISFE flag in the channel status register (USART_CSR).

30.4.7.3 Identifier Parity Error

If there is a parity error in the identifier, this error will be generated in the slave node configuration. This error can only be generated if the parity function is enabled (PARDIS = 0).

This error is indicated by the LINIPE flag in the channel status register (USART_CSR).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.4.7.4 Checksum Error

If the received checksum is incorrect, this error will be generated in the master configuration of the slave node. This flag will only be set if the checksum function is enabled (CHKDIS = 0).

This error is indicated by the LINCE flag in the channel status register (USART_CSR).

30.4.7.5 Slave Not Responding Error

When the USART expects a response from other nodes (NACT = SUBSCRIBE), but no valid message appears on the bus within the maximum length $t_{Frame_Maximum}$ of the message frame, this error will be generated in the master configuration of the slave node. This error is disabled if the USART does not expect any messages (NACT = PUBLISH or NACT = IGNORE).

This error is indicated by the LINSNRE flag in the channel status register (USART_CSR).

30.4.7.6 Synchronization Tolerance Error

If, after the clock synchronization procedure, the calculated baud rate deviation is greater than the maximum tolerance FTol_Unsynch (±15%) compared to the initial baud rate, this error will be generated in the slave node configuration.

This error is indicated by the LINSTE flag in the channel status register (USART_CSR).

30.4.7.7 Message Header Timeout Error

If the message header is not fully received within the maximum length $t_{Header_Maximum}$, this error will be generated in the slave node configuration.

This error is indicated by the LINHTE flag in the channel status register (USART_CSR).

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.4.8 LIN Frame Processing

30.4.8.1 Master Node Configuration

- Write TXEN and RXEN of USART_CR to enable transmission and reception.
- Write USART_MODE of USART_MR to select LIN mode and master node.
- Write CD and FP of USART_BRGR to configure the baud rate.
- Configure the frame transmission by writing NACT, PARDIS, CHKDIS, CHKTYPE, DLCM, FSDIS and DLC of USART_LINMR.
- Check if TXRDY in USART_CSR is set.
- Write IDCHR of USART_LINIR to transmit the message header.

The subsequent operations depend on the NACT configuration:

- Case 1: NACT = PUBLISH, USART sends a response.
    1. Wait for TXRDY of USART_CSR to go high.
    2. Write TCHR of USART_THR to send one byte.
    3. If not all data has been written, repeat the above two steps.
    4. Wait for LINTC of USART_CSR to go high.
    5. Check for LIN errors.
- Case 2: NACT = SUBSCRIBE, USART receives a response.
    1. Wait for RXRDY of USART_CSR to go high.
    2. Read RCHR of USART_RHR.
    3. If not all data has been read, repeat the above two steps.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. Wait for LINTC in USART_CSR to go high.

5. Check for LIN errors.

- Case 3: NACT = IGNORE, USART does not pay attention to the response.

1. Wait for LINTC of USART_CSR to go high.

2. Check for LIN errors.

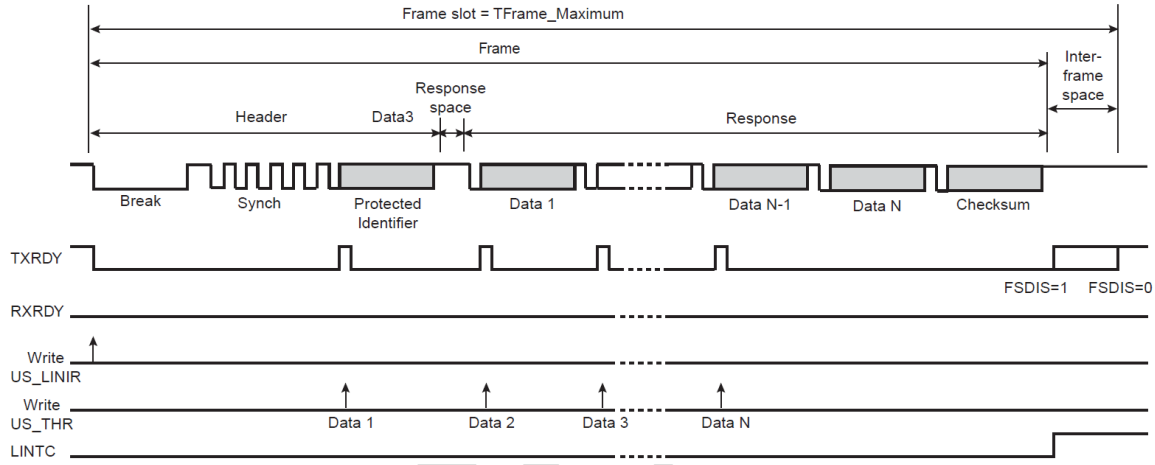

Figure 30-35: Master Node Configuration, NACT = PUBLISH

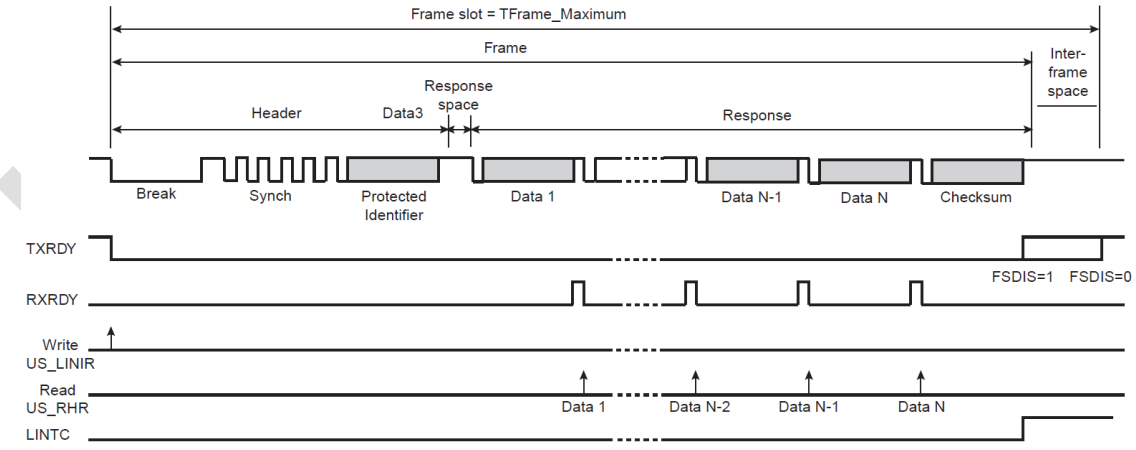

Figure 30-36: Master Node Configuration, NACT = SUBSCRIBE

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

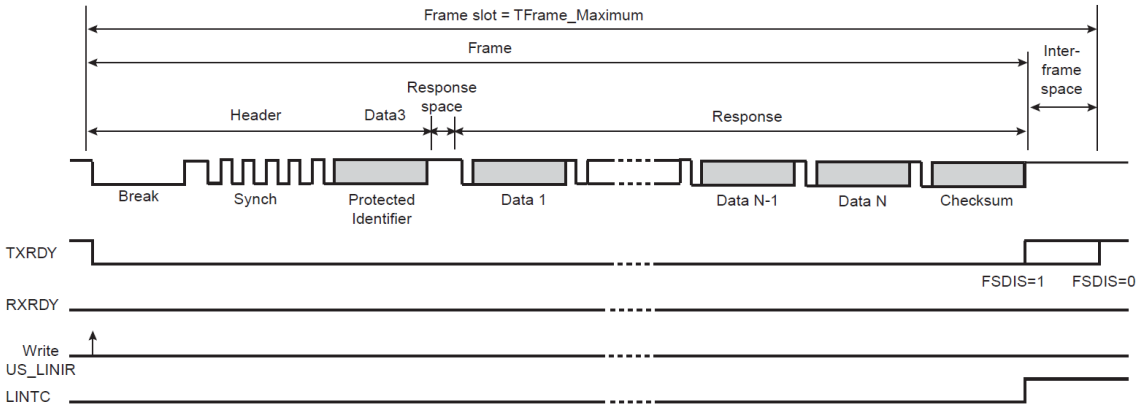

Figure 30-37: Master Node Configuration, NACT = IGNORE

30.4.8.2 Slave Node Configuration

- Write TXEN and RXEN of USART_CR to enable transmission and reception.
- Write USART_MODE of USART_MR to select LIN mode and slave node.
- Write CD and FP of USART_BRGR to configure the baud rate.
- Wait for LINID of USART_CSR to go high.
- Check for LINISFE and LINPE errors.
- Read IDCHR of USART_RHR.
- Configure the frame transmission by writing NACT, PARDIS, CHKDIS, CHKTYPE, DLCM and DLC of USART_LINMR.

Note: If NACT for this frame is configured as PUBLISH, the USART_LINMR register must be written with NACT = PUBLISH even if this field is properly configured, so as to set the TXREADY flag and the corresponding write transmission request.

The subsequent operations depend on the NACT configuration:

- Case 1: NACT = PUBLISH, USART sends a response.

  1. Wait for TXRDY of USART_CSR to go high.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

2. Write TCHR of USART_THR to send one byte.
3. If not all data has been written, repeat the above two steps.
4. Wait for LINTC of USART_CSR to go high.
5. Check for LIN errors.

- Case 2: NACT = SUBSCRIBE, USART receives a response.

1. Wait for RXRDY of USART_CSR to go high.
2. Read RCHR of USART_RHR.
3. If not all data has been read, repeat the above two steps.
4. Wait for LINTC in USART_CSR to go high.
5. Check for LIN errors.

- Case 3: NACT = IGNORE, USART does not pay attention to the response.

1. Wait for LINTC of USART_CSR to go high.
2. Check for LIN errors.

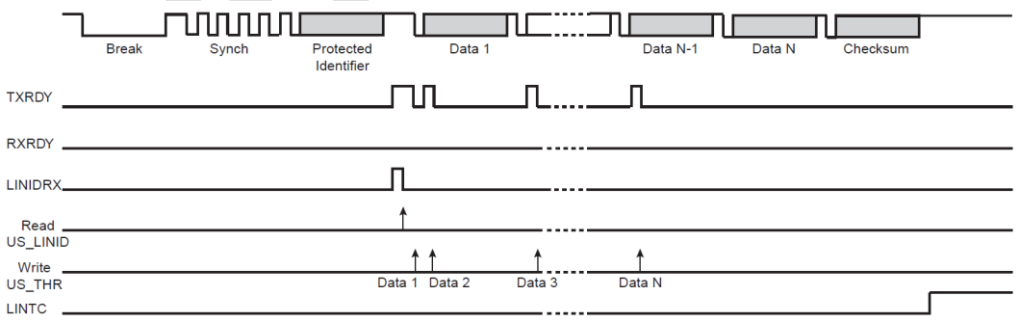

Figure 30-38: Slave Node Configuration, NACT = PUBLISH

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

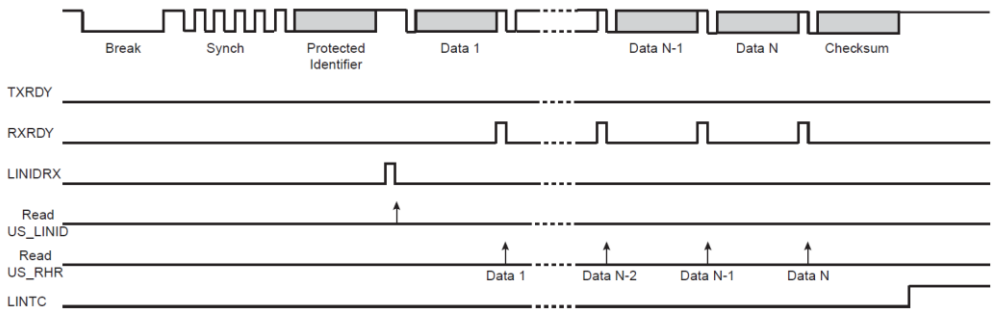

Figure 30-39: Slave Node Configuration, NACT = SUBSCRIBE

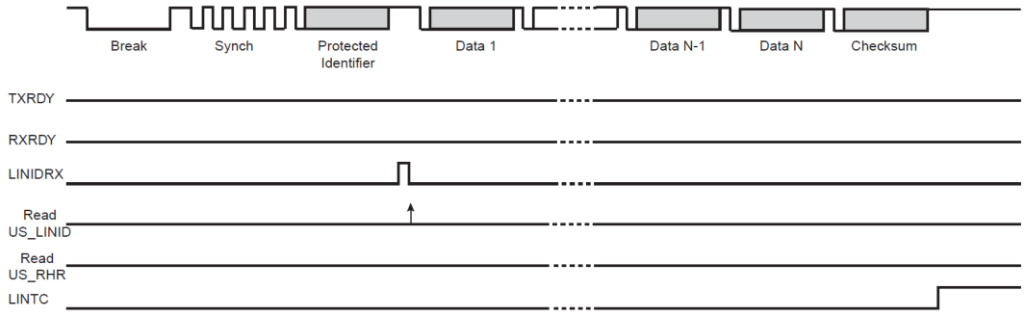

Figure 30-40: Slave Node Configuration, NACT = IGNORE

30.4.9 Using DMA/PDC to Process LIN Frames

USART can be used in conjunction with DMA/PDC to transfer data directly to and from on-chip and off-chip memory without any processor intervention.

DMA/PDC uses the TXRDY and RXRDY trigger flags to write to or read from the USART. DMA/PDC writes to the USART transmit holding register (USART_THR) and reads from the USART receive holding register (USART_RHR). The data size for DMA/PDC writing or reading in the USART is one byte.

30.4.9.1 Master Node Configuration

Users can select between two DMA/PDC modes via the LIN mode register (USART_LINMR) using the PDCM bit:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- PDCM = 1: The LIN configuration is stored in the write buffer, and is written by DMA/PDC to the transmit holding register (USART_THR) rather than the LIN mode register (USART_LINMR). Since the DMA/PDC transfer size is limited to one byte, the transfer is divided into two accesses. During the first access, the bits NACT, PARDIS, CHKDIS, CHKTYP, DLM and FSDIS are written. During the second access, the 8-bit DLC field is written.
- PDCM = 0: The LIN configuration is not stored in the write buffer and must be written by the user to the LIN mode register (USART_LINMR).

If the USART sends a response (NACT = PUBLISH), the write buffer also contains the identifier and data.

If the USART receives a response (NACT = SUBSCRIBE), the read buffer contains the data.

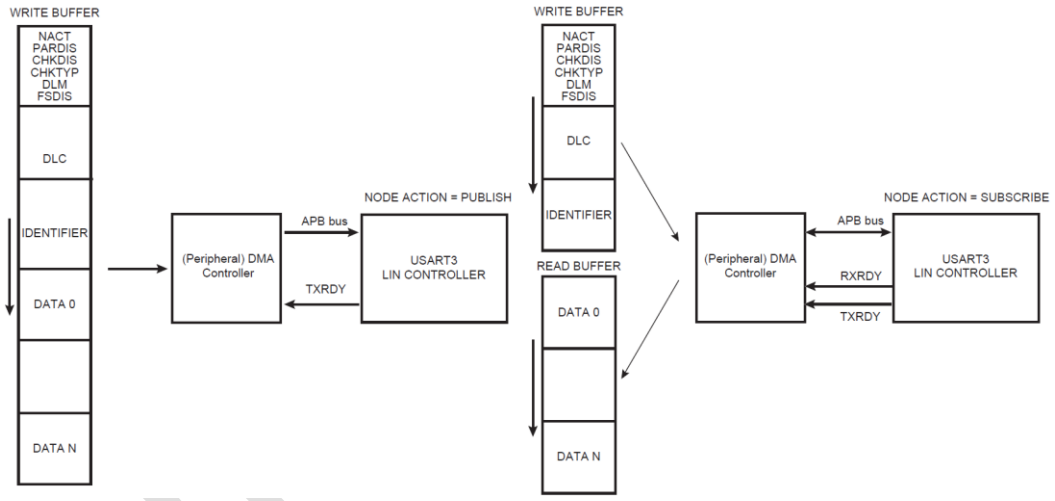

Figure 30-41: Master Node with DMA/PDC (PDCM = 1)

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

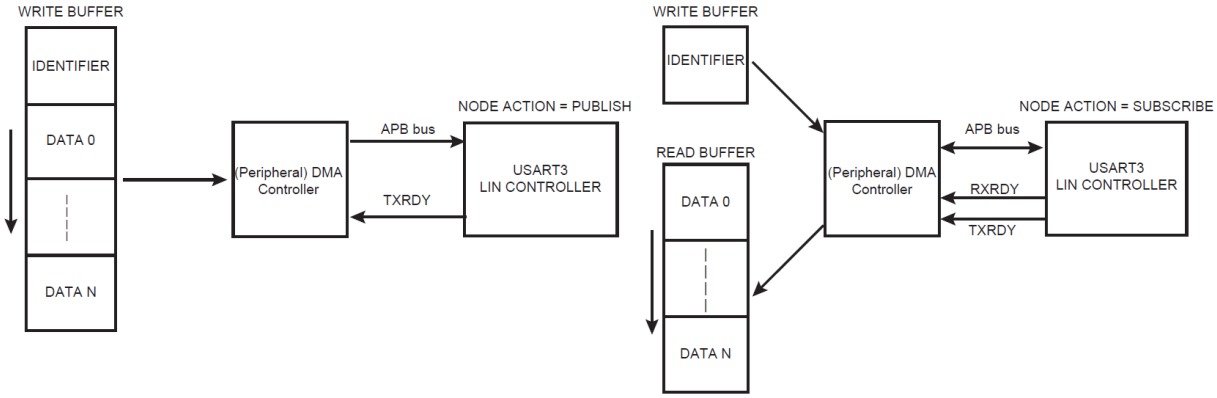

Figure 30-42: Master Node with DMA/PDC (PDCM = 0)

30.4.9.2 Slave Node Configuration

In this configuration, DMA/PDC only transfers data. The identifier must be read by the user from the LIN identifier register (USART_LINIR). The LIN mode must be written by the user to the LIN mode register (USART_LINMR).

If the USART sends a response (NACT = PUBLISH), the write buffer contains the data.

If the USART receives a response (NACT = SUBSCRIBE), the read buffer contains the data.

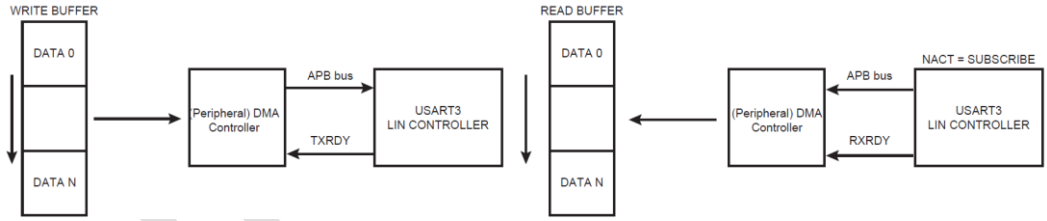

Figure 30-43: USART Slave Node Configuration

30.4.9.3 Wake-up Request

Any node in the LIN group can request to wake up from low power mode.

In the LIN 2.0 specification, a wake-up request is sent by forcing the bus into a dominant state for 250 µs to 5 ms. To achieve this, the character 0xF0 must be sent to impose five consecutive dominant bits. This character expects the specified timing regardless of the baud rate:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Baud rate min = 1 kbps -> t_{bit} = 1 ms -> 5 t_{bit}s = 5 ms
- Baud rate min = 20 kbps -> t_{bit} = 50 µs -> 5 t_{bit}s = 250 µs

In the LIN 1.3 specification, to impose eight consecutive dominant bits, the character 0x80 is used to generate a wake-up request.

Users can choose to send a LIN 2.0 wake-up request (WKUPTYP = 0) or a LIN 1.3 wake-up request (WKUPTYP = 1) via the WKUPTYP bit in the LIN mode register (USART_LINMR).

To transmit the wake-up request, the LINWKUP bit in the control register (USART_CR) must be set to 1. After the transmission is complete, the LINTC flag in the status register (USART_SR) is set. It can be cleared by writing 1 to the RSTSTA bit in USART_CR.

30.4.9.4 Bus Idle Timeout

If the LIN bus remains inactive for a certain period, slave nodes should automatically enter the low power mode. In the LIN 2.0 specification, the timeout is fixed at 4 seconds. In the LIN 1.3 specification, it is fixed at 25000 t_{bit}s.

In the slave node configuration, the receiver detects the idle state on the RXD line for timeout. When a timeout is detected, the TIMEOUT bit in the channel status register (USART_CSR) is set high, generating an interrupt to indicate that the driver should enter the sleep mode.

The timeout delay period (during which the receiver waits to receive a new character) is configured in the TO field of the receiver timeout register (USART_RTOR). If TO is written to 0, the receiver timeout is disabled and no timeout is detected. The TIMEOUT bit in USART_SR remains at 0. Otherwise, the receiver loads a 17-bit counter with the TO set value. This counter decrements with each bit period and is reloaded each time a new character is received. If the counter reaches 0, the TIMEOUT bit in USART_CSR is set high.

If STTTO is executed, the counter clock stops until the first character is received.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If RETTO is executed, the counter immediately starts decrementing from the TO value.

30.4.10 Test Mode

The USART can be configured into three different test modes. The internal loop back enables on-board diagnostics. In loop back mode, it is configured for internal or external loop back depending on whether the USART interface pins are connected or not.

30.4.10.1 Normal Mode

In normal mode, the RXD pin is connected to the receiver input, while the TXD pin is connected to the transmitter output.

Normal mode configuration:

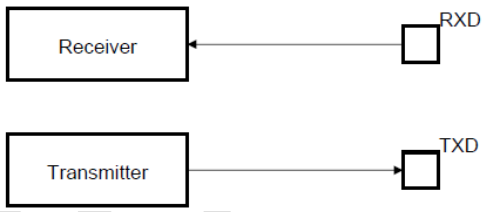

Figure 30-44: USART Normal Mode

30.4.10.2 Auto-response Mode

The auto-response mode allows one-bit retransmission. After receiving a bit at the RXD pin, send it to the TXD pin, as shown in the figure below. Programming the transmitter does not affect the TXD pin; the RXD pin remains connected to the receiver input, keeping the receiver active.

Auto-response mode configuration:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

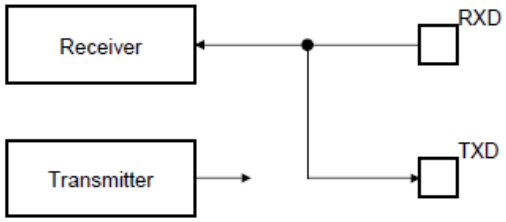

Figure 30-45: USART Auto-response Mode

30.4.10.3 Local Loop-back Mode

In local loop-back mode, the transmitter output is directly connected to the receiver input, as shown in the figure below. The TXD and RXD pins are unused. The RXD pin is inactive for the receiver, while the TXD pin is always high as in the idle state.

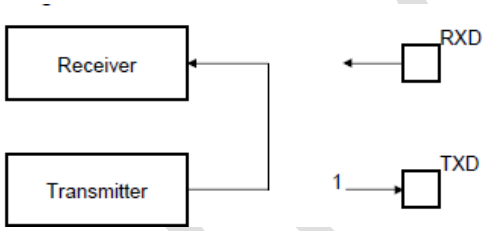

Figure 30-46: USART Local Loop-back Mode

30.4.10.4 Remote Loop-back Mode

In remote loop-back mode, the RXD pin is directly connected to the TXD pin, as shown in the figure below. Disabling the transmitter and receiver has no effect. This mode allows one-bit retransmission.

Remote loop-back mode configuration:

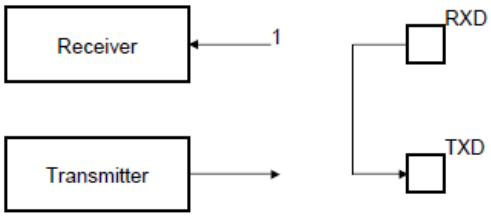

Figure 30-47: USART Remote Loop-back Mode

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5 Register Description

USART7 register base address: 0x40B0_4000

USART8 register base address: 0x4700_C000

The registers are listed below:

Table 30-9: List of USART Registers

Offset Address	Name	Description
0x00	USART_CR	Control register
0x04	USART_MR	Mode register
0x08	USART_IER	Interrupt enable register
0x0C	USART__IDR	Interrupt disable register
0x10	USART_IMR	Interrupt mask register
0x14	USART_CSR	Channel status register
0x18	USART_RHR	Receiver holding register
0x1C	USART_THR	Transmitter holding register
0x20	USART_BRGR	Baud rate generator register
0x24	USART_RTOR	Receiver timeout register
0x28	USART_TTGR	Transmitter time protection register
0x2C–0x3C	–	Reserved
0x40	USART_FIDI	FI/DI ratio register
0x44	-	Reserved
0x48	–	Reserved
0x4C	USART_IF	IrDA filter register
0x50	USART_MAN	Manchester configuration register
0x54	USART_LINMR	LIN mode register
0x58	USART_LINIR	LIN identifier register
0x5C	USART_LINBRR	LIN baud rate register
0x60-0xE0	–	Reserved
0xE4	USART_WPMR	Write protection mode register
0xE8	USART_WPSR	Write protection status register

30.5.1 USART Control Register (USART_CR)

Offset address: 0x00

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:22	RSV	-	-	Reserved
21	LINWKUP	W	0x0	Send LIN wakeup signal: 0: Invalid 1: Send a wakeup signal to the LIN bus
20	LINABT	W	0x0	Abort LIN transmission: 0: Invalid 1: Abort the current LIN transmission
19	RTSDIS	W	0x0	Request transmission disable: 0: Invalid 1: Drive the RTS pin to 1
18	RTSEN	W	0x0	Request transmission enable: 0: Invalid 1: Drive the RTS pin to 0
17	DTRDIS	W	0x0	Data terminal ready disable: 0: Invalid 1: Drive the DTR pin to 0
16	DTREN	W	0x0	Data terminal ready enable: 0: Invalid 1: Drive the DTR pin to 1
15	RETTO	W	0x0	Reload and start timeout: 0: Invalid 1: Restart timeout
14	RSTNACK	W	0x0	No acknowledgment reset: 0: Invalid 1: Reset NACK in the USART_CSR register
13	RSTIT	W	0x0	Iteration reset: 0: Invalid 1: Reset iteration in the USART_CSR register
12	SENDA	W	0x0	Send address: 0: Invalid 1: Only applicable to multipoint mode, send the address character written to USART_THR
11	STTTO	W	0x0	Start timeout:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Invalid 1: Wait for one character before the timeout counter starts counting, reset the TIMEOUT status bit in USART_CSR.
10	STPBRK	W	0x0	Stop break: 0: Invalid 1: Stop sending breaks after at least one character time and a high level of 12-bit cycle. If a break has been sent, this option is invalid.
9	STTBRK	W	0x0	Start break: 0: Invalid 1: Start sending a break after there is a character in USART_THR and the character has been sent from the shift register. If a break has been sent, this option is invalid.
8	RSTSTA	W	0x0	Status bit reset: 0: Invalid 1: Reset the status bits PARE, FRAME, OVRE, MANERR and XBRK in the USART_CSR register
7	TXDIS	W	0x0	Transmitter disable: 0: Invalid 1: Transmitter disabled
6	TXEN	W	0x0	Transmitter enable: 0: Invalid 1: Transmitter enabled when TXDIS is 0
5	RXDIS	W	0x0	Receiver disable: 0: Invalid 1: Receiver disabled
4	RXEN	W	0x0	Receiver enable: 0: Invalid 1: Receiver enabled when RXDIS is 0
3	RSTTX	W	0x0	Transmitter reset: 0: Invalid 1: Transmitter reset
2	RSTRX	W	0x0	Receiver reset:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Invalid 1: Receiver reset
1:0	RSV	-	-	Reserved

30.5.2 USART Control Register (USART_CR, SPI Mode)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	Reserved
19	RCS	W	0x0	Cancel SPI chip select: 0: Invalid 1: Release the slave select line NSS (RTS pin)
18	FCS	W	0x0	Force SPI chip select: 0: Invalid 1: Force the slave select line NSS (RTS pin) to 0, even if USART is not sending data, to enable the SPI slave device to support CSAAT (chip select active after transfer) mode.
17:9	RSV	-	-	Reserved
8	RSTSTA	W	0x0	Status bit reset: 0: Invalid 1: Reset the status bits PARE, FRAME, OVRE, MANERR and XBRK in the USART_CSR register
7	TXDIS	W	0x0	Transmitter disable: 0: Invalid 1: Transmitter disabled
6	TXEN	W	0x0	Transmitter enable: 0: Invalid 1: Transmitter enabled when TXDIS is 0
5	RXDIS	W	0x0	Receiver disable: 0: Invalid

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Receiver disabled
4	RXEN	W	0x0	Receiver enable: 0: Invalid 1: Receiver enabled when RXDIS is 0
3	RSTTX	W	0x0	Transmitter reset: 0: Invalid 1: Transmitter reset
2	RSTRX	W	0x0	Receiver reset: 0: Invalid 1: Receiver reset
1:0	RSV	-	-	Reserved

30.5.3 USART Mode Register (USART_MR)

Offset address: 0x04

Reset value: 0xC000 0000

Bit	Name	Attribute	Reset Value	Description
31	ONEBIT	R/W	0x1	Frame start delimiter selector: 0: Frame start delimiter is COMMAND or DATASYNC. 1: Frame start delimiter is a single bit.
30	MODSYNC	R/W	0x1	Manchester synchronization mode: 0: Manchester start bit is a level transition from 0 to 1. 1: Manchester start bit is a level transition from 1 to 0.
29	MAN	R/W	0x0	Manchester encoder/decoder enable: 0: Manchester encoder/decoder disabled 1: Manchester encoder/decoder enabled
28	FILTER	R/W	0x0	Infrared receiver line filter: 0: USART does not filter the receive line. 1: USART uses a 3-point sampling filter (1/16-bit clock) (2/3 more) to filter the receive line.
27:24	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
23	INVDATA	R/W	0x0	Data inversion: 0: The data transmitted on the TXD line matches the data written to the USART_THR register, or the content read from the USART_RHR register matches the data received on the RXD line. This is the normal mode operation. 1: The data transmitted on the TXD line is inverted (only the voltage polarity) compared to the value written to the USART_THR register; or the value read from the USART_RHR register is inverted compared to the data received on the RXD line. This is the inversion mode operation, which is particularly useful in non-contact smart card applications and can be configured through the MSBF bit.
22	VAR_SYNC	R/W	0x0	Command synchronization factor / data synchronization frame start delimiter: 0: User-defined command configuration or data sync field determined by the SYNC value 1: The sync field is updated when a character is written to the USART_THR register.
21	DSNACK	R/W	0x0	Disable continuous NACK: 0: Once a received character has a parity error, NACK is sent on the ISO line (unless INACK is set). 1: When the number of consecutive parity errors has not reached the value given by the MAX_ITERATION field, these parity errors generate NACK on the ISO line. Once that value is reached, no additional NACK is sent on the ISO line, and the ITERATION flag is set.
20	INACK	R/W	0x0	Suppress no acknowledgment:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Generate NACK. 1: Do not generate NACK. Note: In SPI master mode, if INACK is 0 and a character is written to the USART_THR register (assuming TXRDY is set), the character start bit is sent immediately. When INACK is 1, another condition must also be met, which is that when a character is written to the USART_THR register, the character is sent immediately only if the RXRDY flag is cleared to 0 (achieved by reading the receiver holding register).
19	OVER	R/W	0x0	Oversampling mode: 0: 16 times oversampling 1: 8 times oversampling
18	CLKO	R/W	0x0	Clock output selection: 0: USART does not drive the SCK pin. 1: If USCLKS has not selected an external clock SCK, USART drives the SCK pin.
17	MODE9	R/W	0x0	Character length of 9 bits: 0: CHRL defines the character length. 1: The character length is 9 bits.
16	MSBF	R/W	0x0	Bit order: 0: LSB first 1: MSB first
15:14	CHMODE	R/W	0x0	Channel mode: 00: Normal mode 01: Auto-response mode, with receiver input connected to TXD pin 10: local loop-back mode, with transmitter output connected to receiver input 11: remote loop-back mode, with RXD pin connected to TXD pin
13:12	NBSTOP	R/W	0x0	Number of stop bit: Asynchronous (SYNC = 0) 00: 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				01: 1.5 10: 2 11: Reserved Synchronous (SYNC = 1) 00: 1 01: Reserved 10: 2 11: Reserved
11:9	PAR	R/W	0x0	Parity type: 000: Even parity 001: Odd parity 010: Parity forced to 0 (space) 011: Parity forced to 1 (mark) 10x: No parity 11x: Multi-point mode
8	SYNC	R/W	0x0	Synchronous mode selection: 0: USART operates in asynchronous mode. 1: USART operates in synchronous mode.
7:6	CHRL	R/W	0x0	Character length: 00: 5 bits 01: 6 bits 10: 7 bits 11: 8 bits
5:4	USCLKS	R/W	0x0	Clock selection: 00: MCK 01: MCK/DIV (DIV = 8) 10: Reserved 11: SCK
3:0	MODE	R/W	0x0	Operation mode: 0000: Normal mode 0001: RS485 0010: Hardware handshake 0011: Modem 0100: Reserved 0110: Reserved 1000: IrDA

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1001: Reserved 1010: LIN master 1011: LIN slave 1110: SPI master 1111: SPI slave Others: Reserved

30.5.4 USART Mode Register (USART_MR, SPI Mode)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	-	-	Reserved
20	WRDBT	R/W	0x0	Waiting for data to be read before transmission: 0: Character transmission begins immediately after the character is written to the USART_THR register (assuming TXRDY is set). 1: Character transmission starts only after the RXRDY flag is cleared (USART_RHR has been read).
19	RSV	-	-	Reserved
18	CLKO	R/W	0x0	Clock output selection: 0: USART does not drive the SCK pin. 1: If USCLKS has not selected an external clock SCK, USART drives the SCK pin.
17	RSV	-	-	Reserved
16	CPOL	R/W	0x0	SPI clock polarity: 0: The inactive state of SPCK is logic low (0). 1: The inactive state of SPCK is logic high (1).
15:9	RSV	-	-	Reserved
8	CPHA	R/W	0x0	SPI clock phase:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: USART operates in asynchronous mode. 1: USART operates in synchronous mode.
7:6	CHRL	R/W	0x0	Character length: 11: 8 bits Others: Reserved
5:4	USCLKS	R/W	0x0	Clock selection: 00: MCK 01: MCK/DIV 10: Reserved 11: SCK DIV setting: USART6: CFGR1[27:24] USART7: CFGR1[31:28]
3:0	MODE	R/W	0x0	Operation mode: 1110: SPI master 1111: SPI slave Others: Reserved

30.5.5 USART Interrupt Enable Register (UART_IER)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	-	-	Reserved
20	MANE	W	0x0	Manchester error interrupt enable
19	CTSIC	W	0x0	Transmission input change clear interrupt enable
18	DCDIC	W	0x0	Data carrier detection input change interrupt enable
17	DSRIC	W	0x0	Data ready input change interrupt enable
16	RIIC	W	0x0	Ring indicator input change interrupt enable
15:14	RSV	-	-	Reserved
13	NACK	W	0x0	NACK interrupt enable

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	RXBUFF	W	0x0	Receive buffer full interrupt enable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt enable
10	ITER	W	0x0	Reaching maximum repetition count interrupt enable
9	TXEMPTY	W	0x0	TXEMPTY interrupt enable
8	TIMEOUT	W	0x0	Timeout interrupt enable
7	PARE	W	0x0	Parity error interrupt enable
6	FRAME	W	0x0	Frame error interrupt enable
5	OVRE	W	0x0	Overflow error interrupt enable
4	ENDTX	W	0x0	Transmission complete interrupt enable
3	ENDRX	W	0x0	Reception complete interrupt enable
2	RXBRK	W	0x0	Receiver break interrupt enable
1	TXRDY	W	0x0	TXRDY interrupt enable
0	RXRDY	W	0x0	RXRDY interrupt enable

30.5.6 USART Interrupt Enable Register (USART_IER, SPI Mode)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	RXBUFF	W	0x0	Receive buffer full interrupt enable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt enable
10	UNRE	W	0x0	SPI underrun error interrupt enable
9	TXEMPTY	W	0x0	TXEMPTY interrupt enable
8:6	RSV	-	-	Reserved
5	OVRE	W	0x0	Overflow error interrupt enable
4	ENDTX	W	0x0	Transmission end interrupt enable
3	ENDRX	W	0x0	Reception end interrupt enable
2	RSV	-	-	Reserved
1	TXRDY	W	0x0	TXRDY interrupt enable
0	RXRDY	W	0x0	RXRDY interrupt enable

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.7 USART Interrupt Enable Register (USART_IER, LIN Mode)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	LINHTE	W	0x0	LIN message header timeout error interrupt enable
30	LINSTE	W	0x0	LIN synchronization tolerance error interrupt enable
29	LINSNRE	W	0x0	LIN slave NACK error interrupt enable
28	LINCE	W	0x0	LIN checksum error interrupt enable
27	LINIPE	W	0x0	LIN identifier parity interrupt enable
26	LINISFE	W	0x0	LIN sync field inconsistency error interrupt enable
25	LINBE	W	0x0	LIN bus error interrupt enable
24:16	RSV	-	-	Reserved
15	LINTC	W	0x0	LIN transmission complete interrupt enable
14	LINID	W	0x0	Transmit or receive LIN identifier interrupt enable
13	LINBK	W	0x0	Transmit or receive LIN break interrupt enable
12	RXBUFF	W	0x0	Receive buffer full interrupt enable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt enable
10	RSV	-	-	Reserved
9	TXEMPTY	W	0x0	TXEMPTY interrupt enable
8	TIMEOUT	W	0x0	Timeout interrupt enable
7	PARE	W	0x0	Parity error interrupt enable
6	FRAME	W	0x0	Frame error interrupt enable
5	OVRE	W	0x0	Overflow error interrupt enable
4	ENDTX	W	0x0	Transmission end interrupt enable
3	ENDRX	W	0x0	Reception end interrupt enable
2	RSV	-	-	Reserved
1	TXRDY	W	0x0	TXRDY interrupt enable
0	RXRDY	W	0x0	RXRDY interrupt enable

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.8 USART Interrupt Disable Register (USART_IDR)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	-	-	Reserved
20	MANE	W	0x0	Manchester error interrupt disable
19	CTSIC	W	0x0	Transmission input change clear interrupt disable
18	DCDIC	W	0x0	Data carrier detection input change interrupt disable
17	DSRIC	W	0x0	Data ready input change interrupt disable
16	RIIC	W	0x0	Ring indicator input change interrupt disable
15:14	RSV	-	-	Reserved
13	NACK	W	0x0	NACK interrupt disable
12	RXBUFF	W	0x0	Receive buffer full interrupt disable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt disable
10	ITER	W	0x0	Reaching maximum repetition count interrupt disable
9	TXEMPTY	W	0x0	TXEMPTY interrupt disable
8	TIMEOUT	W	0x0	Timeout interrupt disable
7	PARE	W	0x0	Parity error interrupt mask
6	FRAME	W	0x0	Frame error interrupt disable
5	OVRE	W	0x0	Overflow error interrupt disable
4	ENDTX	W	0x0	Transmission end interrupt disable
3	ENDRX	W	0x0	Reception end interrupt disable
2	RXBRK	W	0x0	Receiver break interrupt disable
1	TXRDY	W	0x0	TXRDY interrupt disable
0	RXRDY	W	0x0	RXRDY interrupt disable

30.5.9 USART Interrupt Disable Register (USART_IDR, SPI Mode)

Offset address: 0x0C

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	RXBUFF	W	0x0	Receive buffer full interrupt disable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt disable
10	UNRE	W	0x0	SPI underrun error interrupt disable
9	TXEMPTY	W	0x0	TXEMPTY interrupt disable
8:6	RSV	-	-	Reserved
5	OVRE	W	0x0	Overflow error interrupt disable
4	ENDTX	W	0x0	Transmission end interrupt disable
3	ENDRX	W	0x0	Reception end interrupt disable
2	RSV	-	-	Reserved
1	TXRDY	W	0x0	TXRDY interrupt disable
0	RXRDY	W	0x0	RXRDY interrupt disable

30.5.10 USART Interrupt Disable Register (USART_IDR, LIN Mode)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	LINHTE	W	0x0	LIN message header timeout error interrupt disable
30	LINSTE	W	0x0	LIN synchronization tolerance error interrupt disable
29	LINSNRE	W	0x0	LIN slave NACK error interrupt disable
28	LINCE	W	0x0	LIN checksum error interrupt disable
27	LINIPE	W	0x0	LIN identifier parity interrupt disable
26	LINISFE	W	0x0	LIN sync field inconsistency error interrupt disable
25	LINBE	W	0x0	LIN bus error interrupt disable
24:16	RSV	-	-	Reserved
15	LINTC	W	0x0	LIN transmission complete interrupt disable
14	LINID	W	0x0	Transmit or receive LIN identifier interrupt disable

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13	LINBK	W	0x0	Transmit or receive LIN break interrupt disable
12	RXBUFF	W	0x0	Receive buffer full interrupt disable
11	TXBUFE	W	0x0	Transmit buffer empty interrupt disable
10	RSV	-	-	Reserved
9	TXEMPTY	W	0x0	TXEMPTY interrupt disable
8	TIMEOUT	W	0x0	Timeout interrupt disable
7	PARE	W	0x0	Parity error interrupt mask
6	FRAME	W	0x0	Frame error interrupt disable
5	OVRE	W	0x0	Overflow error interrupt disable
4	ENDTX	W	0x0	Transmission end interrupt disable
3	ENDRX	W	0x0	Reception end interrupt disable
2	RSV	-	-	Reserved
1	TXRDY	W	0x0	TXRDY interrupt disable
0	RXRDY	W	0x0	RXRDY interrupt disable

30.5.11 USART Interrupt Mask Register (USART_IMR)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24:21	RSV	-	-	Reserved
20	MANE	R	0x0	Manchester error interrupt mask
19	CTSIC	R	0x0	Transmission input change clear interrupt mask
18	DCDIC	R	0x0	Data carrier detection input change interrupt mask
17	DSRIC	R	0x0	Data ready input change interrupt mask
16	RIIC	R	0x0	Ring indicator input change interrupt mask
15:14	RSV	-	-	Reserved
13	NACK	R	0x0	NACK interrupt mask
12	RXBUFF	R	0x0	Receive buffer full interrupt mask
11	TXBUFE	R	0x0	Transmit buffer empty interrupt mask

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	ITER	R	0x0	Reaching maximum repetition count interrupt mask
9	TXEMPTY	R	0x0	TXEMPTY interrupt mask
8	TIMEOUT	R	0x0	Timeout interrupt mask
7	PARE	R	0x0	Parity error interrupt mask
6	FRAME	R	0x0	Frame error interrupt mask
5	OVRE	R	0x0	Overflow error interrupt mask
4	ENDTX	R	0x0	Transmission end interrupt mask
3	ENDRX	R	0x0	Reception end interrupt mask
2	RXBRK	R	0x0	Receiver break interrupt mask
1	TXRDY	R	0x0	TXRDY interrupt mask
0	RXRDY	R	0x0	RXRDY interrupt mask

30.5.12 USART Interrupt Mask Register (USART_IMR, SPI Mode)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	RXBUFF	R	0x0	Receive buffer full interrupt mask
11	TXBUFE	R	0x0	Transmit buffer empty interrupt mask
10	UNRE	R	0x0	SPI underrun error interrupt mask
9	TXEMPTY	R	0x0	TXEMPTY interrupt mask
8:6	RSV	-	-	Reserved
5	OVRE	R	0x0	Overflow error interrupt mask
4	ENDTX	R	0x0	Transmission end interrupt mask
3	ENDRX	R	0x0	Reception end interrupt mask
2	RSV	-	-	Reserved
1	TXRDY	R	0x0	TXRDY interrupt mask
0	RXRDY	R	0x0	RXRDY interrupt mask

30.5.13 USART Interrupt Mask Register (USART_IMR, LIN Mode)

Offset address: 0x10

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	LINHTE	R	0x0	LIN message header timeout error interrupt mask
30	LINSTE	R	0x0	LIN synchronization tolerance error interrupt mask
29	LINSNRE	R	0x0	LIN slave NACK error interrupt mask
28	LINCE	R	0x0	LIN checksum error interrupt mask
27	LINIPE	R	0x0	LIN identifier parity interrupt mask
26	LINISFE	R	0x0	LIN sync field inconsistency error mask
25	LINBE	R	0x0	LIN bus error interrupt mask
24:16	RSV	-	-	Reserved
15	LINTC	R	0x0	LIN transmission complete interrupt mask
14	LINID	R	0x0	Transmit or receive LIN identifier interrupt mask
13	LINBK	R	0x0	Transmit or receive LIN break interrupt mask
12	RXBUFF	R	0x0	Receive buffer full interrupt mask
11	TXBUFE	R	0x0	Transmit buffer empty interrupt mask
10	RSV	-	-	Reserved
9	TXEMPTY	R	0x0	TXEMPTY interrupt mask
8	TIMEOUT	R	0x0	Timeout interrupt mask
7	PARE	R	0x0	Parity error interrupt mask
6	FRAME	R	0x0	Frame error interrupt mask
5	OVRE	R	0x0	Overflow error interrupt mask
4	ENDTX	R	0x0	Transmission end interrupt mask
3	ENDRX	R	0x0	Reception end interrupt mask
2	RSV	-	-	Reserved
1	TXRDY	R	0x0	TXRDY interrupt mask
0	RXRDY	R	0x0	RXRDY interrupt mask

30.5.14 USART Channel Status Register (USART_CSR)

Offset address: 0x14

Reset value: 0x0078 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	MANERR	R	0x0	Manchester error: 0: No Manchester error detected since the last RSTSTA 1: At least one Manchester error detected since the last RSTSTA
23	CTS	R	0x0	CTS input mirror: 0: CTS is 0. 1: CTS is 1.
22:20	RSV	-	-	Reserved
19	CTSIC	R	0x1	Transmission input change clear flag: 0: No input change detected on the CTS pin since the last read of USART_CSR 1: At least one input change detected on the CTS pin since the last read of USART_CSR
18:14	RSV	-	-	Reserved
13	NACK	R	0x0	No acknowledgment: 0: No acknowledgment detected since the last RSTNACK 1: At least one no-acknowledgment detected since the last RSTNACK
12:11	RSV	-	-	Reserved
10	ITER	R	0x0	Reaching maximum repetition count: 0: The maximum repetition count has not been reached since the last RSTSTA. 1: The maximum repetition count has been reached since the last RSTSTA.
9	TXEMPTY	R	0x0	Transmitter empty: 0: There is at least one character in either USART_THR or the transmitter shift register, or the transmitter is disabled. 1: There are no characters in either USART_THR or the transmitter shift register.
8	TIMEOUT	R	0x0	Receiver timeout:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: No timeout has occurred since the last timeout command STTTO was initiated, or the timeout register is 0. 1: A timeout has occurred since the last timeout command STTTO was initiated.
7	PARE	R	0x0	Parity error: 0: No parity error detected since the last RSTSTA 1: At least one parity error detected since the last RSTSTA
6	FRAME	R	0x0	Frame error: 0: No stop bit detected since the last RSTSTA 1: At least one stop bit detected since the last RSTSTA
5	OVRE	R	0x0	Overflow error: 0: No overflow error occurred since the last RSTSTA 1: At least one overflow error occurred since the last RSTSTA
4:3	RSV	-	-	Reserved
2	RXBRK	R	0x0	Break receive/complete: 0: No break receive or break complete detected since the last RSTSTA 1: Break receive or break complete detected since the last RSTSTA
1	TXRDY	R	0x0	Transmitter ready: 0: There are characters waiting to be sent from USART_THR to the transmitter shift register, or the STTBRK command is requested, or the transmitter is disabled. Once the transmitter is enabled, TXRDY becomes 1. 1: There are no characters in USART_THR.
0	RXRDY	R	0x0	Receiver ready:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: No complete character has been received since the last read of USART_RHR, or the receiver is disabled. If the receiver is disabled while receiving characters, RXRDY becomes 1 when the receiver is enabled. 1: At least one complete character has been received since the last read of USART_RHR, and USART_RHR has not been read.

30.5.15 USART Channel Status Register (USART_CSR, SPI Mode)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved
10	UNRE	R	0x0	SPI underrun error: 0: No SPI underrun error occurred since the last RSTSTA 1: At least one SPI underrun error occurred since the last RSTSTA
9	TXEMPTY	R	0x0	Transmitter empty: 0: There is at least one character in either USART_THR or the transmitter shift register, or the transmitter is disabled. 1: There are no characters in either USART_THR or the transmitter shift register.
8:6	RSV	-	-	Reserved
5	OVRE	R	0x0	Overflow error: 0: No overflow error occurred since the last RSTSTA 1: At least one overflow error occurred since the last RSTSTA

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4:2	RSV	-	-	Reserved
1	TXRDY	R	0x0	Transmitter ready: 0: There are characters waiting to be sent from USART_THR to the transmitter shift register, or the STTBRK command is requested, or the transmitter is disabled. Once the transmitter is enabled, TXRDY becomes 1. 1: There are no characters in USART_THR.
0	RXRDY	R	0x0	Receiver ready: 0: No complete character has been received since the last read of USART_RHR, or the receiver is disabled. If the receiver is disabled while receiving characters, RXRDY becomes 1 when the receiver is enabled. 1: At least one complete character has been received since the last read of USART_RHR, and USART_RHR has not been read.

30.5.16 USART Channel Status Register (USART_CSR, LIN Mode)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	LINHTE	R	0x0	LIN message header timeout error: 0: LIN message header timeout error not detected since the last RSTSTA 1: LIN message header timeout error detected since the last RSTSTA
30	LINSTE	R	0x0	LIN synchronization tolerance error: 0: LIN synchronization tolerance error not detected since the last RSTSTA

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: LIN synchronization tolerance error detected since the last RSTSTA
29	LINSNRE	R	0x0	LIN slave NACK error: 0: LIN slave NACK error not detected since the last RSTSTA 1: LIN slave NACK error detected since the last RSTSTA
28	LINCE	R	0x0	LIN checksum error: 0: Checksum error not detected since the last RSTSTA 1: Checksum error detected since the last RSTSTA
27	LINIPE	R	0x0	LIN identifier parity error: 0: LIN identifier parity error not detected since the last RSTSTA 1: LIN identifier parity error detected since the last RSTSTA
26	LINISFE	R	0x0	LIN sync field inconsistency error: 0: LIN sync field inconsistency error not detected since the last RSTSTA 1: USART configured as slave node and sync field inconsistency error detected since the last RSTSTA
25	LINBE	R	0x0	LIN bit error: 0: Bit error not detected since the last RSTSTA 1: Bit error detected since the last RSTSTA
24	RSV	-	-	Reserved
23	LINBLS	R	0x0	LIN bus status: 0: LIN bus is 0. 1: LIN bus is 1.
22:16	RSV	-	-	Reserved
15	LINTC	R	0x0	LIN transmission complete: 0: USART being idle or LIN transmission being in progress

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: LIN transmission completed since the last RSTSTA
14	LINID	R	0x0	Transmit or receive LIN identifier: LIN master 0: No LIN identifier transmitted since the last RSTSTA 1: At least one LIN identifier transmitted since the last RSTSTA LIN slave 0: No LIN identifier received since the last RSTSTA 1: At least one LIN identifier received since the last RSTSTA
13	LINBK	R	0x0	Transmit or receive LIN break: LIN master 0: No LIN break transmitted since the last RSTSTA 1: At least one LIN break transmitted since the last RSTSTA LIN slave 0: No LIN break received since the last RSTSTA 1: At least one LIN break received since the last RSTSTA
12:10	RSV	-	-	Reserved
9	TXEMPTY	R	0x0	Transmitter empty: 0: There are characters in either USART_THR or the shift register, or the transmitter is disabled. 0: There are no characters in either USART_THR or the shift register.
8	TIMEOUT	R	0x0	Timeout: 0: No timeout has occurred since the last timeout command (STTTO in USART_CR) was initiated, or the timeout register is 0.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: A timeout has occurred since the last timeout command (STTTO in USART_CR) was initiated.
7	PARE	R	0x0	Parity error: 0: No parity error occurred since the last RSTSTA 1: At least one parity error occurred since the last RSTSTA
6	FRAME	R	0x0	Frame error: 0: No frame error occurred since the last RSTSTA 1: At least one frame error occurred since the last RSTSTA
5	OVRE	R	0x0	Overflow error: 0: No overflow error occurred since the last RSTSTA 1: At least one overflow error occurred since the last RSTSTA
4:2	RSV	-	-	Reserved
1	TXRDY	R	0x0	Transmitter ready: 0: There are characters waiting to be sent from USART_THR to the transmitter shift register, or the STTBRK command is requested, or the transmitter is disabled. Once the transmitter is enabled, TXRDY becomes 1. 1: There are no characters in USART_THR.
0	RXRDY	R	0x0	Receiver ready: 0: No complete character has been received since the last read of USART_RHR, or the receiver is disabled. If the receiver is disabled while receiving characters, RXRDY becomes 1 when the receiver is enabled. 1: At least one complete character has been received since the last read of

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				USART_RHR, and USART_RHR has not been read.

30.5.17 USART Receive Holding Register (USART_RHR)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	RXSYNH	R	0x0	Received sync field: 0: The last received character is data. 1: The last received character is a command.
14:9	RSV	-	-	Reserved
8:0	RXCHR	R	0x0	Received character: If RXRDY is set, it is the last character received.

30.5.18 USART Transmit Holding Register (USART_THR)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	TXSYNH	W	0x0	Transmitted sync field: 0: The next character to be transmitted is encoded as data and the frame start delimiter is DATA SYNC. 1: The next character to be transmitted is encoded as a command, and the frame start delimiter is COMMAND SYNC.
14:9	RSV	-	-	Reserved
8:0	TXCHR	W	0x0	Character to be transmitted: If TXRDY is not set, the next character is transmitted after the current character.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.19 USART Baud Rate Generator Register (USART_BRGR)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18:16	FP	R/W	0x0	Fractional part: 0: Fractional divider disabled 1–7: Baud rate resolution, defined as FP x 1/8
15:0	CD	R/W	0x0	Clock frequency division Please refer to the table below.

Table 30-10 : Reference Table of USART Clock Frequency Division

CD	SYNC = 0		SYNC = 1 or USART_MODE = SPI (Master or Slave)
	OVER = 0	OVER = 1	
0	Baud rate clock disabled		
1–65535	Baud rate = selected clock / 16 / CD	Baud rate = selected clock / 8 / CD	Baud rate = selected clock / CD

30.5.20 USART Receiver Timeout Register (USART_RTOR)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16:0	TO	R/W	0x0	Timeout value: 0: Receiver timeout disabled 1–131071: Receiver timeout enabled with timeout delay of TO bit cycles

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.21 USART Transmit Time Protection Register (USART_TTGR)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	TG	R/W	0x0	Time protection value: 0: Transmitter time protection disabled 1–255: Transmitter time protection enabled with time protection delay of TG bit cycles

30.5.22 USART FI/DI Ratio Register (USART_FIDI)

Offset address: 0x40

Reset value: 0x0000 0174

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved
15:0	FI_DI_RATIO	R/W	0x174	Ratio of FI to DI

30.5.23 USART IrDA Filter Register (USART_IF)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	IrDA_FILTER	R/W	0x0	IrDA filter: The value of IrDA filter shall meet the following criteria: t_{MCK} * (IRDA_FILTER + 3) < 1.41 μs

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.24 USART Manchester Configuration Register (USART_MAN)

Offset address: 0x50

Reset value: 0xB001 1004

Bit	Name	Attribute	Reset Value	Description
31	RSV	-	-	Reserved
30	DRIFT	R/W	0x0	Drift compensation: 0: This USART cannot recover from a severe clock drift. 1: This USART can recover from clock drift. It must be realized in 16 times clock mode.
29	ONE	R/W	0x1	Must be set to 1: When writing to the USART_MAN register, this bit must always be set to 1.
28	RX_MPOL	R/W	0x1	Receiver Manchester polarity: 0: Logic 0 is encoded as a transition from 0 to 1, and logic 1 is encoded as a transition from 1 to 0. 1: Logic 0 is encoded as a transition from 1 to 0, and logic 1 is encoded as a transition from 0 to 1.
27:26	RSV	-	-	Reserved
25:24	RX_PP	R/W	0x0	Receiver preamble pattern detection: 00: ALL_ONE 01: ALL_ZERO 10: ZERO_ONE 11: ONE_ZERO
23:20	RSV	-	-	Reserved
19:16	RX_PL	R/W	0x1	Receiver preamble length: 0: Receiver preamble pattern detection disabled 1-15: The length of the detected preamble is RX_PL bit periods.
15:13	RSV	-	-	Reserved
12	TX_MPOL	R/W	0x1	Transmitter Manchester polarity:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Logic 0 is encoded as a transition from 0 to 1, and logic 1 is encoded as a transition from 1 to 0. 1: Logic 0 is encoded as a transition from 1 to 0, and logic 1 is encoded as a transition from 0 to 1.
11:10	RSV	-	-	Reserved
9:8	TX_PP	R/W	0x0	Transmitter preamble pattern: 00: ALL_ONE 01: ALL_ZERO 10: ZERO_ONE 11: ONE_ZERO
7:4	RSV	-	-	Reserved
3:0	TX_PL	R/W	0x4	Transmitter preamble length: 0: The generation of transmitter preamble is disabled. 1-15: The preamble length is TX_PL bit periods.

30.5.25 USART LIN Mode Register (USART_LINMR)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:18	RSV	-	-	Reserved
17	SYNCDIS	R/W	0x0	Synchronization disable: 0: Synchronization performed in the LIN slave node configuration 1: Synchronization not performed in the LIN slave node configuration
16	PDCM	R/W	0x0	DMA mode: 0: DMA cannot write to the LIN mode register USART_LINMR. 1: DMA writes to the LIN mode register USART_LINMR (except for flags).

Unicmicro

Bit	Name	Attribute	Reset Value	Description
15:8	DLC	R/W	0x0	Data length: When DLM = 0, it defines the response data length, which is equal to DLC + 1 bytes.
7	WKUPTYP	R/W	0x0	Wake-up signal type: 0: Set the LINWKUP bit in the control register to send a LIN 2.0 wake-up signal. 1: Set the LINWKUP bit in the control register to send a LIN 1.3 wake-up signal.
6	FSDIS	R/W	0x0	Frame slot mode disable: 0: Frame slot mode enabled 1: Frame slot mode disabled
5	DLM	R/W	0x0	Data length mode: 0: The response data length is defined by the DLC of this register. 1: The response data length is defined by bit 5 and bit 6 of the identifier (IDCHR of USART_LINIR).
4	CHKTYP	R/W	0x0	Checksum type: 0: LIN 2.0 “enhanced” checksum 1: LIN 1.3 “classic” checksum
3	CHKDIS	R/W	0x0	Checksum disable: 0: In the master node configuration, the checksum is automatically calculated and sent. In the slave node configuration, the checksum is automatically checked. 1: Regardless of how the node is configured, the checksum will not be calculated/sent, nor will it be checked.
2	PARDIS	R/W	0x0	Parity disable: 0: In the master node configuration, the identifier parity is automatically calculated and sent. In both master node and slave node configurations, the parity is automatically checked.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Regardless of how the node is configured, the identifier parity will not be calculated/sent, nor will it be checked.
1:0	NACT	R/W	0x0	LIN node operation: 00: PUBLISH (send a response) 01: SUBSCRIBE (receive a response) 10: IGNORE (neither send nor receive a response) 11: Reserved

30.5.26 USART LIN Identifier Register (USART_LINIR)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	IDCHR	R/W	0x0	Identifier character: When USART_MODE = 0xA (master node), IDCHR is read/write, and its value is the identifier character to be transmitted. When USART_MODE = 0xB (slave node), IDCHR is read-only, and its value is the identifier character to be received.

30.5.27 USART LIN Baud Rate Register (USART_LINBRR)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18:16	LINFP	R	0x0	Fractional part after synchronization
15:0	LINCD	R	0x0	Clock frequency division after synchronization

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.5.28 USART Write Protection Mode Register (USART_WPMR)

Offset address: 0xE4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	WPKEY	W	0x0	Writing any value other than 0x555341 in this field will abort the write operation for the WPEN bit.
7:1	RSV	-	-	Reserved
0	WPEN	R/W	0x0	0: Write protection is disabled when WPKEY is 0x555341. 1: Write protection is enabled when WPKEY is 0x555341.

Note: The following registers shall be protected:

- USART mode register (USART_MR)
- USART baud rate generator register (USART_BRGR)
- USART receive timeout register (USART_RTOT)
- USART transmit time protection register (USART_TTGR)
- USART FI/DI ratio register (USART_FIDI)
- USART IrDA filter register (USART_IF)
- USART Manchester configuration register (USART_MAN)

30.5.29 USART Write Protection Status Register (USART_WPSR)

Offset address: 0xE8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
23:8	WPVSRC	R	0x0	Write protection conflict status: 0: No write protection conflict has occurred since the last read of USART_WPSR. 1: A write protection conflict has occurred since the last read of USART_WPSR.
7:1	RSV	-	-	Reserved
0	WPVS	R	0x0	Write protection conflict source: When it is valid, an attempt is made to write the write protection register.

30.6 Operation Procedure

30.6.1 UART Mode

30.6.1.1 Initialization

1. Enable the clock for the corresponding GPIO pins and configure the pins for USART_TX and USART_RX alternate functions.
2. Configure the system configuration register for the USART module clock.
3. Configure the USART_BRGR register to set the baud rate for serial communication.
4. Configure the USART_MR register to set the data frame length, parity type and stop bit length for serial communication.
5. Configure the USART_CR register to enable transmission and reception.
6. Configure the USART_IER register to enable the required interrupts.

30.6.1.2 Transmission Process

1. Before transmitting data, the software can configure the baud rate parameter, parity type and data frame format.
2. Write the first data byte to the USART_THR register.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3. Query the transmission complete flag USART_CSR[9], if USART_CSR[9] = 1, the current data transmission is completed.

4. Write the next data byte to USART_THR.

30.6.1.3 Reception Process

1. Before receiving data, the software can configure the baud rate parameters, parity type and data frame format.

2. For data reception, query the USART_CSR[0] flag or wait for an interrupt. If USART_CSR[0] = 1, it means the receiver is not empty, then read the data from the USART_RHR, after which the corresponding flag will be automatically cleared.

3. If there is an error in data reception, wait for the interrupt or query the USART_CSR register flag bit to determine the error type and execute corresponding error handling, after which the software clears the error flag bit.

4. Continue to receive data.

30.6.2 SPI Mode

30.6.2.1 Master Mode

1. Initialization

   A. Enable the clocks for the corresponding GPIO pins and configure the pins for USART_RTS, USART_TX, USART_RX and USART_SCK alternate functions.

   B. Configure the system configuration register for the USART module clock.

   C. Set USART_MR[3:0] to SPI master mode.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

D. Configure USART_MR[7:6] to set the character length to 8 bits.

E. Configure USART_MR[18] to set the SCK pin to output clock.

F. Configure the USART_BRGR register to set the clock frequency division (the internal clock is divided by more than 6).

G. Configure USART_MR[16] and USART_MR[8] to set the SPI clock polarity and clock phase.

H. Configure the USART_CR register to enable the SPI transceiver.

2. Full-duplex data transmission

A. Set USART_CR[18] to 1 to enable the chip select (CS) to go low.

B. Wait until the transmitter is ready, then write data to the USART_THR register.

C. Query the transmission complete flag USART_CSR[9], if USART_CSR[9] = 1, the current data transmission is completed.

D. Query the USART_CSR flag or wait for an interrupt, If USART_CSR[0] = 1, read the data from the USART_RHR, after which the corresponding flag will be automatically cleared.

E. To continue transmitting and receiving data, repeat steps B–D.

F. After completing data transmission, set USART_CR[19] to 1 to disable the chip select (CS) to go high.

30.6.2.2 Slave Mode

1. Initialization

A. Enable the clocks for the corresponding GPIO pins and configure the pins for USART_CTS, USART_TX, USART_RX and USART_SCK alternate functions.

B. Configure the system configuration register for the USART module clock.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

C. Set USART_MR[3:0] to SPI slave mode.

D. Configure USART_MR[7:6] to set the character length to 8 bits.

E. Set USART_MR[18] to 3 to select SCK as the clock.

F. Configure USART_MR[16] and USART_MR[8] to set the SPI clock polarity and clock phase.

G. Configure the USART_CR register to enable the SPI transceiver.

2. Full-duplex data transmission

A. Wait until the transmitter is ready, then write data to the USART_THR register.

B. Query the transmission complete flag USART_CSR[9], if USART_CSR[9] = 1, the current data transmission is completed.

C. Query the USART_CSR flag or wait for an interrupt, If USART_CSR[0] = 1, read the data from the USART_RHR, after which the corresponding flag will be automatically cleared.

D. To continue transmitting and receiving data, repeat steps B–D.

30.6.3 LIN Mode

30.6.3.1 Master Node Mode

1. Initialization

A. Enable the clock for the corresponding GPIO pins and configure the pins for USART_TX and USART_RX alternate functions.

B. Configure the system configuration register for the USART module clock.

C. Set USART_MR[3:0] to LIN master node mode.

D. Configure the USART_CR register to enable the SPI transceiver.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

E. Configure the USART_LINMR register to set the operating mode of the master node.

F. Configure the USART_BRGR register to set the baud rate for LIN communication.

2. Transmitting data

A. Configure USART_LINMR[1:0] to set the master node operation mode as PUBLISH to send an acknowledgment.

B. Configure USART_LINMR[15:8] to set the data length of the LIN frame.

C. Wait for USART_CSR[1] to be set, indicating the transmitter is ready, then write the ID identifier into USART_LINIR[7:0].

D. Wait for USART_CSR[1] to be set, indicating the ID identifier has been sent, then write the data to be transmitted into USART_THR, and wait for USART_CSR[1] to be set again indicating the data transmission is complete before writing the next data, until all data is transmitted.

E. To transmit data again, repeat steps C and D.

3. Receiving data

A. Configure USART_LINMR[1:0] to set the master node operation mode as SUBSCRIBE to receive an acknowledgment.

B. Configure USART_LINMR[15:8] to set the data length of the LIN frame.

C. Wait for USART_CSR[1] to be set, indicating the transmitter is ready, then write the ID identifier into USART_LINIR[7:0].

D. Wait for USART_CSR[0] to be set, then read the data from USART_RHR.

E. Repeat step D until all data has been read.

F. To receive data again, repeat steps C and D.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

30.6.3.2 Slave Node Mode

1. Initialization

   A. Enable the clock for the corresponding GPIO pins and configure the pins for USART_TX and USART_RX alternate functions.

   B. Configure the system configuration register for the USART module clock.

   C. Set USART_MR[3:0] to LIN slave node mode.

   D. Configure the USART_CR register to enable the SPI transceiver.

   E. Configure the USART_LINMR register to set the operating mode of the slave node.

   F. Configure the USART_BRGR register to set the baud rate for LIN communication.

2. Transmitting data

   A. Wait for USART_CSR[14] to be set, indicating that the received identifier is correct.

   B. Configure USART_LINMR[15:8] to set the data length of the LIN frame.

   C. Configure USART_LINMR[1:0] to set the slave node operation mode as PUBLISH to send an acknowledgment, wait for USART_CSR[1] to be set, write the data to be transmitted into USART_THR, and wait for USART_CSR[1] to be set again indicating the data transmission is complete before writing the next data, until all data is transmitted.

3. Receiving data

   A. Configure USART_LINMR[1:0] to set the slave node operation mode as SUBSCRIBE to receive an acknowledgment.

   B. Wait for USART_CSR[14] to be set, indicating that the received identifier is correct.

   C. Configure USART_LINMR[15:8] to set the data length of the LIN frame.

   D. Wait for USART_CSR[0] to be set, then read the data from USART_RHR.

   E. Repeat step D until all data has been read.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

F. Wait for USART_CSR[15] to be set, indicating the LIN bus transmission is complete.

G. Set USART_CR[8] to 1 to reset the status bit.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31 Serial Peripheral Interface (SPI)

31.1 Overview

SPI is widely used to provide economical board-level interface between different devices such as EEPROM, FLASH, micro controller, DAC, ADC, etc.

31.2 Main Features

- Configurable master or slave mode
- Supporting serial full duplex, half duplex, simplex transmitting and simplex receiving in both master and slave modes
- Configurable 16-bit SPI clock frequency control register: SCK frequency up to $f_{PCLK}/2$ in master mode while up to $f_{PCLK}/4$ in slave mode
- Programmable SCK polarity and phase
- Supporting SPI Motorola mode or TI mode
- Programmable data order with MSB-first or LSB-first shifting
- Character length configurable from 1 bit to 32 bits, with the default being 8 bits
- Supporting data concatenation to make full use of FIFO space
- Built-in TX FIFO and RX FIFO with 8 x 32-bit depth
- Programmable software- or hardware-controlled chip select
- DMA operation

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.3 System Block Diagram

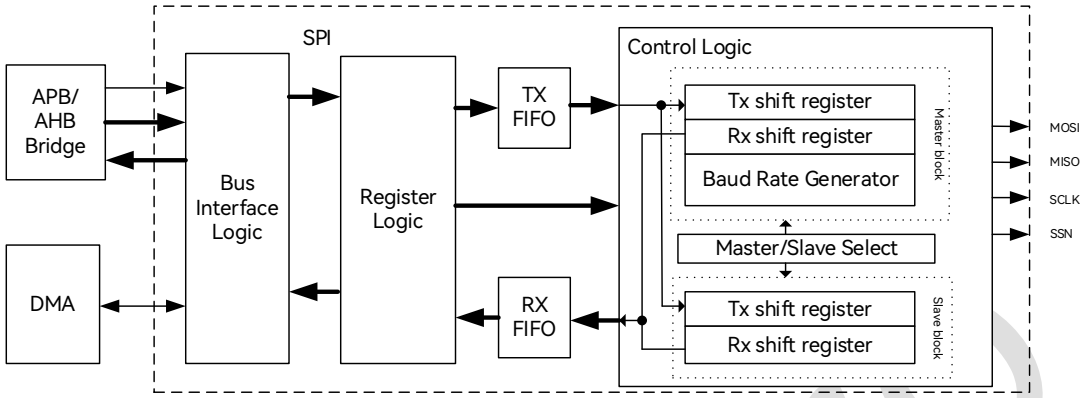

Figure 31-1: SPI Module Block Diagram

As shown in Figure 31-1, the SPI module consists of five blocks: APB bus interface logic, configuration register logic, TX FIFO, RX FIFO, and SPI control logic. The SPI control logic is divided into slave and master modules, where the master module includes transmit shift logic, receive shift logic, and SPI clock frequency generator logic.

31.4 Pin Description

Table 31-1: SPI Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
SPI1_SCK	PA5, PB3	Input/output	Serial clock signal
SPI1_MISO	PA6, PB4	Input/output	Data signal
SPI1_MOSI	PA7, PB5	Input/output	Data signal
SPI1_NSS	PA4, PA15	Input/output	Chip select signal
SPI2_SCK	PB10, PB13	Input/output	Serial clock signal
SPI2_MISO	PC2, PB14	Input/output	Data signal
SPI2_MOSI	PC3, PB15	Input/output	Data signal
SPI2_NSS	PB9, PB12	Input/output	Chip select signal
SPI3_SCK	PB3, PC10	Input/output	Serial clock signal
SPI3_MISO	PB4, PC11	Input/output	Data signal
SPI3_MOSI	PB5, PC12	Input/output	Data signal
SPI3_NSS	PA4, PA15	Input/output	Chip select signal

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
SPI4_SCK	PE3	Input/output	Serial clock signal
SPI4_MISO	PE5	Input/output	Data signal
SPI4_MOSI	PE6	Input/output	Data signal
SPI4_NSS	PE4	Input/output	Chip select signal

31.5 Functional Description

31.5.1 SPI clock frequency control register

The dedicated 16-bit "SPI clock frequency control register (SPBRG)" is applicable only in SPI master mode, and it controls the frequency of data transmission and reception.

The SPI clock (SCLK) is generated by dividing the APBx clock (PCLK), and the value of SPBRG is the division factor.

$$f_{SCLK} = f_{PCLK} / SPBRG$$

Where the value in the SPBRG register defaults to 2 and shall range from 2 to 65535. In master mode, the maximum SCLK frequency is up to $f_{PCLK}/2$ in master mode while up to $f_{PCLK}/4$ in slave mode.

31.5.2 TX FIFO

The TX FIFO has a depth of 8 words (one word is 32 bits). When the CPU or DMA writes data to the "transmit data register (TXREG)," the data is written into the TX FIFO, and the transmission begins. During the transmission, the TX FIFO shifts data to the "transmit shift register (TSR)."

When the "transmit shift register (TSR)" is empty, the TX FIFO outputs data to the TSR. The TSR then shifts the data to the transmit serial port (TX).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When the TX FIFO is not full and data transmission is required, the “transmit burst request (TXBREQ)” signal is asserted. This signal is sent to the DMA to request it to transfer data to the TX FIFO.

During transmission, when the TX FIFO is empty, the “transmitter empty interrupt flag (TXEPT_INTF)” is generated. When the TX FIFO receives enough data (depending on “TXTLF”), the “transmitter data available interrupt flag (TX_INTF)” is generated. When the SPI slave attempts to send a new character while idle, the “slave transmitter underrun interrupt flag (UNDERRUN_INTF)” is generated.

31.5.3 RX FIFO

The RX FIFO has a depth of 8 words (one word is 32 bits). The CPU or DMA receives data from the RX FIFO. The “receive shift register (RSR)” is used to receive data from the receive serial port (RX). Incoming data is placed into the RSR. Once all bits have been shifted in, the data is transferred to the RX FIFO. When the data is transferred from the “shift register” to the “data register,” the “receive register available bit (RXAVL)” in the “current status register (CSTAT)” is set. The “RX FIFO data available interrupt flag (RX_INTF)” can also be set in the “interrupt status register (INTSTAT)”. The CPU then receives data from the “receive data register (RXREG)”. If a “receive burst request (RXBREQ)” is received, the DMA will read data from RXREG.

When the master or slave attempts to write data to a full RX FIFO, the most recent data will not be written, and the “receiver overflow error interrupt flag (RXOERR_INTF)” will be generated.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.5.4 SPI Control Logic

31.5.4.1 Overview

SPI allows synchronous transmission and reception of 4 to 32 bits of data. It can be configured as either a slave or a master in a single-master environment. The clock polarity (CKPL) and clock phase (CKPH) settings can utilize all four SPI timing modes. The bit order can also be set to LSB first or MSB first.

The transmitter and receiver use the same clock. Data is output only during the rising or falling edge of the generated serial clock (SCLK) and is latched on the opposite edge of SCLK. The SCLK polarity and phase of the SPI master device and slave device shall be the same.

SPI is used for data exchange, with the data to be transmitted temporarily stored in the TX FIFO and the received data stored in the RX FIFO. If the SPI module is disabled, all data in these FIFOs will be lost; therefore, it is necessary to read the data from the RX FIFO through the “receive data register (RXREG)” before disabling the SPI module at the end of transmission.

The standard SPI protocol includes two data lines: one clock line and one chip select line, as described below:

- Master out slave in (MOSI or TX or SDA): This data line provides output data transferred from the master to the slave input.
- Master in slave out (MISO or RX): This data line provides the output data from the slave to the master input. During any specific transmission, only one slave device can transmit data.
- Serial clock (SCLK or SCL): This clock line is driven by the master and regulates the data flow. The master can transmit data at various SPI clock frequencies. The SCLK line cycles once for each transmitted bit.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Chip select (CS or SSN): This is the slave select input signal from the master, which is active low.

31.5.4.2 Master Mode

The SPI module only supports single-master environment. The sole SPI master device can initiate transmission and reception.

The core of the transmitter is the "transmit shift register". The software loads the data to be transmitted into the "transmit buffer register (SPI_TXREG)" via the APB bus or DMA bus, and TXREG immediately transmits the data to the TX FIFO. Once the transmission starts, TSR retrieves data from the TX FIFO; if TSR is empty, the data byte from the TX FIFO is immediately transferred to TSR. TSR is not mapped to data memory, so it cannot be accessed by the user.

Once there are enough vacancies in TX FIFO, the "TX FIFO vacancy available interrupt flag (TX_INTF)" is set to 1. When all the data in TX FIFO and the "transmit shift register" has been sent, the "transmitter empty interrupt flag (txept_intf)" is set to 1. It should be noted that whether these interrupt flags are set to 1 is independent of the state of the "interrupt enable register (SPI_TXIEN)". To read the enabled interrupt status, the "masked interrupt status register (SPI_MINTSTAT)" can be read, and writing 1 to the "interrupt clear register (SPI_INTCLR)" will clear both masked and unmasked interrupt flags.

The master, controlling the SPI clock (SCLK), can initiate data transmission at any time. In master mode, once data is written to the TXREG register, the transmission and reception of data begin. If only receiving data is intended, the transmit function can be disabled, and the receive shift register (RSR) will still shift the signal on the MISO pin at the SCLK frequency. After receiving each SPI character, RSR will save the SPI character into RX FIFO, and the "interrupt status register (SPI_INTSTAT)" and "current status register (SPI_CSTAT)" will change accordingly.

The SCLK clock is only valid during data transmission.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.5.4.3 Slave Mode

The SPI slave receives signals from the SPI master. Data is transmitted and received when an external clock pulse appears on the SCLK pin. This external clock must meet the minimum hold time requirements for high and low levels as specified in the electrical specifications.

To prepare for data transmission, the SPI slave must write the data to be transmitted into the TXREG register before the transmission starts. The TXREG register writes the data into TX FIFO, which is then transmitted to the “transmit shift register (TSR)”. Once the master issues SCLK, the slave shifts the data to be transmitted out from the MISO pin while shifting the received data into the MOSI pin.

31.5.5 Data Concatenation

By setting the register SPI_CCTL[7:6], the data concatenation feature can be used to concatenate short characters into a word in the TX FIFO and RX FIFO. The concatenation format is as follows:

1. If SPI_CCTL[12:8] ≤ 7, then every four SPI characters will be concatenated into one word (32 bits).

   Example 1: SPI_CCTL[12:8] = 5, each SPI character is 6 bits long, the concatenation format is as follows:

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

Figure 31-1: Data Concatenation Example 1 (SPI_CCTL[12:8] = 5)

2. If 8 ≤ SPI_CCTL[12:8] ≤ 15, then every two SPI characters will be concatenated into one word.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Example 2: SPI_CCTL[12:8] = 13, each SPI character is 14 bits long, the concatenation format is as follows:

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

Figure 31-2: Data Concatenation Example 2 (SPI_CCTL[12:8] = 13)

3. If 16 ≤ SPI_CCTL[12:8] ≤ 31, then every one SPI character will be concatenated into one word.

   Example 3: SPI_CCTL[12:8] = 28, each SPI character is 29 bits long, the concatenation format is as follows:

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

Figure 31-3: Data Concatenation Example 3 (SPI_CCTL[12:8] = 28)

31.5.6 Interface Timing

31.5.6.1 Motorola Timing Mode

The Motorola timing mode refers to the conventional timing mode.

Table 31-2: Motorola Timing Mode

Mode	Clock Polarity (CKPL)	Clock Phase (CKPH)
0	0: Clock is low when idle.	0: Data sampled on the first edge (rising edge) of the clock
1	0: Clock is low when idle.	1: Data sampled on the second edge (falling edge) of the clock
2	1: Clock is high when idle.	0: Data sampled on the first edge (falling edge) of the clock
3	1: Clock is high when idle	1: Data sampled on the second edge (rising edge) of the clock

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

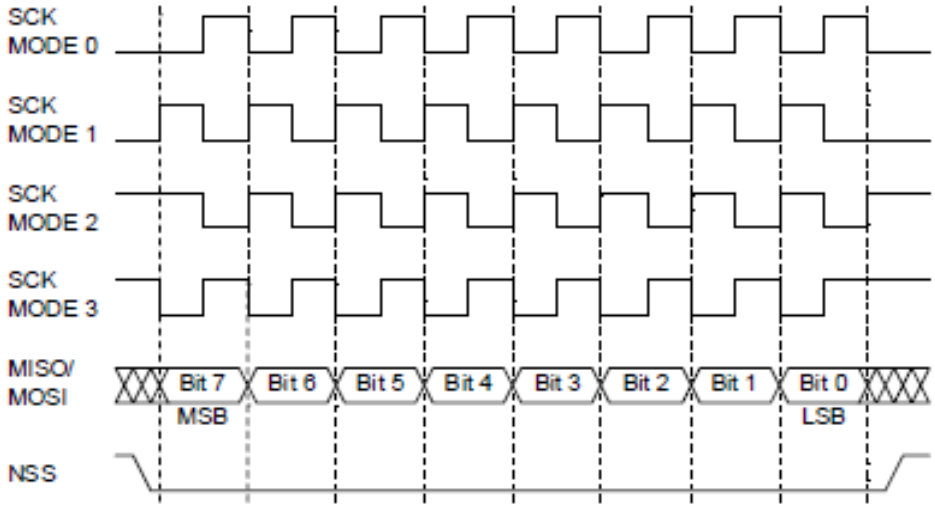

Figure 31-4: Transmission Timing Diagram of SPI in Different Modes in Motorola Mode

Note: SCK refers to the serial clock (SCLK), and NSS refers to the chip select (SSN).

31.5.6.2 TI Timing Mode

In TI timing mode, CKPL is the same as in the Motorola mode, but CKPH is the opposite. Furthermore, as shown in Figure 31-5, before transmitting each piece of data, the chip select (SSN, i.e., SPI_CS_N in Figure 31-5) signal shall be pulled high for one serial clock cycle. This module only supports single transfer format and does not support continuous transfer format.

Table 31-3: TI Timing Mode

Mode	Clock Polarity (CKPL)	Clock Phase (CKPH)
0	0: Clock is low when idle.	1: Data sampled on the second edge (falling edge) of the clock
1	0: Clock is low when idle.	0: Data sampled on the first edge (rising edge) of the clock
2	1: Clock is high when idle.	1: Data sampled on the second edge (rising edge) of the clock
3	1: Clock is high when idle.	0: Data sampled on the first edge (falling edge) of the clock

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

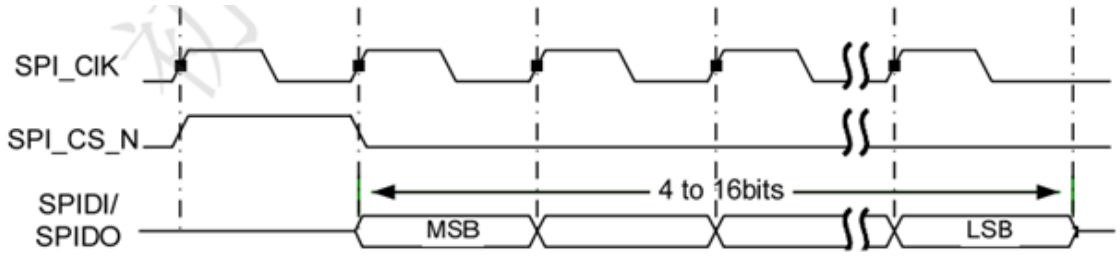

Figure 31-5: Transmission Timing Diagram of SPI in MODE2 in TI Mode

Note: SPI_CLK refers to the serial clock (SCLK). SPI_CS_N refers to the chip select (SSN). SPIDI/SPIDO refer to the serial data MISO and MOSI. This SPI module supports SPI character lengths of 1 to 32 bits.

Application note: The master and slave must maintain consistent timing modes, clock polarities and clock phases.

31.6 Register Description

SPI1 register base address: 0x4600_1000

SPI2 register base address: 0x4700_0000

SPI3 register base address: 0x4700_1000

SPI4 register base address: 0x4700_2000

The registers are listed below:

Table 31-4: List of SPI Registers

Offset Address	Name	Description
0x00	SPI_TXREG	Transmit data register
0x04	SPI_RXREG	Receive data register
0x08	SPI_CSTAT	Current status register
0x0C	SPI_INTSTAT	Interrupt status register
0x10	SPI_MINTSTAT	Masked interrupt status register
0x14	SPI_INTEN	Interrupt enable register
0x18	SPI_INTCLR	Interrupt clear register
0x1C	SPI_GCTL	Global control register

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x20	SPI_CCTL	General control register
0x24	SPI_SPBRG	SPI clock frequency control register
0x28	SPI_RXDNR	Receive data number register
0x2C	SPI_TXDNR	Transmit data number register
0x30	SPI_SCSR	Slave chip select register

31.6.1 Transmit Data Register (SPI_TXREG)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	TXREG	R/W	0x0	This register stores the latest data written into the TX FIFO. The number of valid data bits depends on the value of SPI_CCTL[12:8] and data concatenation.

31.6.2 Receive Data Register (SPI_RXREG)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RXREG	R	0x0	This register stores the last data shifted out from the RX FIFO. The number of valid data bits depends on the value of SPI_CCTL[12:8] and data concatenation. This register is read-only.

31.6.3 Current Status Register (SPI_CSTAT)

Offset address: 0x08

Reset value: 0x0000 0001

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3	RXAVL_4BYTE	R	0x0	RX FIFO has a flag with 4 words of available data. This bit is set to 1 when the RX FIFO has received at least 4 words of available data. 0: Less than 4 words of data in RX FIFO 1: 4 words or more data in RX FIFO
2	TXFULL	R	0x0	TX FIFO full status flag: 0: TX FIFO not full 1: TX FIFO full
1	RXAVL	R	0x0	Available data received flag: This bit is set to 1 when the RX FIFO has received a complete word of data. 0: RX FIFO empty 1: RX FIFO has received a complete word of data.
0	TXEPT	R	0x1	Transmitter empty status flag: 0: Transmitter not empty 1: Both TX FIFO and transmit shift register are empty.

31.6.4 Interrupt Status Register (SPI_INTSTAT)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	TXMATCH_INTF	R	0x0	Transmission complete interrupt flag: When the amount of data actually transmitted matches the amount of data to be transmitted in the SPI_TXDNR register, the transmitting process is completed and an interrupt is generated. This interrupt can occur in master single

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				transmission mode, master duplex mode, or slave single transmission mode. 0: No transmission complete interrupt generated 1: Transmission complete interrupt generated
6	TXEPT_INTF	R	0x0	Transmitter empty interrupt flag: An interrupt is generated when the data in the TX FIFO and the transmit shift register is completely sent out, or when the transmitter function is disabled, causing the TX FIFO and TSR data to be cleared. 0: No transmitter empty interrupt generated 1: Transmitter empty interrupt generated
5	RXFIFO_FULL_INTF	R	0x0	RX FIFO full interrupt flag: An interrupt is generated when RX FIFO is full. 0: No RX FIFO full interrupt generated 1: RX FIFO full Interrupt generated
4	RXMATCH_INTF	R	0x0	Reception complete interrupt flag: When the amount of data actually received matches the amount of data to be received in the SPI_RXDNR register, the receiving process is completed and an interrupt is generated. This interrupt can occur in master single reception mode, slave single reception mode, or slave duplex mode. 0: No reception complete interrupt generated 1: Reception complete interrupt generated
3	RXOERR_INTF	R	0x0	Receiver overflow error interrupt flag: An overflow error occurs when the master or slave attempts to write data into a full

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				RX FIFO, causing the latest data to be lost. 0: No receiver overflow error interrupt generated 1: Receiver overflow error interrupt generated
2	UNDERRUN_INTF	R	0x0	Slave transmitter underrun interrupt flag: An underrun error occurs when the SPI slave attempts to send a new character while in an idle state. 0: No slave transmitter underrun error interrupt generated 1: Slave transmitter underrun error interrupt generated
1	RX_INTF	R	0x0	RX FIFO data available interrupt flag: This bit is set when reception is enabled and the RX FIFO has enough data (depending on “rxtlf”). 0: No RX FIFO data available interrupt generated 1: RX FIFO data available interrupt generated
0	TX_INTF	R	0x0	TX FIFO vacancy available interrupt flag: This bit is set when transmission is enabled and the TX FIFO has enough vacancies (depending on “txtlf”). 0: No TX FIFO vacancy available interrupt generated 1: TX FIFO vacancy available interrupt generated

Note: Setting a bit in the interrupt enable register (SPI_INTEN) to 0 can prevent the corresponding interrupt signal from being generated, but it cannot prevent the corresponding flag in the register from being set to 1 when the condition is met. Writing 1 to a bit in the interrupt clear register (SPI_INTCLR) will clear the corresponding bit in the register.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.6.5 Masked Interrupt Status Register (SPI_MINTSTAT)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	TXMATCH_MINTF	R	0x0	Transmission complete interrupt flag: When the amount of data actually transmitted matches the amount of data to be transmitted in the SPI_TXDNR register, the transmitting process is completed and an interrupt is generated. This interrupt can occur in master single transmission mode, master duplex mode, or slave single transmission mode. 0: No transmission complete interrupt generated 1: Transmission complete interrupt generated
6	TXEPT_MINTF	R	0x0	Transmitter empty interrupt flag: An interrupt is generated when both TX FIFO and the transmit shift register are empty. 0: No transmitter empty interrupt generated 1: Transmitter empty interrupt generated
5	RXFIFO_FULL_MINTF	R	0x0	RX FIFO full interrupt flag: An interrupt is generated when RX FIFO is full. 1: RX FIFO full Interrupt generated 0: No RX FIFO full interrupt generated

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	RXMATCH_MINTF	R	0x0	Reception complete interrupt flag: When the amount of data actually received matches the amount of data to be received in the RXDNR register, the receiving process is completed and an interrupt is generated. This interrupt can occur in master single reception mode, slave single reception mode, or slave duplex mode. 0: No reception complete interrupt generated 1: Reception complete interrupt generated
3	RXOERR_MINTF	R	0x0	Receiver overflow error interrupt flag: An overflow error occurs when the master or slave attempts to write data into a full RX FIFO, causing the latest data to be lost. 0: No receiver overflow error interrupt generated 1: Receiver overflow error interrupt generated
2	UNDERRUN_MINTF	R	0x0	Slave transmitter underrun interrupt flag: An underrun error occurs when the SPI slave attempts to send a new character while in an idle state. 0: No slave transmitter underrun error interrupt generated 1: Slave transmitter underrun error interrupt generated

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	RX_MINTF	R	0x0	Receiver data available interrupt flag: This bit is set to 1 when the RX FIFO has received sufficient data (depending on SPI_GCTL[6:5]). 0: No receiver data available interrupt generated 1: Receiver data available interrupt generated
0	TX_MINTF	R	0x0	TX FIFO vacancy available interrupt flag: This bit is set to 1 when there is sufficient vacancies in the TX FIFO (depending on SPI_GCTL[8:7]). 0: No TX FIFO vacancy available interrupt generated 1: TX FIFO vacancy available interrupt generated

Note: This register is a shadow register, which maps the result of the logical AND interrupt enable register (SPI_INTEN) of the interrupt status register (SPI_INTSTAT). If an interrupt is not enabled, the corresponding interrupt flag in this register will not be set.

31.6.6 Interrupt Enable Register (SPI_INTEN)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7	TXMATCHEN	R/W	0x0	Transmission complete interrupt enable: 0: Disabled 1: Enabled
6	TXEPT_IEN	R/W	0x0	Transmitter empty interrupt enable: 0: Disabled 1: Enabled

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	RXFIFO_FULL_IEN	R/W	0x0	RX FIFO full interrupt enable: 0: Disabled 1: Enabled
4	RXMATCHEN	R/W	0x0	Reception complete interrupt enable: 0: Disabled 1: Enabled
3	RXOERREN	R/W	0x0	Receiver overflow error interrupt enable: 0: Disabled 1: Enabled
2	UNDERRUNEN	R/W	0x0	Slave transmitter underrun interrupt enable: 0: Disabled 1: Enabled
1	RXIEN	R/W	0x0	RX FIFO data available interrupt enable: 0: Disabled 1: Enabled
0	TXIEN	R/W	0x0	TX FIFO vacancy available interrupt enable: 0: Disabled 1: Enabled

Note: Setting a bit in this register to 0 can prevent the corresponding interrupt signal from being generated, but it cannot prevent the corresponding flag in the interrupt status register (SPI_INTSTAT) from being set to 1 when the condition is met.

31.6.7 Interrupt Clear Register (SPI_INTCLR)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7	TXMATCHCLR	W	0x0	Transmission complete interrupt clear: 0: Not cleared 1: Cleared
6	TXEPT_ICLR	W	0x0	Transmitter empty interrupt clear: 0: Not cleared 1: Cleared
5	RXFIFO_FULL_ICLR	W	0x0	RX FIFO full interrupt clear: 0: Not cleared 1: Cleared
4	RXMATCHCLR	W	0x0	Reception complete interrupt clear: 0: Not cleared 1: Cleared
3	RXOERRCLR	W	0x0	Receiver overflow error interrupt clear: 0: Not cleared 1: Cleared
2	UNDERRUNCLR	W	0x0	Slave transmitter underrun interrupt clear: 0: Not cleared 1: Cleared
1	RXICLR	W	0x0	RX FIFO data available interrupt clear: This bit is valid only when I2S_GCR[13] = 0. 0: Not cleared 1: Cleared
0	TXICLR	W	0x0	TX FIFO vacancy available interrupt clear: This bit is valid only when I2S_GCR[13] = 0. 0: Not cleared 1: Cleared

Writing 1 to a bit in this register will clear the corresponding bit in the interrupt status register (SPI_INTSTAT).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.6.8 Global Control Register (SPI_GCTL)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	HALF_DIR	R/W	0x0	Half-duplex mode data output enable: 0: Data input 1: Data output
11	BIDIRMODE	R/W	0x0	Half-duplex mode enable: In the master half-duplex mode, MOSI acts as a bidirectional data line; in the slave half-duplex mode, MISO acts as a bidirectional data line. The direction of data transmission in half-duplex mode depends on “txen & !rxen”, i.e., when txen == 1 and rxen == 0, the pin outputs data; otherwise, the pin inputs data. For full-duplex mode, simply configure the unused pin for GPIO function. 0: Full-duplex mode or simplex mode 1: Half-duplex mode
10	CSN_SEL	R/W	0x0	Chip select signal controller selection: In the Motorola timing mode, the chip select signal can be controlled by software or hardware. Under hardware control, the chip select signal will still be consistent with the SPI_SCSR register during data transmission; however, after transmitting or receiving each SPI character, the chip select signal will be pulled high for one PCLK cycle and then pulled low again to transmit the next SPI character. When all data has been transmitted, the chip select signal will be automatically pulled high.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				In the TI timing mode, the chip select signal is controlled jointly by software and hardware, and this bit is invalid. The chip select signal will be consistent with the SPI_SCSR register during idle or data transmission; however, before transmitting or receiving each SPI character, the chip select signal will be pulled high for one SCLK cycle and then pulled low again to transmit that SPI character. 0: Chip select signal is configured by the SPI_SCSR register. 1: Chip select signal is controlled by hardware.
9	DMAMODE	R/W	0x0	DMA access mode enable: 0: Use CPU access mode (CPU reads and writes TX FIFO and RX FIFO) 1: Use DMA access mode (DMA reads and writes TX FIFO and RX FIFO)
8:7	TXTLF	R/W	0x0	Selection of the number of empty bits that can trigger the TX FIFO to receive data: 00: TX FIFO with at least 1 empty bit (default) 01: TX FIFO with at least 4 empty bits 10: TX FIFO with at least 8 empty bits 11: TX FIFO with at least 16 empty bits (unavailable)
6:5	RXTLF	R/W	0x0	Selection of the number of data that can trigger the RX FIFO to transmit data: 00: RX FIFO with at least 1 data (default) 01: RX FIFO with at least 4 data 10: RX FIFO with at least 8 data 11: RX FIFO with at least 16 data (unavailable)
4	RXEN	R/W	0x0	Receive enable:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Data reception disabled and RX FIFO cleared 0: Data reception enabled
3	TXEN	R/W	0x0	Transmit enable: 0: Data transmission disabled and TX FIFO cleared 1: Data transmission enabled
2	MM	R/W	0x0	Master/slave mode selection: 0: Slave mode (serial clock generated by external module) 1: Master mode (serial clock generated by this module)
1	INT_EN	R/W	0x0	SPI interrupt enable: 0: All SPI interrupts disabled 1: SPI interrupts enabled (SPI_INTEN register also needs to be configured)
0	EN	R/W	0x0	SPI enable: 0: SPI disabled (most circuitry remains in reset state.) 1: SPI enabled

31.6.9 General Control Register (SPI_CCTL)

Offset address: 0x20

Reset value: 0x0000 0700

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12:8	SPILEN	R/W	0x7	SPI character length control: define the length of each SPI character as (spilen + 1) bits; support character lengths from 1 to 32 bits, with a default of 8 bits.
7	RX_STITCH	R/W	0x0	Receive data concatenation enable: 0: Disabled 1: Enabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	TX_STITCH	R/W	0x0	Transmit data concatenation enable: 0: Disabled 1: Enabled
5	LSBFE	R/W	0x0	Transmission bit order selection: 0: Transmit or receive data with MSB first 1: Transmit or receive data with LSB first
4	RXEDGE	R/W	0x0	Data receive phase adjustment in master mode: 0: Data sampled on the receive edge of SCLK 1: Data sampled on the next transmit edge of SCLK (delayed)
3	TXEDGE	R/W	0x0	Data transmit phase adjustment in slave mode: Note: When 2 $f_{SCLK} \le f_{PCLK} < 3 f_{SCLK}$, this bit can only be set to 1; when $f_{PCLK} \ge 3 f_{SCLK}$, this bit is recommended to be set to 0. 0: Transmit data changed on the transmit edge of SCLK 1: Transmit data changed on the rising edge of PCLK (in advance)
2	TI_MOD	R/W	0x0	Timing mode selection: In TI timing mode, an extra idle SCLK cycle is required to pull SSN high before transmitting each character, and its CKPH is opposite to that in Motorola timing mode. 0: Motorola timing mode 1: TI timing mode
1	CKPL	R/W	0x0	SCLK polarity: The clock polarity determines whether the clock is high or low when idle. 0: SCLK is low when idle. 1: SCLK is high when idle.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	CKPH	R/W	0x0	SCLK phase: 0: In Motorola mode, data is sampled on the first SPI clock edge and changed on the second clock edge; in TI mode, data is changed on the first SPI clock edge and sampled on the second clock edge. 1: In Motorola mode, data is changed on the first SPI clock edge and sampled on the second clock edge; in TI mode, data is sampled on the first SPI clock edge and changed on the second clock edge.

31.6.10 SPI Clock Frequency Control Register (SPI_SPBRG)

Offset address: 0x24

Reset value: 0x0000 0002

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	SPBRG	R/W	0x2	This register is used to control the SPI clock frequency. The SPI clock frequency formula is: $f_{SCLK} = f_{PCLK} / SPBRG$ Where f_{PCLK} is the APB clock frequency. Note: Do not write 0 or 1 to this register; writing 0 or 1 is invalid.

31.6.11 Receive Data Number Register (SPI_RXDNR)

Offset address: 0x28

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	RXDNR	R/W	0x1	This register is used to store the number of characters (not concatenated) to be received

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				in the next reception. **SPI master mode:** This register is valid only in simplex receive mode and will affect the number of cycles of the generated SPI clock, so it must be properly configured. **SPI slave mode** (valid only for character concatenation): This register is valid in both simplex receive mode and duplex (receive and transmit) mode. In particular, in duplex mode, it indicates both the number of characters to be received and to be transmitted. The default value of this register is 1, and it will not change until the CPU writes a new value to it. Note: Do not write 0 to this register.

31.6.12 Transmit Data Number Register (SPI_TXDNR)

Offset address: 0x2C

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	TXDNR	R/W	0x1	This register is used to store **the number of characters (not concatenated) to be transmitted in the next transmission**, it is only valid for character concatenation. **SPI master mode:** This register is valid in both simplex transmit mode and duplex (receive and transmit) mode. In particular, in duplex mode, it indicates both the number of characters to be transmitted and to be received.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				SPI slave mode: This register is valid only in simplex transmit mode. The default value of this register is 1, and it will not change until the CPU writes a new value to it. Note: Do not write 0 to this register.

31.6.13 Slave Chip Select Register (SPI_SCSR)

Offset address: 0x30

Reset value: 0x0000 00FF

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CS0	R/W	0x1	The chip select output signal in master mode is active low. This register is not available in SPI slave mode: 0: The corresponding slave selected 1: The corresponding slave not selected Note: Only bit [0] is useful, and bits [7:1] are not required.

31.7 Operation Procedure

31.7.1 Master Mode

31.7.1.1 Master Duplex Transmit/Receive or Simplex Transmit

1. Register configuration
   A. Configure the system registers related to peripheral clock, reset, and pin alternate function, to enable the clock and release the reset.
   B. Set SPI_GCTL[0], GCTL[2] and GCTL[3] to 1, and configure other bits as required.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

C. Configure the SPI_CCTL register to set the timing mode and data format.

D. Configure the SPI_SPBRG register to set the SCLK division factor.

E. Write 0xFF to the SPI_INTCLR register to clear the interrupt status of the SPI_INTSTAT register.

F. Configure the SPI_INTEN register to optionally enable interrupts.

G. When the character concatenation function is enabled, configure the SPI_TXDNR register to set the number of characters to be transmitted.

H. Configure the SPI_SCSR register to pull down the chip select signal corresponding to the selected slave.

2. Data transmission

A. If data needs to be transmitted, write the data to be transmitted into the SPI_TXREG register through CPU or DMA.

B. When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the SPI_RXREG register through CPU or DMA.

3. Interrupt handling

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

A. Wait for SPI_INTSTAT[6] to become 1, or wait for SPI_CSTAT[0] to become 1. The rate of data transmission may be slower than the rate of writing to the TX FIFO, so writing the data to be transmitted to the SPI_TXREG register does not mean that all the data has been transmitted. If the transmitter is not empty, wait for all data in the transmitter to be transmitted before concluding that the transmission is complete.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

B. Configure the SPI_SCSR register to pull up the chip select signal. If hardware control of the chip select signal is chosen, this register may not need to be configured.

C. Configure SPI_GCTL[0] to 0 to disable the SPI module.

D. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.1.2 Master Half-duplex Receive

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function (the master using MOSI for communication), to enable the clock and release the reset.

   B. Configure the SPI_GCTL register, noting that SPI_GCTL[4] and SPI_GCTL[3] shall not be set to 1 temporarily; set SPI_GCTL[11], SPI_GCTL[0] and SPI_GCTL[2] to 1, and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Configure the SPI_SPBRG register to set the SCLK division factor.

   E. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

   F. Configure the SPI_INTEN register to optionally enable interrupts.

   G. Configure the SPI_SCSR register to pull down the chip select signal corresponding to the selected slave.

2. Data transmission

   A. Set SPI_GCTL[12] and SPI_GCTL[3] to 0, and set SPI_GCTL[4] and SPI_GCTL[0] to 1.

   B. When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the RXREG register through CPU or DMA.

   C. Set SPI_GCTL[4] to 0.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3. Interrupt handling

   If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

   A. Configure the SPI_SCSR register to pull up the chip select signal. If hardware control of the chip select signal is chosen, this register may not need to be configured.

   B. Configure SPI_GCTL[0] to 0 to disable the SPI module.

   C. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.1.3 Master Half-duplex Transmit

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function (the master using MOSI for communication), to enable the clock and release the reset.

   B. Configure the SPI_GCTL register, noting that SPI_GCTL[4] and SPI_GCTL[3] shall not be set to 1 temporarily; set SPI_GCTL[11], SPI_GCTL[0] and SPI_GCTL[2] to 1, and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Configure the SPI_SPBRG register to set the SCLK division factor.

   E. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

   F. Configure the SPI_INTEN register to optionally enable interrupts.

   G. Configure the SPI_SCSR register to pull down the chip select signal corresponding to the selected slave.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

2. Data transmission

   A. Set SPI_GCTL[0] and SPI_GCTL[4] to 0, and set SPI_GCTL[12] and SPI_GCTL[3] to 1.

   B. When SPI_CSTAT[2] is 0, it indicates that the TX FIFO is not full, and data can be written into the SPI_TXREG register.

3. Interrupt handling

   If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

   A. When SPI_CSTAT[0] is 1, it indicates that data has been transmitted from the SPI_TXREG register.

   B. Configure the SPI_SCSR register to pull up the chip select signal. If hardware control of the chip select signal is chosen, this register may not need to be configured.

   C. Set SPI_GCTL[0] to 0 to disable the SPI module.

   D. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.1.4 Master Simplex Receive

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function, to enable the clock and release the reset.

   B. Configure the SPI_GCTL register, noting that SPI_GCTL[4] shall not be set to 1 temporarily; set SPI_GCTL[0] and SPI_GCTL[2] to 1, and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Configure the SPI_SPBRG register to set the SCLK division factor.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

E. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

F. Configure the SPI_INTEN register to enable interrupts.

G. Configure the SPI_RXDNR register to set the number of characters to be received.

H. Configure the SPI_SCSR register to pull down the chip select signal corresponding to the selected slave.

I. Configure the SPI_GCTL register and set SPI_GCTL[4] to 1.

2. Data transmission

When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the SPI_RXREG register through CPU or DMA.

3. Interrupt handling

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

A. Since the master controls the clock and chip select signal, when the transmission ends is also controlled by the master. When character concatenation is enabled, the slave may wait for SPI_INTSTAT[4] to become 1, indicating that the specified number of SPI characters set in the SPI_RXDNR register has been received, i.e., the reception is complete. At the same time, set SPI_GCTL[4] to 0.

B. Configure the SPI_SCSR register to pull up the chip select signal. If hardware control of the chip select signal is chosen, this register may not need to be configured.

C. Configure SPI_GCTL[0] to 0 to disable the SPI module.

D. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

31.7.2 Slave Mode

31.7.2.1 Slave Duplex Transmit/Receive or Simplex Receive

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function, to enable the clock and release the reset.

   B. Set SPI_GCTL[0] and SPI_GCTL[4] to 1, set SPI_GCTL[2] to 0 and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

   E. Configure the SPI_INTEN register to enable interrupts.

   F. When the character concatenation function is enabled, configure the SPI_RXDNR register to set the number of characters to be received.

2. Data transmission

   A. If data needs to be transmitted, write the data to be transmitted into the SPI_TXREG register through CPU or DMA. Note: The slave must be prepared with the data to be transmitted before the master pulls the chip select signal low.

   B. When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the SPI_RXREG register through CPU or DMA.

3. Interrupt handling

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

   A. Since the master controls the clock and chip select signal, when the transmission ends is also controlled by the master. When character concatenation is enabled, the slave may wait for SPI_INTSTAT[4] to become 1, indicating that the specified number of SPI characters set in the SPI_RXDNR register has been received, i.e., the reception is complete. At the same time, set SPI_GCTL[4] to 0.

   B. After the transmission ends, configure SPI_GCTL[0] to 0 to disable the SPI module.

   C. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.2.2 Slave Half-duplex Receive

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function (the slave using MISO for communication), to enable the clock and release the reset.

   B. Configure the SPI_GCTL register, noting that SPI_GCTL[4] and SPI_GCTL[3] shall not be set to 1 temporarily, set SPI_GCTL[11] and SPI_GCTL[0] to 1, set SPI_GCTL[2] to 0, and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

E. Configure the SPI_INTEN register to optionally enable interrupts.

2. Data transmission

A. Set SPI_GCTL[12] and SPI_GCTL[3] to 0, and set SPI_GCTL[4] and SPI_GCTL[0] to 1.

B. When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the SPI_RXREG register through CPU or DMA.

3. Interrupt handling

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

A. Since the master controls the clock and chip select signal, when the transmission ends is also controlled by the master.

B. After the transmission ends, configure SPI_GCTL[0] to 0 to disable the SPI module.

C. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.2.3 Slave Half-duplex Transmit

1. Register configuration

A. Configure the system registers related to peripheral clock, reset, and pin alternate function (the slave using MISO for communication), to enable the clock and release the reset.

B. Configure the SPI_GCTL register, noting that SPI_GCTL[4] and SPI_GCTL[3] shall not be set to 1 temporarily, set SPI_GCTL[11] and SPI_GCTL[0] to 1, set SPI_GCTL[2] to 0, and

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

configure other bits as required.

C. Configure the SPI_CCTL register to set the timing mode and data format.

D. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

E. Configure the SPI_INTEN register to optionally enable interrupts.

2. Data transmission

A. If data needs to be transmitted, write the data to be transmitted into the SPI_TXREG register through CPU or DMA. Note: The slave must be prepared with the data to be transmitted before the master pulls the chip select signal low.

B. Set SPI_GCTL[12] and SPI_GCTL[3] to 1, set SPI_GCTL[4] to 0, and then set SPI_GCTL[0] to 1.

C. Write data into the SPI_TXREG register.

D. When SPI_CSTAT[0] is 1, it indicates that the data has been transmitted.

E. Set SPI_GCTL[3] to 0.

3. Interrupt handling

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

A. Since the master controls the clock and chip select signal, when the transmission ends is also controlled by the master. It is recommended for the slave to wait for SPI_INTSTAT[6] to become 1, or wait for SPI_CSTAT[0] to become 1, indicating that all data in the transmitter has been transmitted.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

B. After the transmission ends, configure SPI_GCTL[0] to 0 to disable the SPI module.

C. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

31.7.2.4 Slave Simplex Transmit

1. Register configuration

   A. Configure the system registers related to peripheral clock, reset, and pin alternate function, to enable the clock and release the reset.

   B. Set SPI_GCTL[0] and SPI_GCTL[3] to 1, set SPI_GCTL[2] to 0 and configure other bits as required.

   C. Configure the SPI_CCTL register to set the timing mode and data format.

   D. Write 0xFF to the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

   E. Configure the SPI_INTEN register to enable interrupts.

   F. When the character concatenation function is enabled, configure the SPI_TXDNR register to set the number of characters to be transmitted.

2. Data transmission

   A. If data needs to be transmitted, write the data to be transmitted into the SPI_TXREG register through CPU or DMA. Note: The slave must be prepared with the data to be transmitted before the master pulls the chip select signal low.

   B. When SPI_CSTAT[1] is 1, it indicates that data has been received. If data needs to be received, read the received data from the SPI_RXREG register through CPU or DMA.

3. Interrupt handling

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

If an interrupt occurs, read the SPI_MINTSTAT register to determine the cause of the interrupt. After processing, write 1 to the corresponding bit of the SPI_INTCLR register to clear the interrupt status in the SPI_INTSTAT and SPI_MINTSTAT registers.

4. End of transmission

   A. Since the master controls the clock and chip select signal, when the transmission ends is also controlled by the master. It is recommended for the slave to wait for SPI_INTSTAT[6] to become 1, or wait for SPI_CSTAT[0] to become 1, indicating that all data in the transmitter has been transmitted.

   B. After the transmission ends, configure SPI_GCTL[0] to 0 to disable the SPI module.

   C. Configure the system registers related to peripheral clock and reset to disable the clock and restore the reset.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32 Quad-SPI Controller (QSPI)

32.1 Overview

The QSPI is a specialized communication interface targeting single, dual or quad SPI Flash memories.

32.2 Main Features

- Memory-mapped direct operation mode for Flash data transmission and encoding in Flash memory
- Software-triggered indirect operation mode for non-processor-intensive Flash data transmission with short latency
- Software APB accessible Flash control register group capable of executing any Flash command, including data transfers of up to 8 bytes per transaction
- Supporting XIP (Execute in Place), also known as continuous mode
- Supporting single-SPI, dual-SPI and quad-SPI modes
- Programmable device size
- Programmable write protection areas preventing system from writing to designated areas
- Programmable delay between transfer transactions
- Allowing direct software access to underlying TX and RX FIFOs in traditional mode, bypassing higher-layer processes
- Supporting independent asynchronous SPI communication reference clock
- Serial clock with configurable polarity
- Programmable baud rate generator
- With feature for enhancing the capture mechanism of high-speed data reading

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Optional use of an adjusted clock to further enhance read data capture
- Programmable interrupt generation
- Supporting one external device

32.3 System Block Diagram

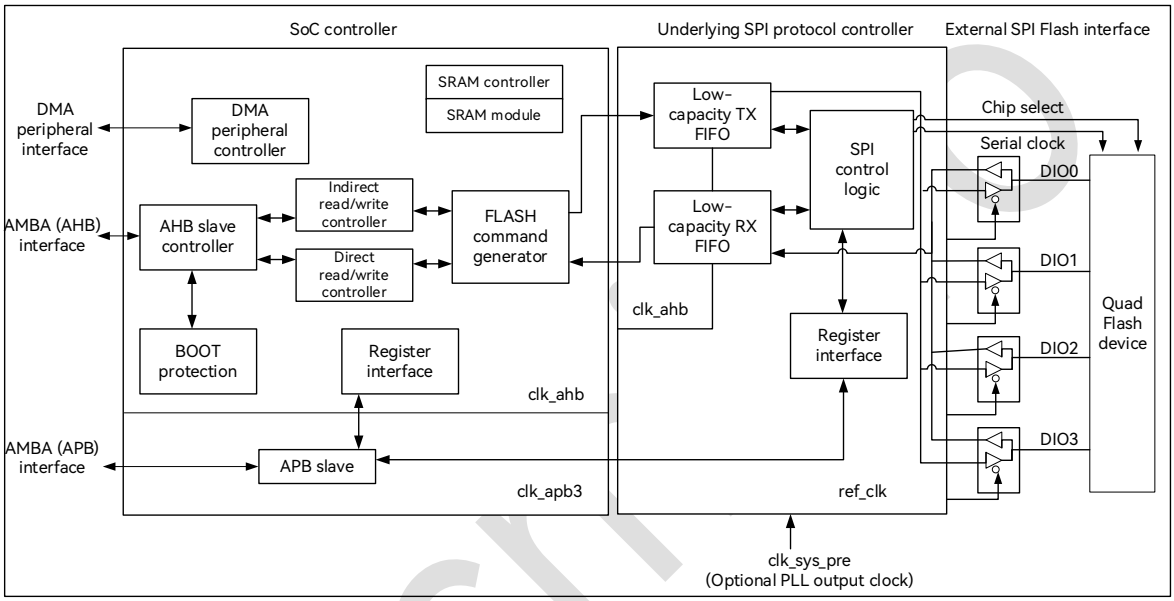

Figure 32-1: QSPI System Block Diagram

32.4 Pin Description

Table 32-1: QSPI Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
QSPI_SCK	PE10, PB10	Output	Serial clock output
QSPI_DIO0	PE12, PD4	Input/output	Serial data input/output signal
QSPI_DIO1	PE13, PD5	Input/output	Serial data input/output signal
QSPI_DIO2	PE14, PD6	Input/output	Serial data input/output signal
QSPI_DIO3	PE15, PD7	Input/output	Serial data input/output signal
QSPI_CSN	PA2, PE11, PD3	Output	Slave device select line

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.5 Functional Description

32.5.1 AHB Control Interface

The AHB slave controller can verify the received AHB access, respond to invalid requests, perform reordering of arbitrary bytes or half-words, mask write operations that violate write protection rules (only for direct access), and forward transfer requests to the direct access controller or the indirect access controller.

32.5.1.1 AHB Interface

The AHB interface complies with the ARM AMBA 3 AHB-Lite protocol specification. Locked transfers (HMASTLOCK) and transfers using protection control signals (HPROT) are not supported. The AHB data width is 32 bits, allowing only byte, half-word, and word accesses. For write operations, only incrementing bursts are supported; wrapping write bursts received will generate an error. The supported burst types include INCR16, INCR8, INCR4, INCR and SINGLE. For read operations, all burst types, including WRAPS, are supported. If a burst transfer is prematurely terminated due to interconnect burst termination, or if an error access occurs in the slave during the burst transfer, the slave will still operate correctly.

32.5.1.2 AHB Address Remapping

The QSPI controller does not perform specific address decoding for received erroneous addresses, but it does have an AHB address decoder. If enabled, the QSPI Flash controller can detect the address range of each individual device and make an automatic device switching choice based on the AHB address. The received AHB address is by default directly mapped to the address sent serially to the Flash device. If the Flash device has a 24-bit address, the lower 24 bits of the AHB address will be sent. The remapping function maps the address on the AHB

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

bus to ADDRESS + N, where N is the value in the address remapping register (QSPI_RAR). The remapping function can be enabled through the QSPI configuration register (QSPI_CR[0]); and it is required when software needs to move the BOOT code to another Flash area.

32.5.1.3 Write Protection

To protect the Flash device, the write protection function can be implemented by hardware or controlled by software. Any detected AHB write operation directed to a protected Flash area will generate an error signal through the HRESP error output. A region on the Flash device can be write-protected if it is defined by multiple Flash “blocks” and with a specific starting “block” number. Three programmable registers are provided. The QSPI write protection lower register (QSPI_WPLR) defines the Flash block at the bottom of the area to be protected. The QSPI write protection upper register (QSPI_WPHR) defines the Flash block at the top of the area to be protected. The QSPI write protection configuration register (QSPI_WPCR) is a 2-bit control register. QSPI_WPCR[0] can invert the protected area, making it the only unprotected area in the Flash memory. QSPI_WPCR[1] is the write protection enable bit. When QSPI_WPCR[1] is at a low level, the Flash device is not protected. The AHB controller is required to map the received AHB address to the relevant Flash block. The size of a block can be set through the device size configuration register (QSPI_DSCR), with a range of 1 to 65K bytes.

32.5.1.4 Access Forwarding

To ensure legitimate access, the AHB controller will forward all AHB accesses to one of the two access controllers: the direct access controller or the indirect access controller. Assuming that the direct access controller has been enabled through the QSPI configuration register (QSPI_CR), by default, all AHB accesses will be forwarded to the direct access controller. Before forwarding any access to the indirect access controller, software configuration is required. This process is fully explained in the section on the indirect access controller. If the direct access controller is

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

disabled, any AHB access that cannot be forwarded to the indirect access controller will be immediately terminated and an HRESP error will be generated. If the direct access controller is enabled, the same access will be forwarded and serviced by the direct access controller.

32.5.1.5 Sequential Access Detection and Burst Length

To maximize performance, the AHB interface signal htrans will not be used to identify sequential and non-sequential accesses. The issue with htrans is that it identifies all accesses at the start of a new AHB burst transfer as non-sequential accesses, and all accesses beyond the 1K boundary are also identified as non-sequential accesses. Instead, non-sequential accesses are detected by comparing the current address with the previous address. If the access direction (write/read) has changed, it is considered a non-sequential access. Similarly, if the size of a read access (byte/half-word/word) has changed, it is also considered a non-sequential access. If the AHB address decoder is enabled and detects a switch between devices, a new SPI transfer must be generated because the currently selected device needs to be initialized. This situation is considered a non-sequential access, even if the address is contiguous. Otherwise, if the current transfer address is contiguous with the previous transfer address, it is considered a sequential access. Additionally, the controller does not use the hburst signal of AHB to identify burst length. The burst transfer will continue until a non-sequential access is received.

32.5.2 Direct Access Controller (DAC)

Direct access refers to AHB accesses that directly trigger read or write operations to the Flash memory. This operation is memory-mapped, allowing access and direct execution of code stored in the external Flash memory. Any received AHB access that is not within the programmable indirect trigger region is assumed to be a direct access and is responded to by the direct access controller (DAC). Note that accesses using the DAC will not use the embedded

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

SRAM. When performing read or write burst operations, the AHB will throttle, with the number of wait states depending on the controller's latency, which has been designed to be as minimal as possible. When the XIP read instruction is enabled, the latency will be kept at its minimum value. When responding to an AHB read operation, the DAC will send an additional downstream access, rather than just forwarding a single AHB burst. This access will not be visible at the system interface. This operation is a prefetch operation to ensure that the underlying SPI core runs at maximum bandwidth. The AHB burst defined here is different from the AHB burst defined by AMBA. The AHB burst here is defined as follows:

1. The first access of an AHB burst is defined by the following criteria:
   - Non-sequential AHB access (determined by address comparison rather than based on htrans)
   - Non-sequential or sequential AHB access (when the downstream module is in an idle state)
2. The last access of an AHB burst is defined by a sequential AHB access, taking precedence over the new burst defined in point 1 above.
3. The size of the AHB burst is the number of AHB accesses from the first access to the last access.
4. The number of DAC requests for each AHB burst = size of the AHB burst + 1.

Regarding AHB write operations, the direct access controller will trigger a series of write commands (similar to the handling of read operations), even though the number of DAC write requests equals the number of received AHB write requests. During write operations, the AHB controller will ensure that Flash bursts do not exceed the Flash page boundary. When a page boundary is detected, only the number of byte accesses up to the boundary will be forwarded. A sequential direct write request that crosses a page boundary must be detected as a non-

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

sequential request to the downstream module, forcing the downstream controller to put the Flash device into a self-timed page program cycle. The core supports page-separated writes, limited to word-aligned addresses. If the system delays sending continuous write operations for too long, the Flash write cycle may start prematurely, reducing the effective lifespan of the device. Note that if the system cannot guarantee the timely provision of data to be written, indirect write operations shall be used to avoid this issue. Flash erase operations are triggered by software using the programming interface. Once the page program cycle is initiated, the QSPI Flash controller will automatically poll the write cycle until completion, before allowing subsequent AHB accesses to complete. This operation is achieved by holding subsequent AHB direct accesses in the wait state.

32.5.3 Indirect Access Controller (INDAC)

32.5.3.1 Indirect Read Controller

The purpose of indirect operations is to read valid byte counts from Flash memory without triggering AHB access. Indirect operations are controlled and triggered by the software through the indirect read transfer control register (QSPI_IRTR), the indirect read transfer watermark register (QSPI_IRTWR), the indirect read transfer start address register (QSPI_IRTSAR), and the indirect read transfer byte number register (QSPI_IRTNR). This module communicates with the underlying SPI protocol state machine module to perform optimized Flash read burst operations, placing the read data in the local SRAM module in preparation for fast and low-latency transfers with the external AHB host. By default, the indirect read controller is disabled. Before enabling it, software must configure the amount of data to be read and the start address. The indirect read transfer start address register (QSPI_IRTSAR) defines the start address, while the indirect read transfer byte number register (QSPI_IRTNR) defines the byte number. Two indirect operations can be configured simultaneously, with the second indirect operation being

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

triggered during the first operation. A brief turnaround time is allowed between the completion of the previous indirect operation and the start of the next one. The total number of bytes read by indirect operations is not limited by the size of the SRAM. The size of SRAM only limits the number of requests sent to DMA (in the case of DMA peripheral interface module being enabled). When an SRA overflow occurs in SRAM, the controller will push back the reading from Flash until there is available space in the SRAM. By completing the current read burst, the controller backpressures the read operations at the SPI interface, waiting until there is available space in the SRAM, and then sends a new read burst operation at the address where the previous burst operation terminated. By sending an AHB read to the QSPI controller, the bus master can obtain the data that the controller has read from the external Flash memory. The received read access AHB address must be within the range from the AHB indirect trigger address (configured via the indirect AHB address trigger register, QSPI_IATR) to (AHB indirect trigger address + 2** (indirect AHB trigger address range) - 1). The default value of this range is 16, effectively handling burst operations of 16 accesses or fewer. The actual AHB address must agree within the indirect range for the SRAM to be the source. Each valid AHB indirect read will cause the internal SRAM to pop, thereby decoupling the AHB address from the Flash address in a non-direct mapping. Therefore, the AHB indirect trigger address has no relation to the Flash address at all. After triggering any valid indirect read, the SRAM is considered the source rather than the Flash memory array. The Flash address for the indirect read is obtained from the indirect read transfer start address register (QSPI_IRTSAR). If the requested data is already in the SRAM, once the QSPI controller receives the AHB access address, the data in the SRAM will be fetched, achieving minimal latency response for the read burst. When data is read from the SRAM, the QSPI controller will release the associated resources in the SRAM. If the received AHB read access address is not within the specified range, the access will not be completed by the indirect controller but will be handled by the direct access controller. If the received AHB read access address is within the specified range but the data is not present in

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the SRAM, the AHB will be in a wait state until data is read from the Flash and pushed into the SRAM. If the accessed elements of the received AHB read burst span the AHB indirect trigger range, the accesses within the indirect trigger range will be processed by the indirect access controller, while the remaining part will be handled by the direct access controller. Note that this may indicate a software configuration error. The bus master is only allowed to send 32-bit AHB reads until the indirect transfer ends, which reduces the complexity of the SRAM control logic. When reading the last non-32-bit aligned data, the bus master can send a 16-bit (half-word) or byte access to complete the transfer. Similarly, the bus master can always send 32-bit reads, with the high bits of the non-32-bit aligned data filled with zeros.

Under ideal conditions, the SRAM will remain full during read operations. The fill level of the SRAM data can be directly read by software via the SRAM fill level status register (QSPI_SFLR). If the DMA peripheral interface controller is enabled, it will automatically request the external DMA to fetch data from SRAM in the form of effective data chunks through AHB, with each data chunk being part of an overall indirect transfer. The indirect operation can be aborted at any time by setting QSPI_IRTR[1]. Any bus master can initiate an indirect access. The DMA bus peripheral interface can share some of the software overhead and effectively manage data transfers. Another option is for the software to directly access the SRAM fill level status via the APB register, to determine when to fetch data from the SRAM. When the DMA peripheral interface is disabled, a fill level watermark is provided that can be accessed via the indirect read transfer watermark register (QSPI_IRTWR). When the SRAM fill level status exceeds this watermark, an interrupt is generated. If the fill level watermark is greater than 0, an interrupt will also be generated when the last byte of data is read by the QSPI controller and placed in the SRAM, even if the actual SRAM fill level status has not yet exceeded the watermark. This feature helps avoid the need for software to track the amount of data that has been read and reset the fill level watermark for the last few bytes of the indirect read transfer. Note that the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

fill level watermark register is a dual-purpose register. When the DMA peripheral interface is enabled, the hardware controls the rate at which DMA requests are sent. When the DMA peripheral interface is disabled, its behavior is as described above. To understand the status of the indirect operation, two additional interrupt sources are provided. The first is an interrupt generated when an indirect operation is completed. The second is an interrupt generated if an indirect read request is issued but not accepted because the QSPI controller has already cached two indirect operations. Setting QSPI_IRTR[0] initiates an indirect read operation, and QSPI_IRTR[2] can be used to check the status.

Indirect read transfer process:

When the DMA peripheral controller is enabled, the following process must be followed:

1. Set the QSPI configuration register (QSPI_CR).
2. Set the SRAM fill level watermark through the Indirect Read Transfer Watermark Register (QSPI_IRTWR).
3. Set the start address of the Flash for the indirect transfer through the indirect read transfer start address register (QSPI_IRTSAR).
4. Set the number of bytes to be transferred in the indirect read transfer byte number register (QSPI_IRTNR).
5. Set the AHB trigger address for the indirect transfer in the indirect AHB address trigger register (QSPI_IATR).
6. Set the range of the AHB trigger address for the indirect transfer in the indirect trigger address range register (QSPI_ITARR).
7. Set the number of bytes for single and burst transfers in the DMA peripheral configuration register (QSPI_DMACR).

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8. Set QSPI_IRTR[0] = 1 to trigger the indirect read access.
9. Poll the completion status of the indirect read operation through QSPI_IRTR[5] in the indirect read transfer control register. Note that this bit is write-to-clear. Alternatively, determine whether the indirect read operation is complete by generating an interrupt.
10. Read the number of completed indirect read operations from the same register.

When the DMA peripheral controller is disabled, the following process must be followed:

1. Set the QSPI configuration register (QSPI_CR).
2. Set the start address of the Flash for the indirect transfer through the indirect read transfer start address register (QSPI_IRTSAR).
3. Set the number of bytes to be transferred in the indirect read transfer number register (QSPI_IRTNR).
4. Set the AHB trigger address for the indirect transfer in the indirect AHB address trigger register (QSPI_IATR).
5. Set the range of the AHB trigger address for the indirect transfer in the indirect trigger address range register (QSPI_ITARR).
6. If the fill level watermark interrupt feature is adopted, an interrupt will be generated when the fill level status exceeds the threshold set in the indirect read transfer watermark register (QSPI_IRTWR). Setting the fill level watermark helps to indicate to the software when to read the next portion of the indirect read transfer. Note that if the fill level watermark is set to a non-zero value, a fill level watermark interrupt will be generated once the last byte of the indirect transfer is fetched and stored in the SRAM, even if the fill level watermark is higher than the actual fill level status.
7. Set QSPI_IRTR[0] = 1 to trigger the indirect read access.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8. If the fill level watermark interrupt feature is adopted, wait for the fill level watermark interrupt to occur. Otherwise, poll the SRAM fill level status to determine when there is enough data in the SRAM to trigger the fetching of AHB data.
9. Read the desired amount of data from the SRAM. If more data needs to be fetched to complete the indirect read transfer, go back to step 8; otherwise, proceed to step 10.
10. Poll the completion status of the indirect read operation through QSPI_IRTR[5] in the indirect read transfer control register.
11. When the indirect read operation is complete, an indirect completion interrupt will be generated.

32.5.3.2 Indirect Write Controller

The purpose of the indirect write operation is to efficiently transfer a large amount of data from the processor or DMA to the Flash memory. Indirect transfers execute as few write cycles as possible within the Flash device, thereby maximizing the lifespan of the device. From a software perspective, the indirect write operation can be seen as the opposite of the indirect read process. It is controlled and triggered through the indirect write transfer control register (QSPI_IWTR), the indirect write transfer watermark register (QSPI_IWTWR), the indirect write transfer start address register (QSPI_IWTSAR), and the indirect write transfer number register (QSPI_IWTNR). The module waits for the transmission of write data via the external AHB host. Before communicating with the existing traditional SPI IP core, the write data is stored in its own SRAM. By default, the indirect write controller is disabled. Before enabling it, software must configure the amount of data and the start address. The indirect write transfer start address register (QSPI_IWTSAR) defines the start address, and the indirect write transfer number register (QSPI_IWTNR) defines the number of bytes. Up to two indirect operations can be programmed simultaneously. The second indirect operation can be triggered during the first operation. A

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

brief turnaround time is allowed between the completion of the previous indirect operation and the start of the next one. The indirect write sequence is similar to the indirect read sequence. In indirect operations, the total number of bytes written is not limited by the size of SRAM. The size of SRAM only limits the amount of data that can be received from the external AHB host. When the bus master is DMA, the amount of data requested by the controller through the DMA peripheral interface will not exceed the existing fill level of SRAM. However, this does not guarantee that DMA or other bus masters will not attempt to send more data than the SRAM can receive. When an SRAM overflow occurs, the controller backpressures the AHB while in the wait state. Note that the fill level status of SRAM can be read through the SRAM fill level status register (QSPI_SFLR) to avoid overflow.

The bus master provides write data and sends it to the QSPI via an AHB write operation. The received write access AHB address must be within the range from the AHB indirect trigger address (configured via the indirect AHB address trigger register, QSPI_IATR) to (AHB indirect trigger address + 2** (indirect AHB trigger address range) - 1). The default value of this range is 16, effectively handling burst operations of 16 accesses or fewer. Additionally, there is no strict requirement for stacking continuous address sequences. The actual AHB address must agree within the indirect range for the SRAM to be the source. Each valid AHB indirect write will cause the internal SRAM to pop, thereby decoupling the AHB address from the Flash address. Therefore, the AHB indirect trigger address has no relation to the Flash address at all. After triggering any valid indirect write, the SRAM is considered the source rather than the Flash memory array. The Flash address for the indirect write is obtained from the indirect write transfer start address register (QSPI_IWTSAR). Assuming that the SRAM status is not full when the QSPI controller receives the AHB access, data will be pushed into the SRAM with minimal delay. If the received AHB write access address is not within the above range, the access will not be completed by the indirect controller but by the direct access controller. If the received

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

AHB write access is within the above range but the SRAM is full, the AHB will be held in a wait state until some or all data is pushed from the SRAM to the Flash. If the accessed elements of the received AHB write burst span the AHB indirect trigger range, the accesses within the indirect trigger range will be processed by the indirect access controller, while the remaining part will be handled by the direct access controller. Note that this may indicate a software configuration error. The bus master is only allowed to send 32-bit AHB writes until the indirect transfer ends, which reduces the complexity of the SRAM control logic. When writing the last byte, the bus master can send a 32-bit, 16-bit (half-word), or a single byte to end the transfer. If the last transfer is less than 4 bytes, the master can still send a 32-bit data transfer, and the extra bytes will be discarded. When the number of bytes in the SRAM is equal to or greater than the capacity of a Flash page (default 256 bytes), or when the bytes in the SRAM are all the remaining bytes for the current indirect transfer being executed, the controller will initiate a write burst to the command generator. The indirect operation can be canceled at any time by setting QSPI_IWTR.WRDIS. Any bus master can initiate an indirect access. The DMA bus peripheral interface can share some of the software overhead and effectively manage data transfers. Another option is to directly access the SRAM fill level status through the APB registers to determine when to write data to the SRAM. When the DMA peripheral interface is disabled, a fill level watermark is provided that can be accessed via the indirect write transfer watermark register (QSPI_IWTWR). An interrupt will be generated when the SRAM fill level status falls below this watermark. Note that the fill level watermark register is a dual-purpose register. When the DMA peripheral interface is enabled, the hardware controls the rate at which DMA requests are sent. In this mode, since the QSPI will not initiate a write operation unless there is at least one Flash page in the SRAM, or the remaining bytes for the indirect transfer are in the SRAM, the data fill level watermark in DMA mode must be set to be greater than or equal to one Flash page. To understand the status of the indirect operation, two additional interrupt sources are provided. The first is an interrupt generated when an indirect operation is

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

completed. The second is an interrupt generated if an indirect write request is issued but not accepted because the QSPI controller has already cached two indirect operations. Setting QSPI_IWTR[0] initiates an indirect write operation, and QSPI_IWTR[5] can be used to check the status.

Indirect write transfer process:

When the DMA peripheral controller is enabled, the following process must be followed:

1. Set the start address of the Flash for the indirect transfer through the indirect write transfer start address register (QSPI_IWTSAR).
2. Set the number of bytes to be transferred in the indirect write transfer number register (QSPI_IWTNR).
3. Set the AHB trigger address for the indirect transfer in the indirect AHB address trigger register (QSPI_IATR).
4. Set the range of the AHB trigger address for the indirect transfer in the indirect trigger address range register (QSPI_ITARR).
5. Set the number of bytes for DMA single and burst transfers in the DMA peripheral configuration register (QSPI_DMACR).
6. Optionally: Set the indirect write transfer watermark register (QSPI_IWTWR) to control the rate of DMA request transmission.
7. Set QSPI_IWTR[0] = 1 to trigger the indirect write access.
8. The QSPI controller will use the DMA request interface to request DMA data transfer.
9. Poll the completion status of the indirect write operation through QSPI_IWTR[5] in the indirect write transfer control register. Note that this bit is write-to-clear. Read the number of completed indirect write operations in the same register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10. When the indirect write operation is complete, an indirect completion interrupt will be generated.

When the DMA peripheral controller is disabled, the following process must be followed:

1. Set the start address of the Flash for the indirect transfer through the indirect write transfer start address register (QSPI_IWTSAR).
2. Set the number of bytes to be transferred in the indirect write transfer number register (QSPI_IWTNR).
3. Set the AHB trigger address for the indirect transfer in the indirect AHB address trigger register (QSPI_IATR).
4. Set the range of the AHB trigger address for the indirect transfer in the indirect trigger address range register (QSPI_ITARR).
5. Functionally, the software can simply write all data to SRAM within a block. However, if the number of bytes written exceeds the size of the SRAM, it is highly likely that the SRAM will become full, causing the QSPI to spend a significant amount of time back-pressuring the system AHB bus. This time is based on the data rate of the Flash and the page write time of the device. To avoid sending all write data in a single block, software can use the fill level watermark interrupt to send one page of data at a time at the appropriate time. Alternatively, software can directly poll the SRAM fill level status register (QSPI_SFLR) to identify the empty state of the SRAM and then determine the most suitable time to send the next portion of data.
6. If the fill level watermark interrupt feature is adopted, an interrupt will be generated when the fill level falls below the watermark. The setting range for the fill level watermark is from 0 to page capacity. For example: if the page capacity is 256 bytes, a reasonable setting for the data depth threshold would be between 10 and 250. An interrupt will be triggered when

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

the fill level drops below the set value. Setting the fill level watermark helps to indicate to the software when to write the next page of data for the indirect write transfer to SRAM.

7. Set QSPI_IWTR[0] = 1 to trigger the indirect write access.
8. If the remaining byte count of the current indirect transfer is greater than one Flash page, write one Flash page of data to the SRAM; otherwise, send all remaining data to the SRAM.
9. If all data in the indirect transfer has been sent to the SRAM, proceed to step 11 and wait for the completion status. If there is more data to be transferred, then:
   - If the fill level watermark interrupt feature is adopted, wait for the interrupt to occur.
   - Use the SRAM data depth status to determine the appropriate time to send data.
10. Loop back to step 8.
11. Optionally: The completion status of the indirect write operation can be queried through the indirect write transfer control register (QSPI_IWTR).
12. When the indirect write operation is complete, an indirect completion interrupt will be generated.

32.5.3.3 Indirect Access Queue

Software allows the indirect write controller and indirect read controller to have two indirect transfers queued and waiting. A brief turnaround time is supported between the completion of the previous indirect operation and the start of the next one. An interrupt will be generated if the waiting queue exceeds two. In terms of software, the indirect access queue is implemented by triggering QSPI_IRTR[0] or QSPI_IWTR[0] twice in a short period. Before each trigger of QSPI_IRTR[0] or QSPI_IWTR[0], the indirect transfer start address registers (QSPI_IRTSAR/QSPI_IWTSAR) and the indirect transfer byte number registers (QSPI_IRTNR/QSPI_IWTNR) must be set. Since these registers will be periodically updated, the

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

hardware needs to maintain a sampled state of these registers during the cycle of the indirect transfer. The internal register block sends only one indirect start trigger to the key data path block at a time. There are two independent data path blocks in the indirect access controller that can receive and independently sample the indirect start trigger information. The first data path block exists at the AHB end of the SRAM. It acts as a read interface when performing an indirect read and as a write interface when performing an indirect write. The second data path block is at the Flash end of the SRAM. It acts as a write interface when performing an indirect read and as a read interface when performing an indirect write. These two blocks can handle indirect transfers at different times. For example: during an indirect read operation, once the last byte of the first transfer has been written into the SRAM, the data path block on the Flash side of the SRAM can begin processing the second queued transfer. Before starting the second transfer, the indirect transfer start address registers (QSPI_IRTSAR/QSPI_IWTSAR) and the indirect transfer byte number registers (QSPI_IRTNR/QSPI_IWTNR) must be resampled. Similarly, when all the Flash data of the first indirect transfer has been sent from the SRAM to the AHB, the data path block on the AHB will also resample the same registers.

32.5.3.4 Indirect Transfer: Continuous Write and Read

During the indirect write operation, the software allows triggering an indirect read operation. Similarly, during an indirect read operation, it is also possible to trigger an indirect write operation. The indirect write operation has a higher priority.

32.5.3.5 Accessing SRAM

Physically, the SRAM is a single-port module, whose depth is configurable. SRAM is divided into two areas, with the lower region reserved for indirect read use and the upper region for indirect write use only. The capacity of both parts can be configured through the SRAM partitioning configuration register (QSPI_SPR), allowing users to choose how many bits of the SRAM address

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

bus to allocate for indirect read operations. By default, this register sets half of the SRAM for use by the indirect read controller. To ensure that the AHB read data bus is not directly provided by the SRAM read data through combinational logic, an additional set of registers is included in the indirect read data path. The following examples illustrate the allocation of SRAM (and another set of registers) between indirect read and indirect write. In the following examples, the depth of SRAM is 8 bits, equivalent to 256 addresses.

- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0x00, 256 addresses are allocated to indirect write, and 1 address is allocated to indirect read.
- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0x01, 255 addresses are allocated to indirect write, and 2 address is allocated to indirect read.
- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0x02, 254 addresses are allocated to indirect write, and 3 address is allocated to indirect read.
- By analogy.
- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0xFD, 3 addresses are allocated to indirect write, and 254 address is allocated to indirect read.
- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0xFE, 2 addresses are allocated to indirect write, and 255 address is allocated to indirect read.
- If the SRAM partitioning configuration register (QSPI_SPR) is set to 0xFF, 1 address is allocated to indirect write, and 256 address is allocated to indirect read.

Note: Avoid setting the SRAM partitioning configuration register (QSPI_SPR) to 0xFF or 0x00, as only the lower 8 bits of the SRAM data depth status can be accessed by software. If the data depth for indirect read or indirect write reaches 256, it will appear as 0 when reading the data depth.

There are four SRAM sources, all of which arbitrate and multiplex on a single SRAM port. Up to

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

three sources can access the port at any given time.

- Indirect write, write source, located on the AHB side of the SRAM.
- Indirect write, read source, located on the Flash side of the SRAM.
- Indirect read, write source, located on the Flash side of the SRAM.
- Indirect read, read source, located on the AHB side of the SRAM.

The arbitration scheme for priority is as follows:

Table 32-2: SRAM Access Priority

Access Method		SRAM Access Priority
Indirect write	Writing to SRAM (from system AHB)	3rd priority (excluding AHB read requests)
	Reading from SRAM (from QSPI controller)	2nd priority
Indirect read	Writing to SRAM (from QSPI controller)	1st priority
	Reading from SRAM (from system AHB)	3rd priority (excluding AHB write requests)

During the indirect read operation (Flash in SRAM), except for the write port, the logic driving the four sources cannot be considered to be completed in a single cycle. To avoid data loss, the write to SRAM must be allowed to complete immediately during the indirect read process. Therefore, this port has the highest priority.

32.5.4 DMA Peripheral Controller

The peripheral interface is used to trigger external DMA for short-latency AHB data burst access. The DMA peripheral interface is only used in indirect operation mode, where data is cached in the built-in SRAM to respond to AHB requests more quickly and allow the core to perform underlying Flash transfers over a longer period. It supports two DMA requests, one for the indirect read controller and another for the indirect write controller. Regarding the indirect read

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

controller, the QSPI controller only sends a DMA request after the data has been retrieved from Flash and written into its own SRAM. Regarding the indirect write controller, the QSPI controller immediately sends a DMA request upon triggering the transfer and continues to send it until the entire indirect write transfer is completed. The indirect transfer watermark registers (QSPI_IRTWR/QSPI_IWTWR) can alter the rate at which requests are sent.

32.5.4.1 Operation Sequence

When an indirect operation is triggered, the DMA peripheral controller is aware of all the transfer data (in bytes) from Flash memory. The controller divides these data into DMA burst requests and single requests: All bytes divided by the number of bytes set in the burst request, with the remainder divided by the number of bytes in the single request. Software must ensure that there are no remainders after these divisions. For example, if the total number of bytes to be read from Flash is 512, the fixed capacity of SRAM is 256 bytes, and the software-configured burst transfer byte count is 256, then after the current 256 bytes are buffered, the SPI controller will trigger a DMA burst request. A second burst request will be triggered when another 256 bytes are buffered in SRAM. Since the capacity of SRAM itself is 256 bytes, before sending the next request, DMA will retrieve all the contents from SRAM. The SRAM fill level status can be designed to be visible to the DMA peripheral controller. Regarding indirect read, requests are sent to the external DMA based on the following state machine.

1. Wait for the START trigger.
2. If the next request to be sent is a BURST request, wait until both of the following conditions are TRUE:
   A. The SRAM fill level status is greater than or equal to the number of bytes set in bits [11:8] of the QSPI_DMACR register.
   B. The SRAM fill level status is greater than or equal to the value in the indirect transfer

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

watermark registers (QSPI_IRTWR/QSPI_IWTWR).

When both conditions are TRUE, perform the following:

A. Send the BURST.

B. Calculate the expected fill level = SRAM fill level - num_bytes_requested_to_DMA.

C. If the next request to be sent is a BURST request, go to step 4.

D. If the next request to be sent is a SINGLE request, go to step 5.

E. For any other case, go to step 1.

3. If the next request to be sent is a SINGLE request, wait until the following condition is TRUE:

The SRAM fill level status is greater than or equal to the number of bytes set in bits [3:0] of the QSPI_DMACR register.

When the condition is TRUE, perform the following:

A. Send the SINGLE.

B. Calculate the expected fill level = SRAM fill level - num_bytes_requested_to_DMA.

C. If the next request to be sent is a SINGLE request, go to step 5.

D. For any other case, go to step 1.

E. If the next request to be sent is a BURST request, and if (SRAM fill level – num_bytes_requested_to_DMA) is less than the number of bytes set in bits [11:8] of the QSPI_DMACR register, go to step 2. Otherwise, perform the following:

F. Send the BURST.

G. Calculate the expected fill level = SRAM fill level - num_bytes_requested_to_DMA.

H. If the next request to be sent is a BURST request, go to step 4.

I. If the next request to be sent is a SINGLE request, go to step 5.

J. For any other case, go to step 1.

4. If the next request to be sent is a SINGLE request, and if (SRAM fill level – num_bytes_requested_to_DMA) is less than the number of bytes set in bits [3:0] of the QSPI_DMACR register, go to step 3. Otherwise, perform the following:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

A. Send the SINGLE.

B. Calculate the expected fill level = SRAM fill level - num_bytes_requested_to_DMA.

C. If the next request to be sent is a SINGLE request, go to step 5.

D. For any other case, go to step 1.

Regarding indirect write, requests are sent to the external DMA based on the following state machine.

1. Wait for the START trigger.
2. If the next request to be sent is a BURST request, wait until both of the following conditions are TRUE:

   A. The remaining space in SRAM is greater than or equal to the number of bytes set in bits [11:8] of the QSPI_DMACR register.

   B. The SRAM fill level status is smaller than or equal to the value in the indirect transfer watermark registers (QSPI_IRTWR/QSPI_IWTWR).

   When the conditions are TRUE, perform the following:

   A. Send the BURST.

   B. Calculate the expected fill level = SRAM fill level + num_bytes_requested_to_DMA.

   C. If the next request to be sent is a BURST request, go to step 4.

   D. If the next request to be sent is a SINGLE request, go to step 5.

   E. For any other case, go to step 1.
3. If the next request to be sent is a SINGLE request, wait until the following condition is TRUE:

   The remaining space in RAM is greater than or equal to the number of bytes set in bits [3:0] of the QSPI_DMACR register.

   When the condition is TRUE, perform the following:

   A. Send the SINGLE.

   B. Calculate the expected fill level = SRAM fill level + num_bytes_requested_to_DMA.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

C. If the next request to be sent is a SINGLE request, go to step 5.

D. For any other case, go to step 1.

4. If the next request to be sent is a BURST request, and if (SRAM remaining space – num_bytes_requested_to_DMA) is less than the number of bytes set in bits [11:8] of the QSPI_DMACR register, go to step 2. Otherwise, perform the following:

   A. Send the BURST.

   B. Calculate the expected fill level = SRAM fill level + num_bytes_requested_to_DMA.

   C. If the next request to be sent is a BURST request, go to step 4.

   D. If the next request to be sent is a SINGLE request, go to step 5.

   E. For any other case, go to step 1.

5. If the next request to be sent is a SINGLE request, and if (SRAM remaining space – num_bytes_requested_to_DMA) is less than the number of bytes set in bits [3:0] of the QSPI_DMACR register, go to step 3. Otherwise, perform the following:

   A. Send the SINGLE.

   B. Calculate the expected fill level = SRAM fill level + num_bytes_requested_to_DMA.

   C. If the next request to be sent is a SINGLE request, go to step 5.

   D. For any other case, go to step 1.

For indirect read operations, the indirect read transfer watermark register (QSPI_IRTWR) can define the minimum fill level for the controller to send the first DMA request. The larger the value set, the more data will be cached in SRAM before DMA fetches it. For indirect write operations, the indirect write transfer watermark register (QSPI_IWTWR) can define the maximum fill level for the controller to send the first DMA burst/single request. The indirect transfer watermark registers (QSPI_IRTWR/QSPI_IWTWR) can enable the system to selectively focus on AHB transfers in short periods. By default, the indirect transfer watermark registers (QSPI_IRTWR/QSPI_IWTWR) are reset to 0, meaning that the peripheral controller can send DMA requests as quickly as possible. In the case of indirect read, this means that there is sufficient

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

data in the SRAM for burst or single requests (if the remaining bytes are less than the set value for burst requests). Note that if SRAM is empty, the DMA peripheral cannot send an indirect read DMA request; if SRAM is full, the DMA peripheral cannot send an indirect write request. The DMA peripheral interface can be disabled by software. When the interface is disabled, no channel requests will be sent. Note that if an AHB master other than the DMA will perform indirect data transfers, the DMA peripheral interface must be disabled.

32.5.5 Software Triggered Instruction Generator (STIG)

The direct and indirect access controllers are used for data transfer. For erasing and accessing volatile and non-volatile configuration registers, traditional SPI status registers, and other status/protection registers, an independent software controller is required. The software triggered instruction generator (STIG) is controlled by the Flash command control register (QSPI_FCR) through setting the command to be sent to the Flash device. This is a general-purpose register that can be used to execute any instruction supported by the extended SPI protocol of the Flash device. Instructions that do not conform to the Flash device specifications may result in undefined controller behavior. When reading, up to 8 data bytes can be placed in the Flash command read data register (QSPI_FCRLR/QSPI_FCRHR) once the command has been responded to (QSPI_FCR[1] flips from 1 to 0). When writing, the write data shall be placed in the Flash command write data register (QSPI_FCWLR/QSPI_FCWHR). The code that implements the above functions is stored in the QSPI APB register block.

32.5.5.1 Responding to STIG Request

When responding to the STIG request, the QSPI Flash controller will query the Flash command control register (QSPI_FCR) to determine the manner in which to send how many bytes to the Flash device. The instruction to be sent is indicated by QSPI_FCR[31:24], and it is always pushed

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

onto the stack first. The address is sent immediately after the instruction, with the address size also configured in this register. The address itself is stored in the Flash command address register (QSPI_FCAR). Dummy bytes are subsequently sent (the number of bytes is still configured in the Flash command control register (QSPI_FCR)). The number of bytes for write/read data is also configured in the Flash command control register (QSPI_FCR). When writing, up to 8 bytes can be sent at most (stored in the Flash command write data register (QSPI_FCWLR/QSPI_FCWHR)). When reading, after the read data has been obtained from the Flash device, the QSPI Flash controller will store the read data in the Flash command read data register (QSPI_FCRLR/QSPI_FCRHR). When the QSPI Flash controller begins to respond to the STIG request, QSPI_FCR1 is set to 1, indicating that the command is being executed. When the QSPI Flash controller is in the automatic polling state, responding to the STIG request will be a little different. After a programming operation, most devices are inaccessible until the write operation is complete. Some of these devices may also stop programming pages. QSPI_PFSR[8] indicates the effective automatic polling state. Upon receiving the STIG request, the QSPI Flash controller will immediately send the appropriate OPCODE to the memory. During the response to STIG (in automatic polling), the command execution status bits are maintained, while other bits (such as the ADDRESS or DUMMY bit) are all disabled. There is also a configurable option: to add a delay between each repeated polling operation (the delay is defined by QSPI_WCR[31:24]). This feature is intended to release SPI bandwidth.

32.5.6 Arbitration Between Direct/Indirect Access Controllers and STIG

When multiple registers are active simultaneously, a simple fixed-priority arbitration scheme can be used to arbitrate each interface for accessing the external Flash. The fixed priorities are defined as follows, with 1 being the highest priority:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

1. Indirect write
2. Direct write
3. STIG
4. Direct read
5. Indirect read

While waiting for a response, each controller will be back-pressured.

32.5.7 SPI Command Conversion

Requests sent by the direct access controller, indirect access controller, or STIG are converted into byte transmission sequences and sent downstream. The following example shows a single-byte non-sequential read:

INSTRUCTION OPCODE ADDRESS Mode Byte Dummy Bytes 1 byte of don't care

When performing sequential access, each read operation will push an additional byte onto the Flash device at the end of the above sequence. This is done to ensure that there are no gaps between each transmitted byte. The actual sequence sent to the Flash device depends on the required transmission, whether it is non-sequential or sequential, whether the device has been configured in XIP mode, and the status of the main device instruction type register (device read instruction register (QSPI_DRIR)/device write instruction register (QSPI_DWIR)). When writing, the write enable latch (WEL) in the Flash device must be high before the write sequence is sent. Before triggering a write command through the direct or indirect access controller (DAC/INDAC), the QSPI Flash controller will automatically send a write enable latch command. To improve flexibility and performance, users can disable this function by setting QSPI_DWIR[8]. The OPCODE for WREN is 0x06, which is shared among devices. When no more write requests are received from the direct or indirect access controller and all requests have been issued, the Flash device will automatically start a page programming write cycle. Any requests received at

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

this time will be kept in a pending state until the write cycle is completed. The QSPI Flash controller will automatically poll the traditional SPI status register of the Flash device to determine whether the write cycle is complete: it sends the RDSR OPCODE to the Flash device and waits until the device itself indicates that the write cycle is complete (until the write in progress bit[0] is cleared and the write enable latch bit is also cleared). The WREN and RDSR device commands are the only two commands sent by the controller at the underlying level. Any other specific commands shall be sent to the device and handled individually by sending Flash commands via STIG.

32.5.8 Flash Instruction Type Selection

To send the correct read and write OPCODEs, the software shall initialize the QSPI device read instruction register (QSPI_DRIR) and the QSPI device write instruction register (QSPI_DWIR). These two registers allow the configuration of the instruction OPCODE, instruction type, and edge mode (DRR or SDR), and the selection of single, dual, or quad pins for address and data transmission. To ensure that the controller can operate in the reset state, the registers will be reset to OPCODEs compatible with SIO devices. The QSPI_DRIR includes an instruction type field, while the QSPI_DWIR does not have an instruction type field. If the software sets the instruction type to a non-zero value, the address transfer type and data transfer type bits in the QSPI_DRIR and the QSPI_DWIR are independent. The software supports uncommon Flash instructions that can send OPCODE, addresses, and data in 2 or 4 channels. Note that for devices that support sending instructions with OPCODE in 2 or 4 channels, the names of these instructions may not be consistent with those in the Flash data manual. One of the devices that supports these instructions is the Numonyx (Micron) N25Q128. Other read instructions are called DCFR and QCFR, while the write instructions are called DCPP and QCPP. The following table, using the N25Q128 as a reference, shows how the software should configure the QSPI.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- **Read**

OPCODE	Number of Channels for OPCODE Transmission	Number of Channels for ADDRESS/DUMMY/MODE Transmission	Number of DATA Bytes Transmitted	QSPI Instruction Type (Device Read Instruction Register)	QSPI Address Xfer Type (Device Read Instruction Register)	QSPI Data Xfer Type (Device Read Instruction Register)
READ	1	1	1	0	0	0
FAST_READ	1	1	1	0	0	0
DOFR (DualO/PFastRead)	1	1	2	0	0	1
DIOFR (DualO/PFastRead)	1	2	2	0	1	1
QOFR (QuadO/PFastRead)	1	1	4	0	0	2
QIOFR (QuadO/PFastRead)	1	4	4	0	2	2
DCFR (DualCommandFastRead)	2	2	2	1	Don't Care	Don't Care
QCFR (QuadCommandFastRead)	4	4	4	2	Don't Care	Don't Care

- **Write**

OPCODE	Number of Channels for OPCODE Transmission	Number of Channels for ADDRESS/DUMMY/MODE Transmission	Number of DATA Bytes Transmitted	QSPI Instruction Type (Device Write Instruction Register)	QSPI Address Xfer Type (Device Write Instruction Register)	QSPI Data Xfer Type (Device Write Instruction Register)
PP	1	1	1	0	0	0
DIFP (Dual Input Fast Program)	1	1	2	0	0	1
DIEFP (Dual Input Extended Fast Program)	1	2	2	0	1	1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

OPCODE	Number of Channels for OPCODE Transmission	Number of Channels for ADDRESS/DUMMY/MODE Transmission	Number of DATA Bytes Transmitted	QSPI Instruction Type (Device Write Instruction Register)	QSPI Address Xfer Type (Device Write Instruction Register)	QSPI Data Xfer Type (Device Write Instruction Register)
QIFP (Quad Input Fast Program)	1	1	4	0	0	2
QIEFP (Quad Input Extended Fast Program)	1	4	4	0	2	2
DCPP (Dual Command Fast Program)	2	2	2	1	Don't Care	Don't Care
QCPP (Quad Command Fast Program)	4	4	4	2	Don't Care	Don't Care

Some devices (for example, Numonyx (Micro) N25Q512A) can handle read operations in double data rate mode (DDR, also known as double transfer rate mode (DTR)). During proprietary command type operations, these devices can transmit data on both the rising and falling edges. This allows the controller to maintain throughput at a frequency lower than twice the spi_clk frequency. QSPI_DRIR[10] can inform the QSPI Flash controller that the OPCODE written to the read OPCODE bit field can handle DDR command types. The read data capture register (QSPI_RDCR) can shift the transmitted data in DDR mode. By default, shifting the data by one clock cycle ensures that the hold time is greater than 0 in DDR transmission. For high ref_clk frequencies, shifting by one clock may not be sufficient. The configuration of DDR commands is compatible with the SDR commands listed in the read table. The MT25 series in Micron devices also includes the DTR protocol, which can handle all commands in DTR mode. DTR read commands can detect the DTR mode based on specific OPCODEs. Therefore, the OPCODE must be sent as STR. When the DTR protocol is enabled, the device does not require an OPCODE to detect the edge mode, as it can be identified from the volatile and non-volatile bits in the configuration register (QSPI_CR).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.5.9 APB Interface and Register Module

The APB interface uses the Flash command control register (QSPI_FCR) to perform Flash access controlled by software. The APB interface provides a single register block that contains a configurable group of registers. This group of registers is timed by the APB clock.

32.5.10 SRAM Module

The indirect operation mode requires a memory module. The memory pins are connected at the top level, allowing the integrator to choose to use SRAM memory, a register array, or a group of FLOPs. The memory module has only one port. The depth of the memory module is selectable and determined by the specific application.

32.6 Register Description

QSPI register base address: 0x40E0_2000

The registers are listed below.

Table 32-3: List of QSPI Flash Controller Registers

Offset Address	Register Name	Register Description
0x00	QSPI_CR	QSPI configuration register
0x04	QSPI_DRIR	QSPI device read instruction register
0x08	QSPI_DWIR	QSPI device write instruction register
0x0C	QSPI_DDLR	QSPI device delay register
0x10	QSPI_RDCR	QSPI read data capture register
0x14	QSPI_DSCR	QSPI device size configuration register
0x18	QSPI_SPR	QSPI SRAM partitioning configuration register
0x1C	QSPI_IATR	QSPI indirect AHB address trigger register
0x20	QSPI_DMACR	QSPI DMA peripheral configuration register
0x24	QSPI_RAR	QSPI address remapping register
0x28	QSPI_MBR	QSPI mode bit register
0x2C	QSPI_SFLR	QSPI SRAM fill level register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Register Name	Register Description
0x30	QSPI_TXHR	QSPI transmit threshold register
0x34	QSPI_RXHR	QSPI receive threshold register
0x38	QSPI_WCR	QSPI write completion control register
0x3C	QSPI_PER	QSPI polling end register
0x40	QSPI_IFR	QSPI interrupt flag register
0x44	QSPI_IMR	QSPI interrupt mask register
0x48–0x4C	RSV	Reserved
0x50	QSPI_WPLR	QSPI write protection low register
0x54	QSPI_WPHR	QSPI write protection high register
0x58	QSPI_WPCR	QSPI write protection configuration register
0x5C	RSV	Reserved
0x60	QSPI_IRTR	QSPI indirect read transfer control register
0x64	QSPI_IRTWR	QSPI indirect read transfer fill level watermark register
0x68	QSPI_IRTSAR	QSPI indirect read transfer start address register
0x6C	QSPI_IRTNR	QSPI indirect read transfer byte number register
0x70	QSPI_IWTR	QSPI indirect write transfer control register
0x74	QSPI_IWTWR	QSPI indirect write transfer fill level watermark register
0x78	QSPI_IWTSAR	QSPI indirect write transfer start address register
0x7C	QSPI_IWTNR	QSPI indirect write transfer byte count register
0x80	QSPI_ITARR	QSPI indirect trigger address range register
0x84–0x8C	RSV	Reserved
0x90	QSPI_FCR	QSPI Flash command control register
0x94	QSPI_FCAR	QSPI Flash command address register
0x98–0x9C	RSV	Reserved
0xA0	QSPI_FCRLR	QSPI Flash command read data low register
0xA4	QSPI_FCRHR	QSPI Flash command read data high register
0xA8	QSPI_FCWLR	QSPI Flash command write data low register
0xAC	QSPI_FCWHR	QSPI Flash command write data high register
0xB0	QSPI_PFSR	QSPI polling Flash status register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.1 QSPI Configuration Register (QSPI_CR)

Offset address: 0x00

Reset value: 0x8078 0081

Bit	Name	Attribute	Reset Value	Description
31	IDLES	R	0x1	**Serial interface and QSPIPIPELINE are idle:** This bit is a re-timed signal, so there will be some delay when generating this status bit signal. It is recommended to wait for at least 4 ref_clk cycles between enabling the IDLE bit and switching the configuration domain. This delay ensures underlying synchronization, that the FIFO is empty, and that the controller can introduce any operational mode.
30:25	RSV	-	-	Reserved
24	DTRM	R/W	0x0	**DTR protocol enable:** Set this bit if the device uses the DTR protocol. Note that the DTR protocol provides all commands in DTR mode. Some DTR read commands can be processed in the STR protocol (in DTR mode). Set this bit to 0 to execute these commands. DTR commands in the STR protocol are controlled by QSPI_DRIR.DDRM.
23	AD_EN	R/W	0x0	**AHB decoder enable (direct access mode only):** When set to 1, the PSL bit is a don't care bit. The effective slave depends on the actual AHB address. The configuration of the device is calculated based on QSPI_DSCR.CS_SIZE.
22:19	BAUD	R/W	0xF	**Master mode baud rate division:** SPI baud rate = (master reference clock) / BD Where BD can be configured as follows: 0000 = / 2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0001 = / 4 0010 = / 6 0011 = / 8 0100 = / 10 0101 = / 12 0110 = / 14 0111 = / 16 1000 = / 18 … 1111 = / 32 Configure this register properly before enabling the QSPI controller. Note: The QSPI communication clock is derived from clk_sys_pre.
18	XIPIM	R/W	0x0	**Enter XIP mode immediately:** 0: If XIP is enabled, the controller will exit XIP mode after the next read instruction is completed. 1: Enter XIP mode immediately. Use this register when the external device wakes up from XIP mode. The controller assumes that the next read instruction will be passed to the device as an XIP instruction, and therefore will not request the transmission of the read operation code. Note: To exit XIP mode, this bit must be set to 0. It only takes effect in the connected device after the next read instruction is executed. Therefore, before exiting XIP mode, software must ensure that at least one read instruction is executed after resetting this bit. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
17	XIPNX	R/W	0x0	**Enter XIP mode with the next read instruction:**

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: If XIP is enabled, the controller will exit XIP mode after the next read instruction is completed. 1: If XIP is disabled, setting this bit to 1 can notify the controller that the device is ready to enter XIP mode with the next read instruction. Then, the controller will send the appropriate command sequence, including the mode bit to allow the device to enter XIP mode. Use this register after the controller has ensured that the connected Flash devices are ready to enter XIP mode. Note: To exit XIP mode, this bit must be set to 0. It only takes effect in the connected device after the next read instruction is executed. Therefore, before exiting XIP mode, software must ensure that at least one read instruction is executed after resetting this bit. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
16	AR_EN	R/W	0x0	**AHB address remapping enable (direct access mode only):** When this bit is set to 1, the received AHB address will be updated, and (address + N) will be sent to the Flash device, where the value of N is stored in the address remapping register (QSPI_RAR). This bit is synchronized by hardware. It can be set or cleared while the controller is running.
15	DMA_EN	R/W	0x0	**DMA peripheral interface enable:** 0: Disabled 1: DMA handshake logic enabled: When enabled, QSPI will trigger DMA transfer requests through the DMA peripheral interface. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
14	SWPP	R/W	0x0	**Write protection pin configuration:**

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				Set this bit to drive the write protection pin of the Flash device, which will be resynchronized with the generated memory clock. Note that the WP pin is only effective in SINGLE or DUAL transfer modes. In QUAD transfer mode, the WP pin is used for data transmission, so the setting of this register bit will be ignored. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
13:9	RSV	-	-	Reserved
8	LIM_EN	R/W	0x0	**Traditional IP mode enable:** 0: Use direct access controller/indirect access controller or STIG interface for data transfer. 1: Enable traditional mode. In traditional mode, any AHB interface write operation is serially sent to the Flash device. Any valid AHB read will pop data from the internal RX-FIFO and retrieve the data forwarded by the external Flash device via the SPI line, with HSIZE input control specifying the transfer of 4, 2, or 1 byte. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
7	DAC_EN	R/W	0x1	**Direct access controller enable:** 0: Disable the direct access controller once the current data word (FF_W) transfer is completed. 1: Enable the direct access controller. When both the direct access controller and indirect access controller are disabled, an error response will be generated for all completed AHB requests. This bit is synchronized by hardware. It can be set or cleared while the controller is running.
6:3	RSV	-	-	Reserved
2	CPHA	R/W	0x0	**Clock phase:**

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				The clock phase maps to the standard SPI clock phase transmission format. Note: Keep this bit low when operating in DDR mode or adhering to DDR protocol.
1	CPOL	R/W	0x0	**Clock polarity outside SPI words:** Map to the standard SPI clock polarity transmission format. Note: Keep this bit low when operating in DDR mode or adhering to DDR protocol.
0	EN	R/W	0x1	**QSPI enable:** 0: Disable QSPI once the current data word (FF_W) transmission is complete. 1: Enable QSPI. When spi_enable=0, all output enables are invalid, and all pins are set to input mode. This bit is synchronized by hardware.

32.6.2 QSPI Device Read Instruction Register (QSPI_DRIR)

Offset address: 0x04

Reset value: 0x0000 0003

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	Reserved
28:24	DCYC	R/W	0x0	Number of dummy clock cycles required for reading instructions
23:21	RSV	-	-	Reserved
20	MB_EN	R/W	0x0	Mode bit enable: Setting this bit to 1 ensures that the mode bits defined in the mode bit register (QSPI_MBR) are sent following the address byte.
19:18	RSV	-	-	Reserved
17:16	DMODE	R/W	0x0	Data transfer type in standard SPI mode (as defined by the IMODE bit):

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: SIO mode – Data is sent to the device only on DQ0, and data is received from the device only on DQ1. 1: Used for DUAL input/output instructions. During data transfer, both DQ0 and DQ1 are used for input and output. 2: Used for QUAD input/output instructions. During data transfer, DQ0, DQ1, DQ2, and DQ3 are all used for input and output.
15:14	RSV	-	-	Reserved
13:12	AD_MODE	R/W	0x0	Address transfer type in standard SPI mode (as defined by the IMODE bit): 0: Address can only be sent to the device on DQ0. 1: Address can only be sent to the device on DQ0 and DQ1. 2: Address can be sent to the device on DQ0, DQ1, DQ2, and DQ3.
11	RSV	-	-	Reserved
10	DDRM	R/W	0x0	DDR bit enable: This bit is set to 1 when QSPI_WCR.OPCODE corresponds to a DDR command.
9:8	IMODE	R/W	0x0	Instruction type: 0: Standard SPI mode (instructions are always sent to the device only on DQ0) 1: DIO-SPI mode (instructions, addresses, and data are always sent on DQ0 and DQ1) 2: QIO-SPI mode (instructions, addresses, and data are always sent on DQ0, DQ1, DQ2, and DQ3) Note: This bit is not only related to read transfers. It is a global setting that will

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				affect DAC and INDAC read, write, and any STIG transfers.
7:0	RINST	R/W	0x3	Read operation code in non-XIP mode.

32.6.3 QSPI Device Write Instruction Register (QSPI_DWIR)

Offset address: 0x08

Reset value: 0x0000 0002

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	Reserved
28:24	DCYC	R/W	0x0	Number of dummy clock cycles required for writing instructions
23:18	RSV	-	-	Reserved
17:16	DMODE	R/W	0x0	Data transfer type in standard SPI mode: 0: SIO mode – Data is sent to the device only on DQ0, and data is received from the device only on DQ1. 1: Used for DUAL input/output instructions. During data transfer, both DQ0 and DQ1 are used for input and output. 2: Used for QUAD input/output instructions. During data transfer, DQ0, DQ1, DQ2, and DQ3 are all used for input and output.
15:14	RSV	-	-	Reserved
13:12	AD_MODE	R/W	0x0	Address transfer type in standard SPI mode: 0: Address can only be sent to the device on DQ0. 1: Address can only be sent to the device on DQ0 and DQ1. 2: Address can be sent to the device on DQ0, DQ1, DQ2, and DQ3.
11:9	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
8	WELD	R/W	0x0	WEL disabled: This bit allows disabling the automatic sending of the WEL command before a write operation.
7:0	WINST	R/W	0x2	Write operation code

32.6.4 QSPI Device Delay Register (QSPI_DDLR)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	CSDA	R/W	0x0	Chip select deactivation: The time between two transfers when the master mode chip select output is set to inactive (delay on the master reference clock ref_clk): The minimum delay for setting the chip select to inactive is: 1sclk_out + 1ref_clk To ensure that the chip select does not become active again within one sclk_out cycle, if CSDA = X, the inactive time for the chip select is: 1sclk_out+1ref_clk+Xref_clks
23:16	RSV	-	-	Reserved
15:8	CSEOT	R/W	0x0	Chip select transfer end: The delay between the last bit of the current transfer and the setting of the chip select (n_ss_out) to inactive (delay on the master reference clock ref_clk): By default, when CSEOT = 0, the chip select will be set to inactive on the last falling edge of sclk_out at the end of the current transfer. If CSEOT = X, the chip select will become inactive after X ref_clks following the last falling edge of sclk_out.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	CSSOT	R/W	0x0	Chip select transfer start: The delay between setting n_ss_out low and transmitting the first bit: By default, when CSSOT = 0, the chip select will be set to active half a SCLK cycle before the first rising edge of sclk_out. If CSSOT = X, the chip select will be set to active half a sclk_out cycle before the first rising edge of sclk_out + Xref_clks.

32.6.5 QSPI Read Data Capture Register (QSPI_RDCR)

Offset address: 0x10

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	Reserved
19:16	DLYT	R/W	0x0	Data transfer delay (The delay time is specified in terms of ref_clk cycles.) (This is intended to enhance the hold time during the transfer process in DDR mode.) This bit is only valid when a DDR read command is executed; otherwise, it is ignored. delay_value=1xref_clk+<field_value>*ref_clk
15:6	RSV	-	-	Reserved
5	SMES	R/W	0x0	Sample edge selection (Flash memory data output): 0: Sample the Flash memory data output on the falling edge of ref_clk 1: Sample the Flash memory data output on the rising edge of ref_clk
4:1	DLYR	R/W	0x0	Read data capture delay (the delay time is specified in terms of ref_clk cycles)
0	RSV	-	-	Reserved; write 1 by software.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.6 QSPI Device Size Configuration Register (QSPI_DSCR)

Offset address: 0x14

Reset value: 0x0010 1002

Bit	Name	Attribute	Reset Value	Description
31:23	RSV	-	-	Reserved
22:21	CS_SIZE	R/W	0x0	Size of Flash device connected to CS pin: This bit is valid when the AHB decoder is enabled. 00: 512MB 01: 1GB 10: 2GB 11: 4GB Note: This bit is useless if the AHB decoder is not supported.
20:16	BK_SIZE	R/W	0x10	Number of bytes per block: The controller requires this bit to implement write protection logic. The number of bytes per block must be a power of 2: 0: 1 byte 1: 2 bytes 2: 4 bytes 3: 8 bytes 16: 65535 bytes
15:4	PA_SIZE	R/W	0x100	Number of bytes per device page: The controller requires this bit to implement Flash write operations across pages.
3:0	AD_SIZE	R/W	0x2	Number of address bytes: 0: 1 byte …

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.7 QSPI SRAM Partitioning Configuration Register (QSPI_SPR)

Offset address: 0x18

Reset value: 0x0000 0080

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	SRAMPS	R/W	0x80	Define the indirect read area size in SRAM: By default, half of the size of SRAM is reserved for indirect read operations, while the other half is allocated for indirect write operations. The size of this register can change with the depth of SRAM. Number of addresses allocated for indirect read = SRAMPS + 1 Number of addresses allocated for indirect write = (2**8) - SRAMPS Note: Do not set SRAMPS to 0x00 or (2**8) - 1.

32.6.8 QSPI Indirect AHB Address Trigger Register (QSPI_IATR)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	IND_TAD	R/W	0x0	Indirect trigger address: This address serves as the base address. When the received AHB read access address matches an address within (the range from the trigger address to the trigger address + <configuration range>), the AHB request is fulfilled by retrieving data from the indirect controller SRAM.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.9 QSPI DMA Peripheral Configuration Register (QSPI_DMACR)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:8	BNUMB	R/W	0x0	Byte count for burst type in DMA peripheral requests: When set to 0, it represents one byte. This bit must be set before starting indirect read/write operations. The actual byte count is 2** (register setting value). The conditions for burst transfer are: 1) The available storage space in SRAM (in words) must be greater than or equal to the setting value of 2**[11:8] (in bytes). 2) The total amount of data for indirect access (indirect read transfer control register (QSPI_IRTR) / indirect write transfer control register (QSPI_IWTR)) must be greater than or equal to the setting value of 2**[11:8]. If the above conditions are not met, the system will automatically switch to single transfer mode. For example, if the SRAM size is 1KB and the SRAM partitioning configuration register (QSPI_SPR) is set to 0x80, when the first data indirect write transfer is initiated, the available storage space in SRAM is 0x80 words. If the value of [11:8] is set to be greater than or equal to 8 at this time, the controller will automatically adopt the single transfer mode.
7:4	RSV	-	-	Reserved
3:0	SNUMB	R/W	0x0	Byte count for single type in DMA peripheral requests:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				When set to 0, it represents one byte. This bit must be set before starting indirect read/write operations. The actual byte count is 2** (register setting value), where 2** (register setting value) indicates the amount of data transferred in a single transfer. For example, to transfer one word in a single transfer, this control bit shall be set to 2. If it is set to 0, the data transferred in a single transfer will be one byte. Note that in application, the size shall be set to an integer multiple of the word size.

32.6.10 QSPI Address Remapping Register (QSPI_RAR)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	READDR	R/W	0x0	This register remaps the received AHB address to a different address.

32.6.11 QSPI Mode Bit Register (QSPI_MBR)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	MODEB	R/W	0x0	Mode bit: After the address byte, these 8 mode bits are sent to the external device.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.12 QSPI SRAM Fill Level Register (QSPI_SFLR)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	IND_WSFL	R	0x0	SRAM fill level (indirect write area), in units of words (4 bytes)
15:0	IND_RSFL	R	0x0	SRAM fill level (indirect read area), in units of words (4 bytes)

32.6.13 QSPI Transmit Threshold Register (QSPI_TXHR)

Offset address: 0x30

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4:0	TXTH	R/W	0x1	Small-capacity TX FIFO not full interrupt threshold setting value: This bit is only valid when accessing the Flash device in traditional mode. It can be ignored in other cases.

32.6.14 QSPI Receive Threshold Register (QSPI_RXHR)

Offset address: 0x34

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:5	RSV	-	-	Reserved
4:0	RXTH	R/W	0x1	Small-capacity RX FIFO not empty interrupt threshold setting value: This bit is only valid when accessing the Flash device in traditional mode. It can be ignored in other cases.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.15 QSPI Write Completion Control Register (QSPI_WCR)

Offset address: 0x38

Reset value: 0x0001 0005

Bit	Name	Attribute	Reset Value	Description
31:24	PREPD	R/W	0x0	Polling repeat delay: The additional delay time for which the chip select remains inactive during the automatic polling phase.
23:16	PCNT	R/W	0x1	Polling count: Define the number of polling cycles. When the controller throughput is high, it is necessary to add extra cycles.
15	POLL_EXP_EN	R/W	0x0	Enable polling end interrupt: 0: Disabled 1: An interrupt is generated after the number of cycles set in QSPI_PER ends.
14	PDIS	R/W	0x0	Polling disable: Disable the automatic polling function.
13	PPLT	R/W	0x0	Polling polarity: 0: Write transfer is complete if the polling bit is 0. 1: Write transfer is complete if the polling bit is 1.
12:11	RSV	-	-	Reserved
10:8	PBIND	R/W	0x0	Polling bit retrieval: When set to 010, it indicates polling bit 2 of the QSPI_PFSR register. When set to 010, it indicates polling bit 2 of the QSPI_PFSR register.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	OPCODE	R/W	0x5	Opcode: When polling the completion status of the device programming in automatic mode, the controller must send an opcode. This command is sent after all device write operations. The default opcode 0x05 is used to poll the standard device status register.

32.6.16 QSPI Polling End Register (QSPI_PER)

Offset address: 0x3C

Reset value: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	PCYCN	R/W	0xFFFFFFFF	Polling cycle count: An interrupt is generated after the specified number of polling cycles.

32.6.17 QSPI Interrupt Flag Register (QSPI_IFR)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	-	Reserved
13	POLLF	R/W1C	0x0	Maximum polling cycle count:
12	IND_RSFF	R/W1C	0x0	The indirect read area in SRAM is full, and the indirect operation cannot be completed immediately.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
11	SRFFF	R/W1C	0x0	Small-capacity RX FIFO full (current FIFO status): This can be ignored in non-SPI traditional mode. 0: FIFO not full 1: RX FIFO full
10	SRFNEF	R/W1C	0x0	Small-capacity RX FIFO not empty (current FIFO status): This can be ignored in non-SPI traditional mode. 0: FIFO < RX threshold 1: FIFO ≥ threshold
9	STFFF	R/W1C	0x0	Small-capacity TX FIFO full (current FIFO status): This can be ignored in non-SPI traditional mode. 0: FIFO not full 1: RX FIFO full
8	STFNFF	R/W1C	0x0	Small-capacity TX FIFO not full (current FIFO status): This can be ignored in non-SPI traditional mode. 0: FIFO > threshold 1: FIFO ≤ threshold
7	ROVF	R/W1C	0x0	Receive overflow (occurs only in traditional mode): 0: No overflow detected 1: Overflow occurred
6	IND_TWF	R/W1C	0x0	Exceeding indirect transfer fill level watermark: 0: Watermark not exceeded 1: Watermark exceeded

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
5	AHB_AEF	R/W1C	0x0	Illegal AHB access detected: AHB write WrappingBurst and accesses using SPLIT/RETRY will trigger this interrupt. This interrupt will also be triggered if AHB access is attempted while DAC is disabled.
4	WPAF	R/W1C	0x0	Attempt to write to protected area denied
3	IND_RRF	R/W1C	0x0	No indirect operation request received. Two indirect operations already exist.
2	IND_CF	R/W1C	0x0	The controller has completed the previous indirect operation.
1	UDFF	R/W1C	0x0	Underflow detection: 0: No underflow detected 1: Underflow detected: When the small-capacity TX FIFO is empty, an attempt to send data occurs. An underflow may occur when AHB write data is too slow to keep up with the write requests.
0	RSV	-	-	Reserved

32.6.18 QSPI Interrupt Mask Register (QSPI_IMR)

Offset address: 0x44

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	-	-	Reserved
13:1	INT_EN	R/W	0x0	0: The corresponding interrupt bit in the interrupt flag register (QSPI_IFR) is disabled. 1: The corresponding interrupt bit in the interrupt flag register (QSPI_IFR) is enabled.
0	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.19 QSPI Write Protection Low Register (QSPI_WPLR)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	LBLK_NUM	R/W	0x0	Define the starting block of the write protection block range. The number of bytes per block can be configured through the device size configuration register (QSPI_DSCR). This bit can only be changed when the write protection function (QSPI_WPCR.WPEN) is at a low level.

32.6.20 QSPI Write Protection High Register (QSPI_WPHR)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	HBLK_NUM	R/W	0x0	Defines the ending block of the write protection block range. The number of bytes per block can be configured through the device size configuration register (QSPI_DSCR). This bit can only be changed when the write protection function (QSPI_WPCR.WPEN) is at a low level.

32.6.21 QSPI Write Protection Configuration Register (QSPI_WPCR)

Offset address: 0x58

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	WP_EN	R/W	0x0	Write protection enable: 1: Deny access to any AHB write access (for addresses within the range defined by the write protection low register (QSPI_WPLR) and the write protection high register (QSPI_WPHR)). An AHB error response is generated and an interrupt source is triggered. 0: Write protection is disabled. This bit is synchronized internally by hardware. When changing the state of this bit, there is no special software requirement.
0	WPINV	R/W	0x0	Write protection inversion control: 1: Write protection inversion: The system allows writing to addresses defined by the write protection low register (QSPI_WPLR) and the write protection high register (QSPI_WPHR). 0: Write protection not inverted: The system does not allow writing to addresses defined by the write protection low register (QSPI_WPLR) and the write protection high register (QSPI_WPHR). This bit can only be changed when the write protection function (QSPI_WPCR.WPEN) is at a low level.

32.6.22 QSPI Indirect Read Transfer Control Register (QSPI_IRTR)

Offset address: 0x60

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:6	IND_RNUM	R	0x0	Indirect operation completion count This bit is used in conjunction with the Rdcs bit. It is incremented by hardware when an indirect operation is completed. It is decremented when the Rdcs bit is written with a 1.
5	RDCS	R/W1	0x0	Indirect read completion status This bit is set to 1 by hardware when an indirect operation is completed. Writing 1 clears it.
4	RDQS	R	0x0	Two indirect read operations queued This bit is set to 1 and cleared by hardware.
3	SRAMFS	R/W1	0x0	SRAM is full, and the indirect operation cannot be completed immediately. This bit is set to 1 by hardware and cleared by writing a 1.
2	RDPS	R	0x0	Indirect read operation in progress This bit is cleared only by hardware.
1	RDDIS	W1	0x0	Cancel indirect read Writing a 1 cancels the ongoing indirect read operation and also cancels all current indirect operations. This bit is synchronized internally by hardware.
0	RDST	W1	0x0	Start indirect read Writing a 1 triggers the indirect read operation. To initiate an indirect read operation, first set the indirect read transfer start address register (QSPI_IRTSAR) and the indirect read transfer byte number register (QSPI_IRTNR). This bit is synchronized internally by hardware.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.23 QSPI Indirect Read Transfer Fill Level Watermark Register (QSPI_IRTWR)

Offset address: 0x64

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	VALUE	R/W	0x0	Data fill level: Before the DMA peripheral accesses, this value represents the minimum data fill level of the SRAM. When the SRAM fill level exceeds this value, an interrupt source is triggered. Writing all 0s disables this feature. The unit is in bytes.

32.6.24 QSPI Indirect Read Transfer Start Address Register (QSPI_IRTSAR)

Offset address: 0x68

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ADDR	R/W	0x0	Indirect access start address

32.6.25 QSPI Indirect Read Transfer Byte Number Register (QSPI_IRTNR)

Offset address: 0x6C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	NUM	R/W	0x0	Number of indirect bytes, which can be larger than the SRAM size.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.26 QSPI Indirect Write Transfer Control Register (QSPI_IWTR)

Offset address: 0x70

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:6	IND_WNUM	R	0x0	Indirect operation completion count This bit is used in conjunction with the Wrcs bit. It is incremented by hardware when an indirect operation is completed. It is decremented when the Wrcs bit is written with a 1.
5	WRCS	R/W1	0x0	Indirect write completion status This bit is set to 1 by hardware when an indirect operation is completed. Writing 1 clears it.
4	WRQS	R	0x0	Two indirect write operations queued This bit is set to 1 and cleared by hardware.
3	RSV	-	-	Reserved
2	WRPS	R	0x0	Indirect write operation in progress This bit is cleared only by hardware.
1	WR_DIS	W1	0x0	Cancel indirect write Writing a 1 cancels the ongoing indirect write operation and also cancels all current indirect operations. This bit is synchronized internally by hardware.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	WR_ST	W1	0x0	Start indirect write Writing a 1 triggers the indirect write operation. To initiate an indirect write operation, first set the indirect write transfer start address register (QSPI_IWTSAR) and the indirect write transfer byte number register (QSPI_IWTNR). This bit is synchronized internally by hardware.

32.6.27 QSPI Indirect Write Transfer Fill Level Watermark Register (QSPI_IWTWR)

Offset address: 0x74

Reset value: 0xFFFF FFFF

Bit	Name	Attribute	Reset Value	Description
31:0	VALUE	R/W	0xFFFFFFFF	Data fill level: Before the DMA peripheral accesses, this value represents the maximum data fill level of the SRAM. When the SRAM fill level falls below this value, an interrupt source is triggered. Writing all 1s disables this feature. The unit is in bytes.

32.6.28 QSPI Indirect Write Transfer Start Address Register (QSPI_IWTSAR)

Offset address: 0x78

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ADDR	R/W	0x0	Indirect access start address

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.29 QSPI Indirect Write Transfer Byte Number Register (QSPI_IWTNR)

Offset address: 0x7C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	NUM	R/W	0x0	Number of indirect bytes, which can be larger than the SRAM size.

32.6.30 QSPI Indirect Trigger Address Range Register (QSPI_ITARR)

Offset address: 0x80

Reset value: 0x0000 0004

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3:0	RNGW	R/W	0x4	Indirect range width This field represents the offset address of the indirect AHB address trigger register (QSPI_IATR). When an indirect access is triggered and the AHB address falls within this range, the access request will be sent to the indirect write/read controller. This value is a power of 2, with a default of 2^4 = 16, allowing for a burst of 16 bytes. This bit field reflects the range width, with the number of valid addresses (each address is 32 bits) ranging from the indirect trigger address to the indirect trigger address + <indirect range width - 1> (default value: Indirect trigger address to indirect trigger address + 15).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.31 QSPI Flash Command Control Register (QSPI_FCR)

Offset address: 0x90

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	OPCODE	R/W	0x0	Command opcode The command opcode must be set before triggering the command. For example, 0x20 maps to SubSector erase, and writing to the CMDT bit can send the command. Note that using this method to send commands will involve the instruction type indicated by QSPI_DRIR.IMODE. The command opcode, command address, command dummy bytes, and command data will be transmitted sequentially if this bit is set to 2'b00, be transmitted in parallel through the DQ0 and DQ1 pins if this bit is set to 2'b01, and be transmitted in parallel through the DQ0, DQ1, DQ2, and DQ3 pins if this bit is set to 2'b10.
23	RD_EN	R/W	0x0	Read data enable: Set this bit to 1 if the command in the OPCODE field requires receiving read data bytes.
22:20	RD_NUM	R/W	0x0	Nuber of read data bytes: Up to 8 data bytes can be read: 0: 1 byte 1: 2 bytes … 7: 8 bytes
19	ADDR_EN	R/W	0x0	Command address enable: Set this bit to 1 if the command in the OPCODE field requires an address. This

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				bit shall be set before triggering the command with the CMDT bit.
18	MODB_EN	R/W	0x0	Mode bit enable: Setting this bit to 1 ensures that the mode bits in the mode bit register (QSPI_MBR) are sent following the address byte.
17:16	AD_NUM	R/W	0x0	Number of address bytes: Set the required number of address bytes (the address itself is set in the Flash command address register (QSPI_FCAR)) before triggering the command with the CMDT bit. 00: 1 address byte 01: 2 address bytes 10: 3 address bytes 11: 4 address bytes
15	WR_EN	R/W	0x0	Write data enable: Set this bit to 1 if the command in the OPCODE field requires sending write data bytes.
14:12	WD_NUM	R/W	0x0	Number of write data bytes: Up to 8 data bytes can be written: 0: 1 byte 1: 2 bytes … 7: 8 bytes
11:7	DUM_NUM	R/W	0x0	Number of dummy clock cycles Set the number of dummy cycles required for the command specified in OPCODE (bits 31:24). This bit shall be set before triggering the command with the CMDT bit.
6:2	RSV	-	-	Reserved
1	CMDS	R	0x0	Command execution in progress
0	CMDT	R/W1C	0x0	Execute command This bit is synchronized internally.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.6.32 QSPI Flash Command Address Register (QSPI_FCAR)

Offset address: 0x94

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_ADR	R/W	0x0	Command address: This bit must be set before triggering QSPI_FCR.CMDT. The address corresponds to the QSPI_FCR.OPCODE command.

32.6.33 QSPI Flash Command Read Data Low Register (QSPI_FCRLR)

Offset address: 0xA0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_DL	R	0x0	Command read data (low byte) The bit filed represents any status or data related to read operation configuration returned by the Flash device by triggering the event in the Flash command control register (QSPI_FCR). The register is valid when QSPI_FCR.Cmds is low.

32.6.34 QSPI Flash Command Read Data High Register (QSPI_FCRHR)

Offset address: 0xA4

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_DH	R	0x0	Command read data (high byte) The bit filed represents any status or data related to read operation configuration returned by the Flash device by triggering the event in the Flash command control register (QSPI_FCR). The register is valid when QSPI_FCR.Cmds is low.

32.6.35 QSPI Flash Command Write Data Low Register (QSPI_FCWLR)

Offset address: 0xA8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_DL	R/W	0x0	Command write data (low byte) This bit must be set before triggering QSPI_FCR.CMDT. The bit filed represents any status or data related to write operation configuration written to the Flash device by triggering the event in the Flash command control register (QSPI_FCR).

32.6.36 QSPI Flash Command Write Data High Register (QSPI_FCWHR)

Offset address: 0xAC

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_DH	R/W	0x0	Command write data (high byte) This bit must be set before triggering QSPI_FCR.CMDT. The bit filed represents any status or data related to write operation configuration written to the Flash device by triggering the event in the Flash command control register (QSPI_FCR)

32.6.37 QSPI Polling Flash Status Register (QSPI_PFSR)

Offset address: 0xB0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	PSV	R	0x0	Polling status valid: When the FLSS bit is valid, this bit is set to 1.
7:0	FLSS	R	0x0	Flash status: Define the actual status register of the device.

32.7 Operation Procedure

Before communicating with the Flash device, the software is responsible for configuring the QSPI controller. The static configuration bits of the controller must be set before enabling the controller through QSPI_CR[0]. The purpose of this is to avoid clock edge issues on paths that are not protected against metastability. If the user wishes to change the controller configuration, it is recommended to set the QSPI enable bit to disable before reconfiguration.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

32.7.1 Configuring QSPI Controller after Reset

The QSPI controller can perform basic read and write operations using the direct access controller after waking up in the appropriate state. All target devices support basic read (opcode 0x03) and basic write (opcode 0x02) operations. When the controller wakes up, the baud rate divisor is set to 32. If the target device does not use three address bytes, the device size configuration register (QSPI_DSCR) must be changed to an appropriate value. For all devices tested in this project, the page size is 256 bytes, and no further updates to the register are required. Unless there are strict requirements, software shall cautiously enable the write protection feature before enabling the QSPI controller. The write protection low register (QSPI_WPLR), write protection high register (QSPI_WPHR), write protection configuration register (QSPI_WPCR), and the number of bytes per device block in the device size configuration register (QSPI_DSCR) must be set. Once initialized, and after enabling the controller and DAC, software can read from and write to the Flash device. The controller can be enabled simply by writing to the configuration register (QSPI_CR). Note that the default values of the baud rate divisor and CPOL/CPHA in this register shall not be changed. It is recommended to write the value 0x00780081.

32.7.2 QSPI Controller Configuration Optimization

To optimize access to the Flash, the software must accurately configure the controller.

1. Wait until all pending STIG or INDAC operations are completed or poll QSPI_CR[31].
2. Disable the DAC (QSPI_CR[7]). It is not necessary to completely disable the QSPI controller (QSPI_CR[0]).
3. Update the device instruction configuration registers (QSPI_DRIR/QSPI_DWIR) according to the types of instructions the user wishes to utilize.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

4. If the mode bit in the device instruction configuration register (QSPI_DRIR[20]) is enabled, update the mode bit register (QSPI_MBR).

5. If the contents are incorrect, update the device size configuration register (QSPI_DSCR). Note that after initialization, part or all of the register may have been updated. The number of address bytes is a critical configuration during read and write operations. The number of bytes per page must be set when performing any write operations. The number of bytes per device block is only required when using write protection. If the default value of the target device is correct, or if certain values (excluding the number of address bytes) are incorrect, device write operations are not allowed.

6. Update the QSPI device delay register (QSPI_DDLR). This register allows the user to adjust how the chip select is driven after each Flash access. This is because each device may have different timing requirements. As the serial clock frequency increases, the timing requirements become increasingly important. Note that the value set in this register is based on the ref_clk cycle. For example, ATMEL devices require at least 50 ns of time before resetting CS to active after CS is deactivated. By default, the controller provides only a minimum of 1 SCLK cycle. When the device operates at 100 MHz, the SCLK cycle is only 10 ns, thus requiring an additional 40 ns. When ref_clk runs at 400 MHz (2.5 ns period), the CSDA bit in this register must be set to at least 16. This delay can be increased during the automatic polling phase. The polling repeat delay can be defined in the write complete control register (QSPI_WCR).

7. If required, update the address remapping register (QSPI_RAR). This only affects the DAC path.

8. If required, set and enable the write protection low register (QSPI_WPLR), write protection high register (QSPI_WPHR), and write protection configuration register (QSPI_WPCR) if they have not been set in previous initializations.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9. Enable the required interrupts through the interrupt mask register (QSPI_IMR).
10. Set the baud rate divider in the configuration register (QSPI_CR[22:19]) to define the clock frequency required for the target device.
11. Update the read data capture register (QSPI_RDCR). This register can introduce a delay when capturing read data. It is helpful when the read data path from the device to the controller is long and the device clock frequency is high.
12. Enable the QSPI controller and DAC through the configuration register (QSPI_CR[0]).

32.7.3 Usage of Flash Command Control Register (STIG Operation)

The Flash command control register (QSPI_FCR) provides a flexible and programmable way to access Flash devices, known as STIG operations. The instruction OPCODE, the number of address bytes, the address itself, the number of dummy bytes, the number of write data bytes, the write data itself, and the number of read data bytes can all be configured. Once configured, software can trigger the command through QSPI_FCR[0] and poll QSPI_FCR[1] to wait for it to be accepted. When this bit becomes inactive, another STIG can be triggered. This typical method of accessing Flash can be used to access Flash device registers and also to perform erase operations. Although a maximum of 8 data bytes (defined by the Flash command read and write data registers (QSPI_FCRLR / QSPI_FCRHR / QSPI_FCWLR / QSPI_FCWHR)) can be read or written at a time, this method can also be used to access the Flash array itself. Compared to all other read accesses via AHB, the commands sent through this interface have higher priority.

32.7.4 Traditional SPI Mode

The traditional SPI mode allows software to directly access the internal TX FIFO and RX FIFO, thereby bypassing the direct, indirect, and STIG controllers. The traditional mode enables users

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

to send arbitrary Flash commands, but it incurs significant software overhead to ensure effective management of the FIFO fill level. The traditional SPI core supports bidirectional transmission, and once the chip select is enabled, data can be continuously transmitted in either direction. Even if the driver only intends to read data from the Flash device, dummy data must be written to maintain the validity of the chip select. Conversely, for write operations, the same applies. This means that to read 4 bytes from a device (with 3 address bytes), the software needs to write a total of 8 bytes into the TX FIFO. The first byte is the instruction OPCODE, the next 3 bytes are the address, and the last 4 bytes are dummy data. Similarly, since 8 bytes are written into the TX FIFO, the software expects to receive 8 bytes back into the RX FIFO. The first 4 bytes will be discarded, while the last 4 bytes are the desired read data. Given the limited fill level of the TX FIFO and RX FIFO, the software is responsible for maintaining the FIFO fill level and ensure that the TX FIFO does not underflow and the RX FIFO does not overflow during command execution. This can increase software overhead. An interrupt will occur when the data fill level exceeds the watermark. When traditional mode is enabled, the software can access the TX FIFO and write any value to any address of the QSPI controller through the AHB interface. Similarly, when traditional mode is enabled, the software can access the RX FIFO and read any address from the QSPI controller via the AHB interface.

32.7.5 Entering and Exiting XIP Mode

32.7.5.1 Entering XIP Mode from POR

If XIP is enabled in non-volatile configuration, it can enter XIP mode in a non-volatile manner. Only a small portion of Flash devices support this function. Since the only operation that Flash devices can recognize is the XIP read operation, software cannot discover the XIP state after POR by polling the Flash status register (QSPI_PFSR). If it is known that the device will enter XIP from POR, the QSPI_MBR register and QSPI_CR[18] shall be set up in the initial BOOT. If it is

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

unknown whether the device will enter XIP from POR, and the connected Flash device supports entering XIP from POR, software can attempt to exit XIP mode by sending an XIP exit command through the STIG command. Software needs to understand the mode bit requirements of the device, as the XIP entry and exit of each device are different. Micron N25Q and MT25 support entering and exiting XIP from POR. When exiting XIP mode, set OPCODE to 8'h00, the number of address bytes to 3, the number of dummy cycles to 16, and the number of read bytes to 1. The address shall be configured as {8 mode bits, 16'h0000}. For Micron devices, the mode bits shall be set to 8'b10000000. Note that sending the NVCR command through the Flash command control register (QSPI_FCR) can enable entering XIP from POR, which will take effect until the next POR sequence.

32.7.5.2 Other Cases of Entering XIP Mode

Most Flash devices support XIP mode. However, there is no consistent standard among Flash manufacturers. Most devices send the signature bits directly after the address bytes. Other devices, such as Micron, require writing to the Flash device configuration register to enable XIP while sending the signature bits. For Flash devices that must comply with this controller, the following steps shall be followed to enter XIP mode. Micron N25Q and MT25 (which do not support basic XIP) enable XIP mode by setting the VCR[3] within the Flash device. The steps to send the VCR write command via Flash command control register (QSPI_FCR) are as follows:

1. Disable the direct access controller and the indirect access controller to ensure no new AHB read accesses are sent to the Flash device.
2. Configure the Flash command control register (QSPI_FCR) to send the VCR write to the Flash memory.
3. Set the XIP mode bit in the mode bit register (QSPI_MBR[7:0]) to 8'b0000000.
4. Enable the XIP mode of the self-controller by setting QSPI_CR[17].

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. If necessary, re-enable the direct access controller and the indirect access controller.

Other Micron devices (supporting basic XIP)

1. Disable the direct access controller and the indirect access controller to ensure no new AHB read accesses are sent to the Flash device.
2. Set the XIP mode bit in the mode bit register (QSPI_MBR[7:0]) to 8'b1000000.
3. Enable the XIP mode of the self-controller by setting QSPI_CR[17].
4. If necessary, re-enable the direct access controller and the indirect access controller.

Winbond devices

1. Disable the direct access controller and the indirect access controller to ensure no new AHB read accesses are sent to the Flash device.
2. Set the XIP mode bit in the mode bit register (QSPI_MBR[7:0]) to 8'b0010000.
3. Enable the XIP mode of the self-controller by setting QSPI_CR[17].
4. If necessary, re-enable the direct access controller and the indirect access controller.

Spansion devices

1. Disable the direct access controller and the indirect access controller to ensure no new AHB read accesses are sent to the Flash device.
2. Set the XIP mode bit in the mode bit register (QSPI_MBR[7:0]) to 8'b1010000.
3. Enable the XIP mode of the self-controller by setting QSPI_CR[17].
4. If necessary, re-enable the direct access controller and the indirect access controller.

32.7.5.3 Exiting XIP Mode

To exit XIP mode, the software must disable the direct access controller and the indirect access controller to ensure no new AHB read accesses are sent to the Flash device. Then, the mode

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

bits shall be set to any value other than the specific mode bits corresponding to the Flash device specifications. Next, the software must reset QSPI_CR[17]. Note that before disabling the internal XIP mode state, the Flash device must receive a read command, which means that XIP mode will remain active until the next read command is responded to. Ensure that the XIP mode is disabled before the end of the read sequence.

32.7.6 Indirect Data Transfer Mode

1. Initialize the QSPI-related GPIO pins.
2. Initialize the QSPI clock and release the reset, waiting for QSPI_CR[31] to be set.
3. Set the QSPI_DDLR register to configure the communication timing delay.
4. Set the QSPI_DSCR register to establish the device address, device page size, and device block size.
5. Set the QSPI_CFGR register to configure the QSPI operating mode, main frequency, and enable INDAC mode.
6. Set the QSPI_RDCR register to configure the data capture delay.
7. If the above configuration commands require the device to enter QE mode, send the relevant entry commands to the device and wait until it is idle; if not, proceed to step 8.
8. To perform an indirect write operation, configure steps 9–15; to perform an indirect read operation, configure steps 16–22.
9. Set QSPI_IWTSAR to configure the starting address for the indirect write operation.
10. Set QSPI_IWTNR to configure the number of bytes for the indirect write operation.
11. Set QSPI_IATR to configure the external flash address for the indirect write operation.
12. Set QSPI_ITARR to configure the transfer address range for the indirect write operation.
13. Set QSPI_IWTR[5] to 1 to complete the indirect write operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

14. Directly write a 32-bit value to the external flash address.
15. Wait for QSPI_IWTR[5] to be 1, indicating that the indirect write operation is complete, then write 1 to clear it.
16. Set QSPI_IRTSAR to configure the starting address for the indirect read operation.
17. Set QSPI_IRTNR to configure the number of bytes for the indirect read operation.
18. Set QSPI_IATR to configure the external flash address for the indirect read operation.
19. Set QSPI_ITARR to configure the transfer address range for the indirect read operation.
20. Set QSPI_IRTR[0] to 1 to start the indirect read operation.
21. Directly read a 32-bit value from the external flash address.
22. Wait for QSPI_IRTR[5] to be 1, indicating that the indirect read operation is complete.

32.7.7 AHB Address Remapping

The remapping feature of QSPI defines how to translate the received AHB addresses when accessed through the direct access controller. By default, it is a 1:1 mapping. When QSPI_CR[16] is enabled, the received AHB address is remapped to address + N, where the value of N is stored in the address remapping register (QSPI_RAR). It is recommended to disable QSPI before configuring the address remapping register (QSPI_RAR).

32.7.8 Interrupt Response

QSPI provides a high-active interrupt pin. For information regarding all interrupt sources, please refer to the interrupt flag register (QSPI_IFR). When the software reads the interrupt flag register (QSPI_IFR), the interrupt source is cleared. The QSPI controller can also be configured such that writing to the interrupt flag register (QSPI_IFR) clears the interrupt. The default clearing method is write-to-clear. The interrupt mask register (QSPI_IMR) can mask all interrupts.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Writing 0 enables the interrupt, while writing 1 clears it.

32.7.9 AHB Protection Register

When a POR occurs, the AHB protection mechanism is disabled. Software can use the protection register to prevent AHB from writing to specific registers. It is recommended to disable QSPI before configuring the protection register.

32.7.10 Example of DAC Mode Configuration Process

1. Initialize the QSPI-related GPIO pins.
2. Initialize the QSPI clock and release the reset, waiting for QSPI_CR[31] to be set.
3. Set the QSPI_DRIR register to configure the dummy clock for commands in DAC mode, the data transfer address width, the address transfer width, and the read command.
4. Set the QSPI_DWIR register to configure the data transfer address width, address transfer width, and write command for commands in DAC mode.
5. Set the QSPI_DDLR register to configure the communication timing delay.
6. Set the QSPI_DSCR register to establish the device address, device page size, and device block size.
7. Set the QSPI_CFGR register to configure the QSPI operating mode, main frequency, and enable DAC mode.
8. Set the QSPI_RDCR register to configure the data capture delay.
9. If the above configuration commands require the device to enter QE mode, send the relevant entry commands to the device and wait until it is idle; if not, proceed to step 10.
10. At this point, the initialization of the flash in DAC mode is complete, and read, write, and erase operations can be performed on the flash.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33 Secure Digital Input and Output (SDIO)

33.1 Overview

As a data transmission interface, SDIO controller compliant with SD and eMMC standard protocols can be used as the master controller of SD card reader and eMMC card reader, and also supports secure digital I/O (SDIO).

33.2 Main Features

- SD2.0 and eMMC4.4.1 protocols
- Built-in FIFO with a width of 32 bits and a depth of 16, can stop the clock at overrun and underrun
- CRC generation and verification
- Programmable baud rates, with a clock distribution ratio to meet communication requirements at various baud rates
- Clock control switch
- Card detection
- Card write protection
- SDIO interrupt in 1-bit, 4-bit and 8-bit modes
- Block size ranging from 1 to 65536 bytes
- DMA transfer

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.3 Pin Description

Table 33-1: SDIO Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
SDIO_CK	PC12	Output	Clock
SDIO_CMD	PD2	Output	Command
SDIO_DATA0	PC8	Input/output	Data line
SDIO_DATA1	PC9	Input/output	Data line
SDIO_DATA2	PC10	Input/output	Data line
SDIO_DATA3	PC11	Input/output	Data line
SDIO_DATA4	PB8	Input/output	Data line
SDIO_DATA5	PB9	Input/output	Data line
SDIO_DATA6	PC6	Input/output	Data line
SDIO_DATA7	PC7	Input/output	Data line
SDIO_CD	PB4	Input	Card detection signal, active low
SDIO_WP	PB5	Input	Write protection
SDIO_RSTN	PB6	Output	eMMC reset output signal

33.4 Functional Description

33.4.1 Internal DMA (IDMA)

IDMA features a control and status register (CSR) and a transmit/receive engine that transfers data between the host memory and device ports in both directions. The controller efficiently moves data from the source to the destination using descriptors with minimal CPU intervention. Once programmed, the controller can interrupt the host CPU upon the completion of data transmission and reception, as well as under other normal or error conditions.

IDMA and the host driver communicate through a single data structure. CSR addresses from 0x80 to 0x98 are reserved for host programming.

IDMA transfers the data received from the card to the host data buffer and transfers the data

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

from the host data buffer to the FIFO. Descriptors residing in host memory serve as pointers to these buffers.

The data buffers are located in the physical memory of the host and consist of either complete data or partial data. The buffers contain only data, while the buffer status is retained in the descriptors. A data chain refers to data that spans multiple data buffers. However, a single descriptor cannot span multiple data.

The base address of the list is written to DBADDR. The descriptor list is forward-linked, and the last descriptor can point back to the first entry to create a circular structure. The descriptor list is located in the physical memory address space of the host, with each descriptor capable of pointing to a maximum of two data buffers.

Descriptors

IDMA uses the following types of descriptor structures:

1. Double buffer structure: The distance between two descriptors is determined by the DSL field of the BMOD register.

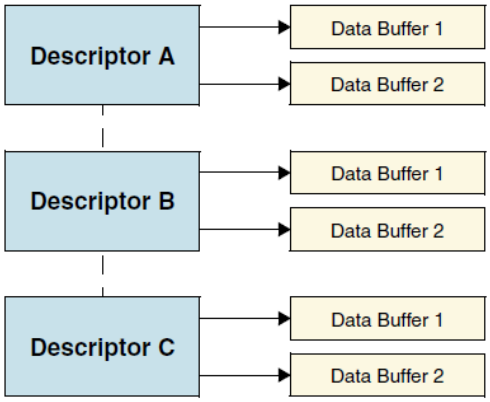

Figure 33-1: Double Buffer Structure Diagram

2. Chain structure: Each descriptor points to a unique buffer and the next descriptor.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

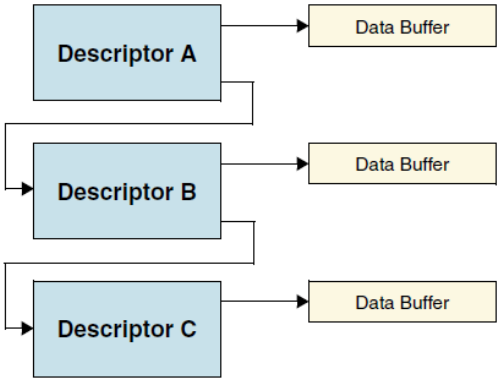

Figure 33-2: Chain Structure Diagram

Descriptor addresses must be aligned with the 32-bit AHB data bus width. Each descriptor contains 16 bytes of control and status information.

DES0 represents bits [31:0], DES1 represents bits [63:32], DES2 represents bits [95:64], and DES3 represents bits [127:96].

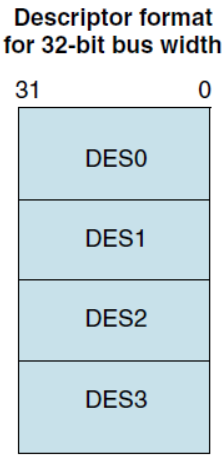

Figure 33-3: DES0 ~ S3 Structure Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.4.1.1 DES0

Table 33-2: DES0 Definition

Bit	Name	Description
31	OWN	0: The descriptor is owned by the host. 1: The descriptor is owned by IDMA. IDMA clears this bit upon completion of the data transfer.
30	Card Error Summary (CES)	It indicates the status of transactions to/from the card and is also reflected in RINTSTS. Logical OR of the following bits: ■ EBE: End Bit Error ■ RTO: Response Timeout/Boot ACK Timeout ■ RCRC: Response CRC ■ SBE: Start Bit Error ■ DRTO: Data Read Timeout/BDS Timeout ■ DCRC: Data CRC for Receive ■ RE: Response Error
29:6	RSV	Reserved
5	End of Ring (ER)	When this bit is set to 1, it indicates that the descriptor list has reached the last descriptor. IDMA returns to the base address of the list, creating a descriptor ring. This is meaningful only for double-buffer descriptor structure.
4	Second Address Chained (CH)	When this bit is set to 1, it indicates that the second address in the descriptor is the address of the next descriptor, rather than the address of the second buffer. When this bit is set, BS2 (DES1[25:13]) shall be all zeros.
3	First Descriptor (FS)	When this bit is set to 1, it indicates that this descriptor contains the first buffer of data. If the size of the first buffer is 0, the next descriptor contains the beginning of the data.
2	Last Descriptor (LD)	This bit is associated with the last block of the DMA transfer. When set to 1, it indicates that the buffer pointed to by this descriptor is the last buffer of data. When this descriptor is completed, the remaining byte count will be 0.
1	Disable Interrupt on Completion (DIC)	When this bit is set to 1, it will prevent the setting of the TI/RI bits in IDSTS, indicating that the buffer pointed to by the descriptor contains the last data.
0	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.4.1.2 DES1

Table 33-3: DES1 Definition

Bit	Name	Description
31:26	RSV	Reserved
25:13	Buffer 2 Size (BS2)	This field indicates the byte size of the second data buffer. If this field is 0, the DMA will ignore this field. If it is a double-buffer structure, it will return to the next buffer. This field is invalid for chain structures (i.e., when DES0[4] is set). If the buffer 2 size is set to 0 in any descriptor, the value of buffer 2 size cannot be non-zero for the remaining descriptors before the end of the descriptor.
12:0	Buffer 1 Size (BS1)	This field indicates the byte size of the data buffer, which cannot be zero. Note: If only one buffer is required to be programmed, only buffer 1 shall be used, and buffer 2 is not required.

33.4.1.3 DES2

Table 33-4: DES2 Definition

Bit	Name	Description
31:0	Buffer Address Pointer 1	When using a double-buffer structure, these bits indicate the physical address of the first data buffer. For chain descriptors, these bits indicate the physical address of the data buffer.

33.4.1.4 DES3

Table 33-5: DES3 Definition

Bit	Name	Description
31:0	Buffer Address Pointer 2/ Next Descriptor Address (BAP2)	When using a double-buffer structure, these bits indicate the physical address of the second buffer. If the second address chain (DES0[4]) bit is set, this address contains a pointer to the physical memory where the next descriptor is located.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.4.2 Interrupt

Table 33-6: SDIO Interrupt Source Descriptions

Bit	Interrupt	Description
15	Final bit error/ no CRC/ CRC error	In read operations, the final bit of the data is not 1; in write operations, there is no data CRC or a CRC error occurs.
14	Auto stop command	A stop command is automatically sent by the host when the data transmission ends.
13	Start bit error	In read operations, the start bit of the data is not 0; in 4-bit mode, all start bits are not 0.
12	Rewrite lock register error	An attempt is made to rewrite the locked command-related register within 6 system clock cycles after the command start bit is asserted.
11	FIFO underrun / overrun	The FIFO is full, but the host continues to move data into the FIFO; or the FIFO is empty, but the host continues to fetch data from the FIFO.
10	Data loss timeout interrupt	If the FIFO is full and there is still an attempt to read data from the card, or if the FIFO is empty and there is still an attempt to write data to the card, the output clock will be stopped to avoid data loss. If the problem is not resolved within the timeout register's output clock time, an interrupt will be generated.
9	Read data timeout	This interrupt is generated when the read data times out, and a data transmission complete interrupt will also be generated.
8	Response timeout	This interrupt is generated when the response times out, and a command complete interrupt will also be generated.
7	Data CRC error	The CRC of the data received from the card does not match the internally generated CRC.
6	Response CRC error	The CRC of the response received from the card does not match the internally generated CRC.
5	Receive data FIFO request	In card read operations, FIFO count is greater than or equal to rx_level.
4	Transmit data FIFO request	In card read operations, FIFO count is less than or equal to tx_level.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Interrupt	Description
3	Data transfer complete	This interrupt is generated when the data transfer is complete, even if there are start bit errors or CRC errors. When a read data timeout interrupt occurs, this bit will also be set. (Suggestion: After the data transfer is complete, the host shall move the remaining data out of the FIFO.)
2	Command complete	This interrupt is generated when a command is sent and a response from the card is received, even if the response is erroneous or has a CRC error. This bit will also be set when a response timeout occurs.
1	Response error	This occurs in the following cases: 1. The start bit of the response is not 0. 2. The command index does not match. 3. The termination bit of the response is not 1.
0	Card detection	When a card is inserted or removed.

33.4.3 Data Transfer

The host will begin data transfer two clock cycles after receiving the response to a data write command. This is the case even if there is a response error or a response CRC error. However, if a response timeout occurs, the data transfer will be halted.

- Single block transfer

  If the transfer mode is block transfer and the byte count register is equal to the block size register, a single block transfer will occur.

- Multiple block transfer

  If the transfer mode is block transfer and the byte count register is not equal to the block size register, a multiple block transfer will occur.

  If the byte number is 0 while the block size is not 0, this indicates an endless transfer. The host will continue to transfer data until a stop command is sent.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Automatic stop

  When the send_auto_stop bit in the command register is set, the host will send an additional block transfer command to transmit any remaining data in the FIFO, followed by a stop command.

- Clock control

  Low-power mode for the clock: Setting the low-power mode bit in the clock control register to 1 will put the output clock into low-power mode. The output clock will be turned off after at least 8 system clock cycles when the card is in the idle state. The low-power mode bit will be refreshed when a new command is sent. Additionally, the output clock will be turned off when a FIFO overrun or underrun occurs.

  Summary: The output clock will be turned off under the following conditions: clk_en = 1

  The card is in idle state for at least 8 system cycles in low-power mode.

  The FIFO is full and cannot accept more data from the card, so the clock shall be turned off to prevent FIFO overrun and data loss. The FIFO is empty and cannot send data to the card, so the clock shall be turned off to prevent FIFO underrun.

  Note: To change the output clock frequency, disable the clock enable first before making any changes.

- Error detection and response:

  Response timeout: No start bit of the response is received within the number of clock cycles specified in the timeout register. Response CRC error: The CRC7 of the response does not match the internally generated CRC7.

  Response error: The start bit of the response is not 0, or the command index does not match the sent command index, or the termination bit of the response is not 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Data transmission:

No CRC state: During a data write transfer, if no start bit of the CRC is received two output clock cycles after the last bit of the data section, a data CRC error is generated, and the corresponding bit in the interrupt register is set to 1. The host will then stop subsequent data transfers. If the CRC status following the CRC start bit is not 010, a data CRC error will also be generated, and the corresponding bit in the interrupt register will be set to 1.

FIFO underrun: A data transfer error will occur when a FIFO underrun happens. Data reception timeout: During a data read transfer, if no start bit of the data is received within the number of clock cycles specified in the timeout register, a data timeout will occur, and a DTO (data timeout) interrupt will be generated. The host will then stop subsequent data transfers.

Start bit error: In a 4-bit or 8-bit read transfer, if there is no start bit on the data line, a start bit error will be generated.

Data CRC error: During a data read transfer, if the received CRC16 does not match the internally generated CRC16, a CRC data error will be generated, and subsequent data transfers will be halted.

Stop bit error: During a read transfer, if the stop bit is not 1, a stop bit error will be generated, and subsequent data transfers will be halted.

FIFO overrun: A data transfer error will occur when a FIFO overrun happens.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5 Register Description

SDIO register base address: 0x4050_0000

The registers are listed below:

Table 33-7: List of SDIO Registers

Offset Address	Name	Description
0x00	SDIO_CTRL	Control register
0x04	SDIO_POWEN	Power enable register
0x08	SDIO_CLKDIV	Clock division register
0x10	SDIO_CLKENA	Clock enable register
0x14	SDIO_TIMEOUT	Timeout register
0x18	SDIO_WIDTH	Bit width register
0x1C	SDIO__BLKSIZ	Block size register
0x20	SDIO_BYTCNT	Byte count register
0x24	SDIO_INTMASK	Interrupt mask register
0x28	SDIO_CMDARG	Command parameter register
0x2C	SDIO_CMD	Command register
0x30	SDIO_RESP0	Response 0 register
0x34	SDIO_RESP1	Response 1 register
0x38	SDIO_RESP2	Response 2 register
0x3C	SDIO_RESP3	Response 3 register
0x40	SDIO_MNTSTS	Maskable interrupt status register
0x44	SDIO_RINTSTS	Raw interrupt status register
0x48	SDIO_STATUS	Status register
0x4C	SDIO_FIFOTH	FIFO compare register
0x50	SDIO_CDETECT	Card detection register
0x54	SDIO_WRTPRT	Write protection register
0x5C	SDIO_TCBCNT	TCBCNT register
0x60	SDIO_TBBCNT	TBBCNT register
0x64	SDIO_DEBOUNCE	DEBOUNCE register
0x78	SDIO_RST	Hardware reset register
0x80	SDIO_BMOD	Bus mode register
0x84	SDIO_PLDMND	Poll demand register
0x88	SDIO_DBADDR	Descriptor list base address register
0x8C	SDIO_IDSTS	Internal DMA status register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Description
0x90	SDIO_IDINTEN	Internal DMA interrupt enable register
0x94	SDIO_DSCADDR	Current host descriptor address register
0x98	SDIO_BUFADDR	Current buffer descriptor address register
0x110	SDIO_ENABLESHIFT	Phase shift enable register
0x200	SDIO_DATA	Data register

33.5.1 Control Register (SDIO_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	Reserved
25	USE_INTERNAL_DMAC	R/W	0x0	Applicable only for internal DMA configuration; otherwise, reserved: 0: The host performs data transfer via the slave interface. 1: Data is transferred using internal DMA.
24	OD_PULLUP_EN	R/W	0x0	Open-drain pull-up enable: 0: No open-drain pull-up 1: Open-drain pull-up enabled When this bit is set, the command is typically output in open-drain mode; that is, it drives only 0 or high impedance, and cannot drive 1.
23:12	RSV	-	-	Reserved
11	CEATA_DEVICE_INTERRUPT_STATUS	R/W	0x0	0: Interrupts are disabled in CE-ATA devices. 1: Interrupts are enabled in CE-ATA devices.
10	SEND_AUTO_STOP_CCSD	R/W	0x0	0: If the module has not reset this bit, clear this bit. 1: Send CCSD to the CE-ATA device.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
9	SEND_CCSD	R/W	0x0	0: If the module has not reset this bit, clear this bit. 1: After sending CCSD, send an internally generated STOP to the CE-ATA device.
8	ABORT_READ_DATA	R/W	0x0	0: No change 1: If a suspend command is sent during a read data operation, software must poll each card to determine when the suspend occurred. Once the suspend is sent, software sets this bit to reset the data state machine (which was previously waiting for the next data block). After the data state machine is reset, this bit is automatically cleared.
7	SEND_IRQ_RESP	R/W	0x0	0: No change 1: Automatically send IRQ response To wait for the MMC card interrupt, the host sends CMD40. If the host wishes to exit the wait-for-interrupt state, this bit can be set to 1, causing the CMD state machine to exit the wait-for-interrupt state and return to the idle state.
6	READ_WAIT	R/W	0x0	0: Clear read wait. 1: Send read_wait to the SDIO card.
5	RSV	-	-	Reserved
4	INT_EN	R/W	0x0	Interrupt enable: 0: Disable interrupt. 1: Enable interrupt.
3	RSV	-	-	Reserved
2	DMA_RST	R/W	0x0	DMA reset: 0: No effect 1: DMA reset After DMA reset, this bit is automatically cleared.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	FIFO_RST	R/W	0x0	FIFO reset: 0: No effect 1: FIFO reset After FIFO reset, this bit is automatically cleared.
0	HOST_RST	R/W	0x0	Host reset: 0: No effect 1: Host reset After host reset, this bit is automatically cleared.

33.5.2 Power Enable Register (SDIO_POWEN)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	POWER_ENABLE	R/W	0x0	0: Power off 1: Power on

33.5.3 Clock Division Register (SDIO_CLKDIV)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	CLK_DIV0	R/W	0x0	Bits 7 to 0: The clock division for clock 0 is 2*n. For example, if clk_div0 = 0, the division factor is 2*0 = 0, meaning there is no division, and the output card clock is at the original frequency; if clk_div0 = 1, the division factor is 2*1 = 2, meaning the clock is divided by 2, and the output card clock is half of the original clock frequency, and so on.

33.5.4 Clock Enable Register (SDIO_CLKENA)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	CLK_LOW_POWER	R/W	0x0	Low power control: 0: Non-low power mode 1: Low power mode When the card is in idle state, the clock is stopped.
15:1	RSV	-	-	Reserved
0	CLK_EN	R/W	0x0	Card clock output enable: 0: Clock disabled 1: Clock enabled

33.5.5 Timeout Register (SDIO_TIMEOUT)

Offset address: 0x14

Reset value: 0xFFFF FF40

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:8	DATA_TIMEOUT	R/W	0xFFFFFF	Timeout for reading from or writing to the card The timeout counter starts only after the card clock is stopped.
7:0	RESPONSE_TIMEOUT	R/W	0x40	The timeout period for the response is measured in the number of cclk_out clock cycles.

33.5.6 Bit Width Register (SDIO_WIDTH)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	WIDTH1	R/W	0x0	0: Non 8-bit mode 1: 8-bit mode Note: The width 1 register takes precedence over the width 0 register. When width 1 is set to 1, the controller operates in 8-bit mode.
15:1	RSV	-	-	Reserved
0	WIDTH0	R/W	0x0	0: 1-bit mode 1: 4-bit mode

33.5.7 Block Size Register (SDIO_BLKSIZ)

Offset address: 0x1C

Reset value: 0x0000 0200

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:0	BLOCK_SIZE	R/W	0x200	The number of bytes contained in one block.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.8 Byte Count Register (SDIO_BYTCNT)

Offset address: 0x20

Reset value: 0x0000 0200

Bit	Name	Attribute	Reset Value	Description
31:0	BYTE_CNT	R/W	0x200	The number of bytes of data to be transmitted. The byte_cnt shall be an integer multiple of block_size. If byte_cnt is 0, it indicates that the quantity of data to be transmitted is undefined. In this case, the host shall send a stop or abort command to terminate the data transmission.

33.5.9 Interrupt Mask Register (SDIO_INTMASK)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	SDIO_INT_MASK	R/W	0x0	Masking SDIO card interrupts involves clearing the corresponding bit to mask the interrupt and setting it to 1 to enable the interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	INT_MASK	R/W	0x0	Mask interrupts for each bit: 0: Interrupt masked 1: Interrupt enabled Bit 15: End-bit error (read) / Write without CRC Bit 14: Auto command completion Bit 13: Start-bit error / Busy clear interrupt Bit 12: Hardware locked write error Bit 11: FIFO underrun / overrun Bit 10: Data missing timeout interrupt Bit 9: Read data timeout Bit 8: Response timeout Bit 7: Data CRC error Bit 6: Response CRC error Bit 5: RX FIFO data request Bit 4: TX FIFO data request Bit 3: Data transfer complete Bit 2: Command end Bit 1: Response error Bit 0: Card detection

33.5.10 Command Parameter Register (SDIO_CMDARG)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CMD_ARG	R/W	0x0	Command parameters passed to the card

33.5.11 Command Register (SDIO_CMD)

Offset address: 0x2C

Reset value: 0x2000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31	START_CMD	R/W	0x0	Start command. When start_cmd is set to 1, the host shall not attempt to write to any command registers; if it does, the hardware lock error interrupt in the raw interrupt register will be set to 1. After the start command is issued, this bit will be cleared to 0.
30	RSV	-	-	Reserved
29	USE_HOLD_REG	R/W	0x1	Use hold register: 0: CMD and DATA are sent to the card bypassing the hold register. 1: CMD and DATA are sent to the card through the hold register.
28	VOLT_SWITCH	R/W	0x0	0: No voltage switching. 1: Enable voltage switching, set only during CMD11.
27	BOOT_MODE	R/W	0x0	0: Force boot 1: Alternate boot
26	DISABLE_BOOT	R/W	0x0	Disable boot: Do not set both enable_boot and disable_boot simultaneously.
25	EXPECT_BOOT_ACK	R/W	0x0	Expect boot ACK: When this bit is set together with enable_boot, the selected card is expected to respond with a 0-1-0 boot ACK.
24	ENABLE_BOOT	R/W	0x0	Enable boot: Only set in force boot mode. Do not set both enable_boot and disable_boot simultaneously.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
23	CCS_EXPECTED	R/W	0x0	0: CE-ATA device disables interrupts, or the command does not expect a CCS from the device. 1: CE-ATA device enables interrupts, and RW_BLK commands require the CE-ATA device to send a command completion signal.
22	READ_CEATA_DEVICE	R/W	0x0	0: The host does not perform a read operation on the CE-ATA device. 1: The host performs a read operation on the CE-ATA device.
21	UPDATE_CLK	R/W	0x0	0: Normal command sequence 1: Do not send command, only update the values of clock-related registers. Clock-related registers include: CLKDIV, CLRSRC, CLKENA
20:17	RSV	-	-	Reserved
16	CARD_NUM	R/W	0x0	Select card number
15	SEND_INI_SEQ	R/W	0x0	0: Do not send an initialization sequence (80 card clocks) before sending the command. 1: Send an initialization sequence (80 card clocks) before sending the command. After power on, 80 clocks shall be sent to the card before sending any command.
14	STOP_ABORT_CMD	R/W	0x0	When sending a stop command (CMD12), this bit must be set to 1. If a reset command (CMD0, CMD15, CMD52_reset) is to be sent during data transfer, this bit must also be set to 1 to stop the data transfer after the command is sent.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13	WAIT_PRV_DATA_FINISH	R/W	0x0	0: Send the command immediately, even if the previous data transfer is not yet complete. 1: Wait for the previous data transfer to complete before sending the command. There are two cases: 1) If the previous data transfer has already been completed or if the byte count register is 0, the command is sent immediately. 2) If the byte count register is not 0, the command is sent after the transfer is complete.
12	SEND_AUTO_STOP	R/W	0x0	0: Do not send a stop command after the data transfer is complete. 1: Send a stop command after the data transfer is complete.
11	TRANSFER_MODE	R/W	0x0	0: Block data transfer command 1: Stream data transfer command This bit has no effect when there is no data transfer.
10	READ_WRITE	R/W	0x0	0: Read data from the card. 1: Write data to the card.
9	DATA_TRANSFER_EXPECTED	R/W	0x0	0: Do not expect data transfer. 1: Expect data transfer.
8	CHECK_REP_CRC	R/W	0x0	0: Do not check response CRC. 1: Check response CRC.
7	REP_LONG	R/W	0x0	0: Expect a 48-bit response from the card. 1: Expect a 136-bit response from the card.
6	REP_EXPECT	R/W	0x0	0: Do not expect a response from the card. 1: Expect a response from the card.
5:0	CMD_INDEX	R/W	0x0	Command index

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.12 Response 0 Register (SDIO_RESP0)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RESPONSE0	R	0x0	Bits [31:0] of the response

33.5.13 Response 1 Register (SDIO_RESP1)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RESPONSE1	R	0x0	Bits [63:32] of the response

33.5.14 Response 2 Register (SDIO_RESP2)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RESPONSE2	R	0x0	Bits [95:64] of the response

33.5.15 Response 3 Register (SDIO_RESP3)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RESPONSE3	R	0x0	Bits [127:96] of the response

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.16 Maskable Interrupt Status Register (SDIO_MINTSTS)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	SDIO_INTERRUPT	R	0x0	The SDIO card interrupt indicates that the card has generated an interrupt. The interrupt bit is only valid if the corresponding bit in the interrupt mask register is not masked.
15:0	INT_STATUS	R	0x0	Interrupts are enabled only when the corresponding bit in the interrupt mask register is set. Bit 15: End-bit error (read) / Write without CRC Bit 14: Auto command completion Bit 13: Start-bit error / Busy clear interrupt Bit 12: Hardware locked write error Bit 11: FIFO underrun / overrun Bit 10: Data missing timeout interrupt Bit 9: Read data timeout Bit 8: Response timeout Bit 7: Data CRC error Bit 6: Response CRC error Bit 5: RX FIFO data request Bit 4: TX FIFO data request Bit 3: Data transfer complete Bit 2: Command end Bit 1: Response error Bit 0: Card detection

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.17 Raw Interrupt Status Register (SDIO_RINTSTS)

Offset address: 0x44

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	SDIO_INTERRUPT	R/W1	0x0	The SDIO card interrupt indicates that the card has generated an interrupt. This interrupt bit is not affected by the interrupt mask register and is always valid. Writing a 1 clears it, and writing a 0 has no effect.
15:0	INT_STATUS	R/W1	0x0	Writing to each bit clears the status bits. Writing a 1 clears the status bit, while writing a 0 keeps it unchanged. Regardless of the interrupt mask status, the bits will be recorded. Bit 15: End-bit error (read) / Write without CRC Bit 14: Auto command completion Bit 13: Start-bit error / Busy clear interrupt Bit 12: Hardware locked write error Bit 11: FIFO underrun / overrun Bit 10: Data missing timeout interrupt Bit 9: Read data timeout Bit 8: Response timeout Bit 7: Data CRC error Bit 6: Response CRC error Bit 5: RX FIFO data request Bit 4: TX FIFO data request Bit 3: Data transfer complete

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				Bit 2: Command end Bit 1: Response error Bit 0: Card detection

33.5.18 Status Register (SDIO_STATUS)

Offset address: 0x48

Reset value: 0x0000 0406

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29:17	FIFO_COUNT	R	0x0	FIFO count
16:11	RESPONSE_INDEX	R	0x0	Response index of the previous command
10	DATA_STATE_ MACHINE_ BUSY	R	0x1	Data state machine is in busy state: 0: Not in busy state 1: In busy state
9	DATA_ BUSY	R	0x1 or 0x0, depending on the value of dat[0]	The inversion of Data[0] indicates whether the card is in the busy state: 0: Not in busy state 1: In busy state
8	DATA_3_STATUS	R	0x1 or 0x0, depending on the value of dat[3]	Can check if the card is inserted into the card reader: 0: No card inserted 1: Card inserted
7:4	RSV	-	-	Reserved
3	FIFO_FULL	R	0x0	FIFO is full.
2	FIFO_EMPTY	R	0x1	FIFO is empty.
1	FIFO_TX_ WATERMARK	R	0x1	FIFO has reached the transmit watermark level; does not meet the data transfer requirements.
0	FIFO_RX_ MATERMARK	R	0x0	FIFO has reached the receive watermark level; does not meet the data transfer requirements.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.19 FIFO Compare Register (SDIO_FIFOTH)

Offset address: 0x4C

Reset value: 0x000F 0000

Bit	Name	Attribute	Reset Value	Description
31	RSV	-	-	Reserved
30:28	MULTI_TRANSACTION_SIZE	R/W	0x0	The burst size during multi-block read/write shall be equal to the source/destination msize of the DMA controller. 000: 1 transfer 001: 4 transfers 010: 8 transfers 011: 16 transfers 100: 32 transfers 101: 64 transfers 110: 128 transfers 111: 256 transfers
27:16	RX_WMARK	R/W	SDIO: 0x0F	This is the FIFO level value when reading data from the card. When the data count in the FIFO is greater than or equal to this value, the DMA/FIFO request signal will be activated. At the end of the entire transfer, regardless of the level value, the DMA/FIFO request signal will be activated to ensure that the remaining data is transferred. In non-DMA mode, when the RXDR interrupt is enabled, an RXDR interrupt will be generated to replace the DMA request. At the end of the transfer, if the level value is greater than the number of remaining data items, no interrupt will be generated. When the data transfer done interrupt is observed, the host shall read the remaining unfinished data.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				In DMA mode, at the end of the transfer, even if the number of remaining data items is less than the level value, the DMA request signal will be activated to perform a single-block transfer, completing all data transfer before the data transfer done interrupt is triggered.
15:12	RSV	-	-	Reserved
11:0	TX_WMARK	R/W	0x0	This is the FIFO level value when writing data to the card. The DMA/FIFO request signal is activated when the data count in the FIFO is less than or equal to this value. At the end of the entire transfer, the DMA/FIFO request signal will be activated regardless of the level value. In non-DMA mode, when the TXDR interrupt is enabled, an TXDR interrupt will be generated to replace the DMA request. At the end of the transfer, the host must fill the remaining incomplete data into the FIFO upon the last interrupt generation (not before the FIFO is full or after the transfer is complete, as the FIFO may not be empty). In DMA mode, at the end of the entire transfer, if the last transfer is less than one burst size, the DMA controller will perform a single cycle until the required amount of data is transferred.

Note: Every time the data is moved into the FIFO, the FIFO data count increases by 1. Every time the data is moved out of the FIFO, the FIFO data count decreases by 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.20 Card Detection Register (SDIO_CDETECT)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CARD_DETECT	R/W	Input of card detection	0: Card detected successfully 1: Card not detected

33.5.21 Write Protection Register (SDIO_WRTPRT)

Offset address: 0x54

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	WRITE_PROTECT	R	Input of write protection	0: No write protection 1: With write protection

33.5.22 TCBCNT Register (SDIO_TCBCNT)

Offset address: 0x5C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	TRANS_CARD_BYTE_COUNT	R	0x0	Number of bytes of data sent from host to card During data transfer, this register reads 0.

33.5.23 TBBCNT Register (SDIO_TBBCNT)

Offset address: 0x60

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	TRANS_FIFO_BYTE_COUNT	R	0x0	Number of bytes transferred between host/DMA and FIFO

33.5.24 DEBOUNCE Register (SDIO_DEBOUNCE)

Offset address: 0x64

Reset value: 0x00FF FFFF

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	-	-	Reserved
23:0	DEBOUNCE_COUNT	R/W	0xFFFFFF	Number of host clocks (clk) used by the debounce filter logic The typical debounce time is 5–25 ms.

33.5.25 Hardware Reset Register (SDIO_RST)

Offset address: 0x78

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:1	RSV	-	-	Reserved
0	CARD_RESET	R/W	1	Hardware reset: 0: Reset 1: Active

33.5.26 Bus Mode Register (SDIO_BMOD)

Offset address: 0x80

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10:8	PBL	R	0x0	Programmable burst length: 000: 1 burst 001: 4 bursts 010: 8 bursts 011: 16 bursts 100: 32 bursts 101: 64 bursts 110: 128 bursts 111: 256 bursts
7	DE	R/W	0x0	IDMA enable: 0: IDMA disabled 1: IDMA enabled
6:2	DSL	R/W	0x0	Descriptor skip length specifying the number of HWords/Words/Dwords to be skipped between two unmarked descriptors (depending on whether it is a 16/32/64-bit bus). This applies only to double-buffer structures.
1	FB	R/W	0x0	Fixed burst transfer controlling whether AHB performs fixed burst transfers: 0: AHB will use single pulse and incremental pulse transfer operations. 1: AHB will use only SINGLE, INCR4, INCR8, or INCR16 during normal burst transfers.
0	SWR	R/W	0x0	Software reset When this bit is set to 1, the DMA controller will reset all internal registers. This bit will be automatically cleared.

33.5.27 Poll Demand Register (SDIO_PLDMND)

Offset address: 0x84

Reset value: 0x0000 0000

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	PD	W	0x0	Polling request If the OWN bit of the descriptor is not set, the FSM transitions to the suspended state. When the host writes any value to this register, the IDMA FSM resumes its normal descriptor fetching operation.

33.5.28 Descriptor List Base Address Register (SDIO_DBADDR)

Offset address: 0x88

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SDL	R/W	0x0	Start of the descriptor list: Contains the base address of the first descriptor.

33.5.29 Internal DMA Status Register (SDIO_IDSTS)

Offset address: 0x8C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12:10	FBE_CODE	R	0x0	Bus error code indicating the type of error that caused a bus error: 3'b001: Host abort received during transfer 3'b010: Host abort received during reception Others: Reserved
9	AIS	R/W1	0x0	Exception interrupt summary, logical OR of the following two bits: IDSTS[2]: Fatal bus interrupt IDSTS[4]: DU bit interrupt Only unmasked bits affect this bit; writing 1 clears it.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
8	NIS	R/W1	0x0	Normal interrupt summary, logical OR of the following two bits: IDSTS[0]: Transfer interrupt IDSTS[1]: Receive interrupt Only unmasked bits affect this bit; writing 1 clears it.
7:6	RSV	-	-	Reserved
5	CES	R/W1	0x0	Card error summary It indicates the status of transactions to/from the card and is also reflected in RINTSTS. Logical OR of the following bits: EBE: End Bit Error RTO: Response Timeout/Boot ACK Timeout RCRC: Response CRC SBE: Start Bit Error DRTO: Data Read Timeout/BDS Timeout DCRC: Data CRC for Receive RE: Response Error Writing 1 clears this bit.
4	DU	R/W1	0x0	Descriptor unavailable interrupt: This bit is set to 1 when the descriptor is unavailable due to OWN bit = 0 (DES0[31] = 0). Writing 1 clears this bit.
3	RSV	-	-	Reserved
2	FBE	R/W1	0x0	Fatal bus error interrupt: This bits indicates that a bus error has occurred (IDSTS[12:10]). When this bit is set to 1, DMA will disable all bus access. Writing 1 clears this bit.
1	RI	R/W1	0x0	Receive interrupt: This bit indicates that data reception for the descriptor is complete. Writing 1 clears it.
0	TI	R/W1	0x0	Transmit interrupt:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				This bit indicates that data transmission for the descriptor is complete. Writing 1 clears it.

33.5.30 Internal DMA Interrupt Enable Register (SDIO_IDINTEN)

Offset address: 0x90

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:10	RSV	-	-	Reserved
9	AI	R/W	0x0	Exception interrupt summary enable This bit enables the following two bits: IDINTEN[2]: Fatal bus error interrupt IDINTEN[4]: DU bit interrupt
8	NIS	R/W	0x0	Normal interrupt summary enable This bit enables the following two bits: IDINTEN[0]: Transmit interrupt IDINTEN[1]: Receive interrupt
7:6	RSV	-	-	Reserved
5	CES	R/W	0x0	Card error summary interrupt enable
4	DU	R/W	0x0	Descriptor unavailable interrupt enable
3	RSV	-	-	Reserved
2	FBE	R/W	0x0	Fatal bus error interrupt enable
1	RI	R/W	0x0	Receive interrupt enable
0	TI	R/W	0x0	Transmit interrupt enable

33.5.31 Current Host Descriptor Address Register (SDIO_DSCADDR)

Offset address: 0x94

Reset value: 0x0000 0000

Unicmicro

Bit	Name	Attribute	Reset Value	Description
31:0	HDA	R	0x0	Host descriptor address pointer: Cleared on reset During operation, the IDMA updates the pointer. This register points to the starting address of the current descriptor being read by the IDMA.

33.5.32 Current Buffer Descriptor Address Register (SDIO_BUFADDR)

Offset address: 0x98

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	HBA	R	0x0	Host buffer address pointer: Cleared on reset During operation, the IDMA updates the pointer. This register points to the current data buffer address being accessed by the IDMA.

33.5.33 Phase Shift Enable Register (SDIO_ENABLESHIFT)

Offset address: 0x110

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1:0	ENABLE_SHIFT	R/W	0x0	Enable control for phase shift on default settings: 00: Default phase shift 01: Enable shift to the next positive edge 10: Enable shift to the next negative edge 11: Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

33.5.34 Data Register (SDIO_DATA)

Offset address: 0x200

Reset value: uncertain

Bit	Name	Attribute	Reset Value	Description
31:0	DATA	R/W	-	Write: Write data to FIFO Read: Read data from FIFO

33.6 Operation Procedure

33.6.1 Initialization

1. Enable SDIO_CTRL[24], which is only applicable to the MMC-Ver3.3 mode.
2. Enable the power supply SDIO_POWEN[0].
3. Write 0xFFFF_FFFF to SDIO_RINTSTS to clear the raw interrupt status.
4. Enable the interrupt SDIO_CTRL[4] and set the interrupt mask register SDIO_INTMASK.
5. Enumerate the card, limiting the clock frequency to 400 kHz according to the SD_MMC_CEATA standard.
6. Set the clock source.
7. Set parameters: response time, data timeout, FIFO threshold.
8. According to the MMC standard, the open-drain pull-up resistor is only required during the enumeration phase. Therefore, for MMC-Ver3.3 mode, clear SDIO_CTRL[24] to disable the open-drain pull-up.

Unicmicro

33.6.2 Card Enumeration

33.6.2.1 MMC_Card

1. Clear the bit width register SDIO_WIDTH and set the card width to 1-bit mode.
2. Set the clock division register to set the frequency to 400 kHz.
3. Send CMD0, an 80-cycle initialization sequence.
4. Send CMD1.
5. Repeat CMD1 until the card busy bit (bit 31) in the received response is set.
6. Send CMD2.
7. Send CMD3 and check the card detection register SDIO_CDETECT.

33.6.2.2 SD_Card

1. Start from port 0.
2. Check if the card is connected.
3. Clear the bit width register SDIO_WIDTH.
4. Identify the card type, whether it is SD, MMC, SDIO, or a combo card:
   A. Send CMD5 with a parameter of 0.
   B. Read the response register (SDIO_RESP0), which provides the voltage values supported by the card.
   C. Send CMD5 again with the voltage value. CMD5 is used to set the voltage and move the card state out of the initialization state.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

D. Check bit 27 of the response. If bit[27] = 1, it indicates that the memory exists and it is a combo card.

E. If it is an SDIO card, jump to step 5; if it is a combo card or no response is received, proceed with the following steps:

F. Send CMD8 with the parameters as follows:

Bit[31:12] = 20'h0

Bit[11:8] = 4'b0001

Bit[7:0] = 8'b10101010

G. If a response is received, the card supports high-capacity SD2.0. Send ACMD41 with the following parameters:

Bit[31] = 1'b0

Bit[30] = 1'b1

Bit[29:25] = 6'h0

Bit[24] = 1'b1

Bit[23:0] = supported voltage range

H. If a response to ACMD41 is received, the card is an SD card; otherwise, it is an MMC or CEATA card.

I. If bit[24] of the ACMD41 response is 1'b1, the host can choose to switch the voltage to 1.8 V, as the card supports 1.8 V. Voltage switching can be performed by sending CMD11.

J. If no response is received to the initial CMD8, the card does not support high-capacity SD2.0. send CMD0 followed by ACMD41 with the following parameters:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit[31] = 1'b0

Bit[30] = 1'b0

Bit[29:24] = 6'h0

Bit[23:0] = supported voltage range

K. If a response to ACMD41 is received, the card is an SD card; otherwise, it is an MMC or CEATA card.

5. Enumerate the card based on the card type.
6. Use a clock source of 400 kHz to perform the following enumeration commands:
   - SD: Send CMD0, CMD8, ACMD41, CMD2, CMD3.
   - SDIO: Send CMD5. If the function count is valid, send CMD3. For the SDIO memory part, perform the same commands as for the SD card.
   - MMC: Send CMD0, CMD1, CMD2, CMD3.
7. Identify MMC/CE-ATA devices.
8. Change the clock frequency after enumeration.

33.6.3 IDMA

33.6.3.1 Initialization

The IDMA initialization process is as follows:

1. Write to the IDMA bus mode register (SDIO_BMOD) to set the host bus access parameters.
2. Write to the IDMA interrupt enable register (SDIO_IDINTEN) to mask unnecessary interrupts.
3. The software driver creates a list of transmit or receive descriptors and then writes it to the IDMA descriptor list base address register (SDIO_DBADDR), providing the starting address

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

of the list to the IDMA.

4. IDMA attempts to fetch descriptors from the descriptor list.

33.6.3.2 Transmit

The IDMA writing data process is as follows:

1. The host sets DES0–DES3 and the OWN bit (DES0[31]).
2. The host writes the write data command to the SDIO_CMD register.
3. The host sets the transmit threshold (SDIO_FIFOTH[11:0]).
4. Based on step 2, IDMA determines that a write data transfer is required.
5. IDMA fetches the descriptor and checks the OWN bit. If the OWN bit is not set, it indicates that the host owns the descriptor. In this case, IDMA enters a suspended state and sets SDIO_IDSTS[4] to 1. In this situation, the host needs to release IDMA by writing any value to the poll demand register (SDIO_PLDMND).
6. Wait for the command done (CD) bit and ensure that there are no errors on the bus interface unit, indicating that the transfer can be completed.
7. IDMA waits for the DMA interface request (dw_DMA_req), which will be generated based on the set transmit threshold. A single transfer will be executed for the last data byte that cannot use burst access.
8. IDMA fetches the transfer data from the data buffer and transfers it to the FIFO.
9. When the data spans multiple descriptors, IDMA will fetch the next descriptor and continue processing the next descriptor. The last descriptor bit in the descriptor indicates whether the data spans multiple descriptors.
10. After the data transfer is complete, if the transfer interrupt is enabled, the status

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

information in the IDSTS register will be updated in the transfer interrupt. Additionally, IDMA clears the OWN bit by performing a write transaction to DES0.

33.6.3.3 Receive

The IDMA reading data process is as follows:

1. The host sets DES0–DES3 and the OWN bit (DES0[31]).
2. The host writes the read data command to the SDIO_CMD register.
3. The host sets the receive threshold (SDIO_FIFOTH[27:16]).
4. Based on step 2, IDMA determines that a read data transfer is required.
5. IDMA fetches the descriptor and checks the OWN bit. If the OWN bit is not set, it indicates that the host owns the descriptor. In this case, IDMA enters a suspended state and sets SDIO_IDSTS[4] to 1. In this situation, the host needs to release IDMA by writing any value to the poll demand register (SDIO_PLDMND).
6. Wait for the command done (CD) bit and ensure that there are no errors on the bus interface unit, indicating that the transfer can be completed.
7. IDMA waits for the DMA interface request (dw_DMA_req), which will be generated based on the set receive threshold. A single transfer will be executed for the last data byte that cannot use burst access.
8. IDMA fetches the data from FIFO and transfers it to the host.
9. When the data spans multiple descriptors, IDMA will fetch the next descriptor and continue processing the next descriptor. The last descriptor bit in the descriptor indicates whether the data spans multiple descriptors.
10. After the data reception is completed, if the receive interrupt is enabled, the status

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

information in the IDSTS register will be updated in the receive interrupt. Additionally, IDMA clears the OWN bit by performing a write transaction to DES0.

33.6.4 SDIO Application Notes

33.6.4.1 Software Programming Requirements

1. Do not change any registers related to a command before the command is completed. These registers include: command register, command argument register, byte count register, block size register, clock divider register, clock enable register, timeout register, and bus width register.
2. Set all parameters related to data transfer before the transfer begins. Do not change these parameters once the transfer has started.
3. When sending a command, set SDIO_CMD[31] to 1. At this point, if SDIO_CMD[21] = 0, all parameters related to command sending (command register, command argument register, byte count register, block size register, bus width register, timeout register, clock enable register, clock divider register) will be updated. This update requires a certain amount of time (6 system clocks), during which command-related parameters shall not be changed; otherwise, a register overwrite lock error may occur. Once the command is accepted by the card, the next command can be sent, but there are the following scenarios:
   - If the previous command is not a data transfer command, the new command will be sent after the previous command is completed.
   - If the previous command is a data transfer command and SDIO_CMD[13] is set, the new command will be sent after the data transfer is completed.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- If SDIO_CMD[13] is 0, the new command will be sent after the previous command is completed. For example, if you want to stop data transfer or query the card status during data transfer.

4. Commands will not be sent during data transfer.
5. Only one card can be read from or written to at a time, and commands can only be sent to one card at a time. For instance, while transferring data with one card, you cannot send commands to another card simultaneously.
6. Only one command can be sent to the same card at a time.
7. In a write operation with a byte count register set to 0, if the clock stops due to an empty FIFO, first the FIFO shall be filled with data to restart the clock before sending a stop command.
8. If a reset command (CMD0, CMD15, CMD52_reset) needs to be sent during data transfer, the SDIO_CMD[14] bit in the command register must be set, which will stop the data transfer after the command is sent.
9. If the clock stops due to a full FIFO during a read operation, the software shall read at least two FIFOs to restart the clock.
10. If the card is paused or stopped during transfer, the FIFO shall be reset when resuming.

Before sending a new command, ensure that the card is not in a busy state. Before changing the output clock frequency, the software must ensure that there are no ongoing data or command transfers.

To avoid glitches in the output clock, the following steps shall be taken before changing the card clock:

1. Disable the clock enable register and update with update_clk. To ensure that the previous

Unicmicro

command has been completed, set the following bits: SDIO_CMD[31], SDIO_CMD[21], and SDIO_CMD[13].

2. Write to the clock divider register and set SDIO_CMD[31].

3. Enable the clock and then set SDIO_CMD[31].

33.6.4.2 Read Operation Process

1. Set the byte count register.

2. Set the block size register.

3. Initialize the command.

4. Set the command argument register.

5. Send CMD17/18.

33.6.4.3 Write Operation Process

1. Set the byte count register.

2. Set the block size register.

3. Initialize the command.

4. Set the command argument register.

5. Send CMD24/25.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34 Ethernet Media Access Controller (EMAC)

34.1 Overview

The EMAC can receive and transmit Ethernet data, complying with the IEEE 802.3-2002 standard.

34.2 Main Features

- Supports MII, RMII, and RGMII interfaces
- Supports external PHY interfaces to achieve data transmission rates of 10M/100M/1000M bit/s
- Full-duplex and half-duplex operation modes
- Configure and manage PHY devices using MDIO interface
- Supports Ethernet timestamping (IEEE 1588–2002), providing a 64-bit timestamp for each frame during transmission or reception.
- Header and start frame delimiter (SFD) being inserted in the transmit path and removed in the receive path
- Validity detection of frame length supported for discarding frames that are too long or too short
- Supports CRC check of incoming frames for discarding frames with errors
- Supports CRC check to outgoing frames
- Short frame padding
- Statistical counting of received and transmitted frames

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Filtering of broadcast, multicast and unicast frames
- Configurable rate-limiting for control packets, IP packets, and broadcast or multicast packets
- Packet filtering
- Two types of interrupts: queue interrupt and timeout interrupt
- Packet buffering for transmission and reception
- Energy-efficient Ethernet (EEE)
- Two independent 2-Kbyte FIFOs for transmission and reception respectively

34.3 Pin Description

Table 34-1: EMAC Pin Description

Function Pin	RGMII	RMII	MII	Alternate Function Pin	Direction	Functional Description
ETH_GTXCLK/ ETH_TXCLK	ETH_GTXCLK	-	ETH_TXCLK	PC3	Input/ Output	RGMII: Channel transmission clock output at 125M MII: Channel transmission clock input
ETH_GTXCLK/ ETH_TXCLK	ETH_RXCLK	ETH_RXCLK/ ETH_REF_50M	ETH_RXCLK	PA1	Input	Channel reception clock at 10M/100M/1000M, RGMII/RMII/MII, RMII (50M)
ETH_CLK125M (REF_50M_125M)	ETH_CLK125M	-	-	PD2	Input	Channel reference clock at 125M, RGMII
ETH_TXD0	ETH_TXD0	ETH_TXD0	ETH_TXD0	PB12	Output	Channel transmit data 0
ETH_TXD1	ETH_TXD1	ETH_TXD1	ETH_TXD1	PB13	Output	Channel transmit data 1
ETH_TXD2	ETH_TXD2	-	ETH_TXD2	PC2	Output	Channel transmit data 2
ETH_TXD3	ETH_TXD3	-	ETH_TXD3	PB8, PE2	Output	Channel transmit data 3
ETH_TXEN	ETH_TXEN	ETH_TXEN	ETH_TXEN	PB11	Output	Channel transmit data enable
ETH_RXD0	ETH_RXD0	ETH_RXD0	ETH_RXD0	PC4	Input	Channel receive data 0
ETH_RXD1	ETH_RXD1	ETH_RXD1	ETH_RXD1	PC5	Input	Channel receive data 1
ETH_RXD2	ETH_RXD2	-	ETH_RXD2	PB0	Input	Channel receive data 2
ETH_RXD3	ETH_RXD3	-	ETH_RXD3	PB1	Input	Channel receive data 3
ETH_RXDV	ETH_RXDV	ETH_RXDV	ETH_RXDV	PA7	Input	Channel receive data valid
ETH_MDIO	ETH_MDIO	ETH_MDIO	ETH_MDIO	PA2	Input/ Output	Channel serial data
ETH_MDC	ETH_MDC	ETH_MDC	ETH_MDC	PC1	Output	Channel serial clock output
ETH_CRS	-	-	-	PA0	Input	-
ETH_COL	-	-	-	PA3	Input	-
ETH_RX_ER	-	-	-	PB10	Input	-
ETH_TX_ER	-	-	-	PD6	Output	-

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.4 List of EMAC Registers

EMAC register base address: 0x4010_0000

EMAC registers are listed below:

Table 34-2: List of EMAC Registers

Address Offset	Register Name	Description
0x00	EMAC_CONFIG	EMAC configuration register
0x04	EMAC_FRAMEFILTER	Frame filter register
0x08	EMAC_HASHTABLEHIGH	Hash table high register
0x0C	EMAC_HASHTABLELOW	Hash table low register
0x10	EMAC_GMIIADDRESS	GMII address register
0x14	EMAC_GMIIDATA	GMII data register
0x18	EMAC_FLOWCONTROL	Flow control register
0x1C	EMAC_VLANTAG	VLAN tag register
0x24	EMAC_DEBUGER	Debug register
0x30	EMAC_LPICS	LPI control status register
0x34	EMAC_LPIT	LPI timer register
0x38	EMAC_INTSTATUS	Interrupt status register
0x3C	EMAC_INTMASK	Interrupt mask register
0x40	EMAC_ADDR0H	EMAC address 0 high register
0x44	EMAC_ADDR0L	EMAC address 0 low register
0x48	EMAC_ADDR1H	EMAC address 1 high register
0x4C	EMAC_ADDR1L	EMAC address 1 low register
0x50	EMAC_ADDR2H	EMAC address 2 high register
0x54	EMAC_ADDR2L	EMAC address 2 low register
0x58	EMAC_ADDR3H	EMAC address 3 high register
0x5C	EMAC_ADDR3L	EMAC address 3 low register
0x60	EMAC_ADDR4H	EMAC address 4 high register
0x64	EMAC_ADDR4L	EMAC address 4 low register
0x68	EMAC_ADDR5H	EMAC address 5 high register
0x6C	EMAC_ADDR5L	EMAC address 5 low register
0x70	EMAC_ADDR6H	EMAC address 6 high register
0x74	EMAC_ADDR6L	EMAC address 6 low register
0x78	EMAC_ADDR7H	EMAC address 7 high register
0x7C	EMAC_ADDR7L	EMAC address 7 low register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x80	EMAC_ADDR8H	EMAC address 8 high register
0x84	EMAC_ADDR8L	EMAC address 8 low register
0x88	EMAC_ADDR9H	EMAC address 9 high register
0x8C	EMAC_ADDR9L	EMAC address 9 low register
0x90	EMAC_ADDR10H	EMAC address 10 high register
0x94	EMAC_ADDR10L	EMAC address 10 low register
0x98	EMAC_ADDR11H	EMAC address 11 high register
0x9C	EMAC_ADDR11L	EMAC address 11 low register
0xA0	EMAC_ADDR12H	EMAC address 12 high register
0xA4	EMAC_ADDR12L	EMAC address 12 low register
0xA8	EMAC_ADDR13H	EMAC address 13 high register
0xAC	EMAC_ADDR13L	EMAC address 13 low register
0xB0	EMAC_ADDR14H	EMAC address 14 high register
0xB4	EMAC_ADDR14L	EMAC address 14 low register
0xB8	EMAC_ADDR15H	EMAC address 15 high register
0xBC	EMAC_ADDR15L	EMAC address 15 low register
0xD8	EMAC_RGMIICS	RGMII control status register
0xDC	EMAC_WDG	Timeout watchdog register

34.4.1 EMAC Configuration Register (EMAC_CONFIG)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	-
27	TKPE	R/W	0	When bit 20 (JE) is 0 and this bit is 1, the EMAC will classify all frames exceeding 2,000 bytes as jumbo frames and those less than 2,000 bytes as normal frames. When both bit 20 (JE) and this bit are 0, the EMAC will classify all frames exceeding 1,518 bytes as jumbo frames.
26:25	RSV	-	-	-

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
24	TC	R/W	0	1: Activate the duplex mode, connection speed, and uplink/downlink information of RGMII PHY. 0: Do not transmit this information to the PHY.
23	WDGPD	R/W	0	1: Disable the receiver watchdog, allowing reception of frames up to 16,383 bytes. 0: Enable the watchdog, allowing only the reception of data up to 2,048 bytes (10,240 bytes when JE is enabled).
22	JPD	R/W	0	1: Disable the Jabber timer, allowing the transmission of data up to 16,383 bytes. 0: Enable the Jabber timer, allowing only the transmission of data up to 2,048 bytes (10,240 bytes when JE is enabled).
21	BURST	R/W	0	1: Reserved. 0: Do not enable.
20	JE	R/W	0	1: Allow reception of frames up to 9,018 bytes without triggering a giant frame error. 0: Do not enable.
19:17	IFG	R/W	0	Set the inter-frame interval: 000: 96-bit time 001: 88-bit time 010: 80-bit time …… 111: 40-bit time At least 64-bit time is required in half-duplex mode (set to 100).
16	CSPD	R/W	0	1: In half-duplex mode, ignore the MII CRS signal; no error reported on carrier loss. 0: Enable carrier detection.
15:14	SPEED	R/W	0	Select speed: 00/01: 1,000 Mbps 10: 10 Mbps 11: 100 Mbps

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13	RCPD	R/W	0	1: Do not receive frames while transmitting in half-duplex mode. 0: Receive frames while transmitting in half-duplex mode.
12	LB	R/W	0	1: Enable self-loop mode in MII, requiring RX clock input. 0: Do not enable.
11	DP	R/W	0	1: Full duplex 0: Half duplex
10	CSE	R/W	0	1: Enable TCP/UDP/ICMP header and payload checksum. 0: Disable.
9	RTPD	R/W	0	1: Disable retry; the transmission will be abandoned and an error will be reported when a conflict occurs at the MII interface. 0: Enable retry.
8	RSV	-	-	Reserved
7	ACS	R/W	0	1: Automatically remove padding bytes and CRC field when receiving frames less than 1,536 bytes. 0: Disable.
6:5	BL	R/W	0	Set the waiting time before retrying after a transmission collision in half-duplex mode: 00: k = min (n, 10) 01: k = min (n, 8) 10: k = min (n, 4) 11: k = min (n, 1) n = number of retries The random waiting time is initially selected from the range [0, 2^k], then multiplied by 4096-bit time at 1000 Mbps, or by 512-bit time at 10/100 Mbps.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	DC	R/W	0	1: In half-duplex mode, if the frame transmission is delayed by 24,288 (in 10/100 Mbps mode) or 155,680 (in 1000 Mbps mode or when JE is enabled) bit times, the transmission will be aborted and a timeout error will be generated. 0: Keep delaying until the CRS signal is invalid, then proceed with the transmission.
3	TE	R/W	0	1: Enable transmission. 0: Disable transmission; frames that are already being transmitted will be completed.
2	RE	R/W	0	1: Enable reception. 0: Disable reception; frames that are already being received will be completed.
1:0	PL	R/W	0	Preamble length: 00: 7 bytes 01: 5 bytes 10: 3 bytes

34.4.2 Frame Filter Register (EMAC_FRAMEFILTER)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	RA	R/W	0	1: Receive all frames, ignoring filter settings. 0: Receive only frames that meet the filter conditions, according to the filter settings.
30:22	RSV	-	-	-
21	DNTU	R/W	0	1: Filter IP out-of-bounds frames that are not TCP or UDP. 0: Do not filter.
20	IPFE	R/W	0	1: Filter frames specified by the filters on line 3 and line 4. 0: Do not filter.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19:17	RSV	-	-	-
16	VTFE	R/W	0	1: Filter frames that cannot successfully compare VLAN tags. 0: Do not filter.
15:11	RSV	-	-	-
10	HPF	R/W	0	1: Determine the filtering conditions based on HMC and HUC settings. 0: If HMC or HUC is 1, frames that meet the HASH filter will pass.
9	SAF	R/W	0	1: Filter frames whose source address does not meet the settings. 0: Do not filter.
8	SAIF	R/W	0	1: Source address filtering condition being inverted 0: Source address filtering condition being normal
7:6	PCF	R/W	0	Configure control frame transmission: 00: Filter all control frames. 01: Filter pause frames. 10: Transmit all control frames. 11: Filter control frames that do not meet the address filter.
5	DBF	R/W	0	1: Filter broadcast frames. 0: Do not filter.
4	PM	R/W	0	1: Do not filter multicast frames (frames with the first address bit as 1). 0: Multicast frames are affected by HMC.
3	DAIF	R/W	0	1: Destination address filtering condition being inverted 0: Destination address filtering condition being normal
2	HMC	R/W	0	1: Filter destination addresses based on HASH list when receiving multicast frames. 0: Perfectly filter destination addresses when receiving multicast frames (compare with the destination address register).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	HUC	R/W	0	1: Filter destination addresses based on HASH list when receiving unicast frames. 0: Perfectly filter destination addresses when receiving unicast frames (compare with the destination address register).
0	PR	R/W	0	1: Disable address filtering. 0: Enabled address filtering.

34.4.3 Hash Table High Register (EMAC_HASHTABLEHIGH)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	HTH	R/W	0	During HASH filtering, the destination address in the frame is sent to the CRC logic. Subsequently, the high 6 bits of the CRC register are used to specify one of the 64 bits in the HASH table. If that bit in the HASH table is set to 1, the frame can be filtered.

34.4.4 Hash Table Low Register (EMAC_HASHTABLELOW)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	HTL	R/W	0	During HASH filtering, the destination address in the frame is sent to the CRC logic. Subsequently, the high 6 bits of the CRC register are used to specify one of the 64 bits in the HASH table. If that bit in the HASH table is set to 1, the frame can be filtered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.4.5 GMII Address Register (EMAC_GMIIADDRESS)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	-
15:11	PA	R/W	0	Address of the PHY being accessed
10:6	GR	R/W	0	Used to select the register within the PHY
5:2	CR	R/W	0	Select the MDC clock frequency: 0000: MDCCLK = HCLK/42, used when HCLK is between 60–100 MHz 0001: MDCCLK = HCLK/62, used when HCLK is between 100–150 MHz 0010: MDCCLK = HCLK/16, used when HCLK is between 20–35 MHz 0011: MDCCLK = HCLK/26, used when HCLK is between 35–60 MHz 0100: MDCCLK = HCLK/102, used when HCLK is between 150–250 MHz 0101: MDCCLK = HCLK/124, used when HCLK is between 250–300 MHz The following settings will result in an excessively high MDC clock frequency. It is necessary to confirm whether the device can support them: 1000: MDCCLK = HCLK/4 1001: MDCCLK = HCLK/6 1010: MDCCLK = HCLK/8 1011: MDCCLK = HCLK/10 1100: MDCCLK = HCLK/12 1101: MDCCLK = HCLK/14 1110: MDCCLK = HCLK/16 1111: MDCCLK = HCLK/18
1	WR	R/W	0	1: Write to the PHY. 0: Read from the PHY.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	BUSY	R	0	This indicates that an operation is being performed on the PHY register, during which the values in the GMII data register shall not be changed.

34.4.6 GMII Data Register (EMAC_GMIIDATA)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	-
15:0	DATA	R/W	0	PHY register reads/writes data

34.4.7 Flow Control Register (EMAC_FLOWCONTROL)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	PT	R/W	0	Set the value of the PAUSE time field in the control frame.
15:8	RSV	-	-	-
7	DZPQ	R/W	0	1: When the FIFO layer flow control signal is released, no zero-value pause frame will be automatically generated. 0: A zero-value pause frame will be automatically generated.
6	RSV	-	-	-
5:4	PLT	R/W	0	Set the pause low threshold, which must be less than the PT value: 00: 4 01: 28 10: 144 11: 256

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				The unit is the time required to transmit 64 bytes over the interface.
3	UP	R/W	0	1: The EMAC will detect pause frames with both unique multicast addresses and unicast addresses. 0: It will detect only pause frames with unique multicast addresses.
2	RFE	R/W	0	1: The EMAC will parse pause frames and send for a period of time. 0: It will not parse pause frames.
1	TFE	R/W	0	**In full-duplex mode:** 1: Enable flow control, allowing the transmission of pause frames. 0: Disable flow control, no pause frames will be sent. **In half-duplex mode:** 1: Enable backpressure functionality. 0: Disable backpressure functionality.
0	FCBBPA	R/W	0	**In full-duplex mode:** Writing 1 to this bit will send a pause frame, which will be automatically cleared. Do not operate this register if it has not been cleared. **In half-duplex mode:** When this bit and TFE are both set to 1, the EMAC will enable backpressure functionality, sending a blocking signal upon receiving a frame to create a line collision.

34.4.8 VLAN Tag Register (EMAC_VLANTAG)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	-

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19	VTHE	R/W	0	1: Enable the VLAN tag HASH table matching function. 0: Do not enable.
18	ESVL	R/W	0	1: Recognize frames of S-VLAN (88A8) type as VLAN tag frames. 0: Do not use S-VLAN.
17	VTIM	R/W	0	1: VLAN tag matching condition being inverted. 0: VLAN tag matching condition being normal.
16	ETV	R/W	0	1: Compare using 12-bit tags. 0: Compare using 16-bit tags.
15:0	VLT	R/W	0	VLAN tag identifier: [15:13]: User priority [12]: CFI/DEI [11:0]: VID If all bits of VLT are 0, VLAN tags will not be compared, and all 8100 and 88A8 type frames will be recognized as VLAN frames.

34.4.9 Debug Register (EMAC_DEBUGER)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25	TXSTSFSTS	R	0	MTL TX FIFO is full.
24	TXDSTS	R	0	MTL TX FIFO is not empty.
23	RSV	-	-	-
22	TWCSTS	R	0	The MTL TX FIFO write controller is working.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
21:20	TRCSTS	R	0	Status of the MTL TX FIFO read controller: 00: Idle 01: Reading 10: Waiting for EMAC transmit request 11: Writing or clearing
19	TXPAUSED	R	0	The EMAC transmission is currently paused.
18:17	TFCSTS	R	0	Status of the EMAC transmit controller: 00: Idle 01: Waiting for previous frame to complete 10: Generating or sending a pause frame 11: Fetching the frame to be sent
16	TPESTS	R	0	The EMAC MII transmit protocol engine is working.
15:10	RSV	-	-	-
9:8	RXFSTS	R	0	Status of MTL RX FIFO: 00: Empty 01: Below non-activated threshold 10: Above activated threshold 11: Full
7	RSV	-	-	-
6:5	RRCSTS	R	0	Status of the MTL RX FIFO read controller: 00: Idle 01: Reading frame data 10: Reading frame status or timestamp 11: Clearing
4	RWCSTS	R	0	The MTL RX FIFO controller is working.
3	RSV	-	-	Reserved
2	RFCFCSTRS	R	0	The receive frame FIFO read controller is working.
1	RFCFCSTWS	R	0	The receive frame FIFO write controller is working.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	RPESTS	R	0	The EMAC MII receive protocol engine is working.

34.4.10 LPI Control Status Register (EMAC_LPICS)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:20	RSV	-	-	-
19	LPITXA	R/W	0	1: EMAC enters LPI mode affected by LPIEN only after all frames have been transmitted. 0: EMAC enters LPI mode directly affected by LPIEN.
18	PLSEN	R/W	0	1: LPI triggering is affected by the RGMII control status register and PLS. 0: LPI triggering is only affected by PLS.
17	PLS	R/W	0	1: Recognize the connection as normal. 0: Recognize the connection as disconnected.
16	LPIEN	R/W	0	1: Put EMAC into LPI mode. 0: EMAC is in normal mode. If LPITXA is 1, this bit will be automatically cleared when a new transmit operation begins.
15:10	RSV	-	-	-
9	RLPIST	R	0	EMAC is receiving an LPI signal from the MII interface.
8	TLPIST	R	0	EMAC is sending an LPI signal to the MII interface.
7:4	RSV	-	-	-
3	RLPIEX	R	0	This bit indicates that the LPI signal is no longer received from MII and the LPI mode is exited, cleared after reading this register.
2	RLPIEN	R	0	This bit indicates that an LPI signal is received from MII and the LPI mode is entered, cleared after reading this register.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	TLPIEX	R	0	This bit indicates that the EMAC transmitter has exited LPI mode due to the LPIEN bit setting, cleared after reading this register.
0	TLPIEN	R	0	This bit indicates that the EMAC transmitter has entered LPI mode due to the LPIEN bit setting, cleared after reading this register.

34.4.11 LPI Timer Register (EMAC_LPIT)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25:16	LST	R/W	0x3E8	Set the waiting time before EMAC sends the LPI signal to PHY after meeting the conditions.
15:0	TWT	R/W	0	Set the waiting time after EMAC exits the LPI mode before performing the next transmission.

34.4.12 Interrupt Status Register (EMAC_INTSTATUS)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	-
10	LPIIS	R	0	LPI mode entry and exit interrupt status, cleared by reading the LPICS register.
9	TSIS	R	0	Timestamp interrupt status, triggered when the system time exceeds the target high/low time threshold, cleared by reading this register.
8	RSV	-	-	-

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7	MMCRIPIS	R	0	MMC receive parity interrupt status, reflecting the interrupt status in the MMCRCI register, cleared by reading the MMCRCI register.
6	MMCTIS	R	0	MMC transmit interrupt status, reflecting the interrupt status in the MMCTI register, cleared by reading the MMCTI register.
5	MMCRIS	R	0	MMC receive interrupt status, reflecting the interrupt status in the MMRTI register, cleared by reading the MMRTI register.
4	MMCIS	R	0	MMC interrupt status, reflecting the interrupt status in bits [7:5].
3:1	RSV	-	-	-
0	RGMIIIS	R	0	RGMII interrupt status, triggered when the RGMII connection status changes, cleared by reading the RGMIICS register.

34.4.13 Interrupt Mask Register (EMAC_INTMASK)

Offset address: 0x3C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	-
10	LPIIM	R/W	0	1: LPI mode entry and exit interrupts will not be triggered. 0: Interrupts will be triggered.
9	TSIM	R/W	0	1: Timestamp interrupts will not be triggered. 0: Interrupts will be triggered.
8:1	RSV	-	-	-
0	RGMIIIM	R/W	0	1: RGMII interrupts will not be triggered. 0: Interrupts will be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.4.14 EMAC Address Register High 0–15 (EMAC_ADDR0–15H)

Offset addresses: 0x40, 0x48, …, 0xB8, x = 0–15

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	0x8000	-
15:0	AxH	R/W	0	High 16 bits of EMAC address

34.4.15 EMAC Address Register Low 0–15 (EMAC_ADDR0–15L)

Offset addresses: 0x44, 0x4C, ..., 0xBC, x = 0–15

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	AxL	R/W	0	Low 32 bits of EMAC address

34.4.16 RGMII Control Status Register (EMAC_RGMIICS)

Offset address: 0xD8

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	-
3	LSTATUS	R	0	Connection status between MAC and PHY
2:1	LSPEED	R	0	Connection speed: 00: 2.5 MHz 01: 25 MHz 10: 125 MHz
0	LM	R	0	Connection mode: 1: Full duplex 0: Half duplex

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.4.17 Timeout Watchdog Register (EMAC_WDG)

Offset address: 0xDC

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	-
16	WDGEN	R/W	0	If this bit is 1 and the WDGPD bit (bit 23) in the EMAC Config is 0, then the timing in this register will be used to control the determination of timeout frames.
15:14	RSV	-	-	-
13:0	TOT	R/W	0	Set the timeout value (which shall be at least greater than 1522), timeout frames will be considered as error frames.

34.5 List of PTP Registers

EMAC register base address: 0x4010_0000

The PTP registers are listed below:

Table 34-3: List of PTP Registers

Address Offset	Register Name	Description
0x700	PTP_TSCTL	Timestamp control register
0x704	PTP_SUBSECINC	Timestamp sub-second increment register
0x708	PTP_SEC	Timestamp seconds register
0x70C	PTP_NANOSEC	Timestamp nanoseconds register
0x710	PTP_SECUD	Timestamp seconds update register
0x714	PTP_NANOSECUD	Timestamp nanoseconds update register
0x718	PTP_TSADDEND	Timestamp addend register
0x71C	PTP_TGTSEC	Timestamp target seconds register
0x720	PTP_TGTNANOSEC	Timestamp target nanoseconds register

34.5.1 Timestamp Control Register (PTP_TSCTL)

Offset address: 0x700

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 2000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	-
18	ADDRFLT	R/W	0	1: Filter addresses when transmitting PTP frames. 0: Do not filter.
17:16	SNAPTPYE	R/W	0	Select PTP packets that require timestamp snapshots. Refer to Table 34-3.
15	MSTSNAP	R/W	0	1: Use snapshot for messages related to the master node. 0: Do not use.
14	EVENTSNAP	R/W	0	1: Use snapshot for messages related to event. 0: Do not use.
13	TSIPV4EN	R/W	1	1: The receiver processes PTP packets encapsulated in IPV4-UDP. 0: Ignore PTP packets encapsulated in IPV4-UDP.
12	TSIPV6EN	R/W	0	1: The receiver processes PTP packets encapsulated in IPV6-UDP. 0: Ignore PTP packets encapsulated in IPV6-UDP.
11	TSIPEN	R/W	0	1: The receiver processes directly transmitted PTP packets. 0: Ignore directly transmitted PTP packets.
10	TSV2EN	R/W	0	1: Process packets using the IEEE 1588 V2 format. 0: Process packets using the IEEE 1588 V1 format.
9	TSDBRC	R/W	0	1: The timestamp low register rolls over after reaching 0x3B9AC9FF and increments the timestamp high seconds. 0: The timestamp low register rolls over after reaching 0x7FFFFFFF and

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				increments the timestamp high seconds. The sub-second increment must be set according to the PTP reference clock frequency and the value of this bit.
8	TSENALL	R/W	0	1: Enable timestamp snapshot for all received frames. 0: Do not enable.
7:6	RSV	-	-	-
5	ADDENDUD	R/W	0	Writing 1 to this bit updates the content of the TSADDEND register. The bit is automatically cleared after the update is complete. Ensure that this bit is 0 before writing 1.
4	INTRTRIG	R/W	0	Writing 1 to this bit enables the interrupt, which is triggered when the system time exceeds the target seconds. The bit is automatically cleared after the interrupt is triggered.
3	TSUD	R/W	0	Writing 1 to this bit adds the contents of the SECUD and NANOSECUD registers to the SEC and NANOSEC registers. The bit is automatically cleared after the update is complete. Ensure that this bit is 0 before writing 1.
2	TSINIT	R/W	0	Writing 1 to this bit copies the contents of the SECUD and NANOSECUD registers to the SEC and NANOSEC registers. The bit is automatically cleared after the update is complete. Ensure that this bit is 0 before writing 1.
1	TSFUD	R/W	0	1: Timestamp fine update 0: Timestamp coarse update
0	TSEN	R/W	0	1: Enable timestamp; initialization is required after enabling.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Disable timestamp; the receiver does not process IEEE 1588 frames.

Table 34-3 : Timestamp Snapshot Relationship Table

SNAPTPYE (Bits 17:16)	MSTSNAP (Bit 15)	EVENTSNAP (Bit 14)	PTP Information
00	X	0	SYNC, Follow_Up, Delay_Req, Delay_Resp
00	0	1	SYNC
00	1	1	Delay_Req
01	X	0	SYNC, Follow_Up, Delay_Req, Delay_Resp, Pdelay_Req, Pdelay_Resp, Pdelay_Resp_Follow_Up
01	0	1	SYNC, Pdelay_Req, Pdelay_Resp
01	1	1	Delay_Req, Pdelay_Req, Pdelay_Resp
10	X	X	SYNC, Delay_Req
11	X	X	Pdelay_Req, Pdelay_Resp

34.5.2 Timestamp Sub-second Increment Register (PTP_SUBSECINC)

Offset address: 0x704

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	-
7:0	SSINC	R/W	0	Sub-second increment value The set value of this bit field will increment with the value of the sub-second register during each clock cycle (clk_ptp_i). For example, if the PTP clock operates at 50 MHz (with a period of 20 ns), and the system time nanoseconds register has a precision of 1 ns (when PTP_TSCTL[9] is set), these bits shall be set to 20 (0x14). When PTP_TSCTL[9] is 0, the

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				nanoseconds register has a precision of approximately 0.465 ns. In this case, these bits shall be set to 43 (0x2B), which is calculated as 20 ns / 0.465.

34.5.3 Timestamp Seconds Register (PTP_SEC)

Offset address: 0x708

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SEC	RO	0	Display system time in seconds.

34.5.4 Timestamp Nanoseconds Register (PTP_NANOSEC)

Offset address: 0x70C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	RSV	-	-	-
30:0	NANOSEC	RO	0	Display the sub-second time with a precision of 0.46 ns. When PTP_TSCTL[9] is set to 1, the precision is 1 ns, and the value must not exceed 0x3B9AC9FF.

34.5.5 Timestamp Seconds Update Register (PTP_SECUD)

Offset address: 0x710

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SECUD	R/W	0	Set the number of seconds to be copied or added to system time.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.5.6 Timestamp Nanoseconds Update Register (PTP_NANOSECUD)

Offset address: 0x714

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	SIGN	R/W	0	1: When updating the time, the time value is subtracted by the value in the update register. 0: When updating the time, the time value is added by the value in the update register.
30:0	NANOSECUD	R/W	0	Set the number of subseconds to be copied or added to system time.

34.5.7 Timestamp Addend Register (PTP_TSADDEND)

Offset address: 0x718

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	ADDEND	R/W	0	This value is used for synchronizing real-time and is only useful during fine updates of the timestamp, where the value is added to a 32-bit counter in each cycle (clk_ptp_ref_i). When this counter overflows, the system time will be updated.

34.5.8 Timestamp Target Seconds Register (PTP_TGTSEC)

Offset address: 0x71C

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:0	TGTSEC	R/W	0	Target seconds for interrupt generation, with the interrupt status displayed in bit 9 of the EMAC_INTSTATUS register.

34.5.9 Timestamp Target Nanoseconds Register (PTP_TGTNANOSEC)

Offset address: 0x720

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	RSV	-	-	-
31:0	TGTNSEC	R/W	0	Target subseconds for interrupt generation, with the interrupt status displayed in bit 9 of the EMAC_INTSTATUS register. The precision is 0.46 ns. When PTP_TSCTL[9] is set to 1, the precision is 1 ns, and the value must not exceed 0x3B9AC9FF.

34.6 List of DMA Registers

EMAC register base address: 0x4010_0000

The registers are listed below:

Table 34-4: List of DMA Registers

Address Offset	Register Name	Description
0x1000	DMA_BUSMODE	Bus mode register
0x1004	DMA_TPD	Transmit register
0x1008	DMA_RPD	Receive register
0x100C	DMA_RLA	Receive descriptor address register
0x1010	DMA_TLA	Transmit descriptor address register
0x1014	DMA_STATUS	DMA status register
0x1018	DMA_OPMODE	DMA operation mode register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x101C	DMA_INTEN	DMA interrupt enable register
0x1020	DMA_FMBOCNT	DMA lost frame and buffer overflow counter register
0x1024	DMA_RIWDT	Receive interrupt watchdog timer register
0x1048	DMA_CTD	DMA current transmit descriptor register
0x104C	DMA_CRD	DMA current receive descriptor register
0x1050	DMA_CTB	DMA current transmit buffer address register
0x1054	DMA_CRB	DMA current receive buffer address register

34.6.1 Bus Mode Register (DMA_BUSMODE)

Offset address: 0x1000

Reset value: 0x0002 0100

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25	ALIGN	R/W	0	Address alignment 1: If FB is 1, align all transfers; if FB is 0, the first burst transfer (starting address) is not aligned, but subsequent burst transfers are aligned.
24	PBL8	R/W	0	1: Multiply the values in bits [22:17] and [13:8] by 8.
23	SEPPBL	R/W	0	1: Bits [22:17] control reception, and bits [13:8] control transmission. 0: Bits [13:8] control both transmission and reception.
22:17	RXPBL	R/W	1	Set the maximum number of transfers for a single DMA transfer; the value can only be selected from 1, 2, 4, 8, 16, or 32.
16	FB	R/W	0	1: Use SINGLE, INCR4, INCR8, and INCR16 during transfers. 0: Use SINGLE and INCR during transfers.
15:14	PE	R/W	0	Specify the priority ratio for reception and transmission: 00: 1:1

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				01: 2:1 10: 3:1 11: 4:1
13:8	TXPBL	R/W	1	Set the maximum number of transfers for a single DMA transfer; the value can only be selected from 1, 2, 4, 8, 16, or 32.
7	RSV	-	-	-
6:2	DSL	R/W	0	Set the distance between descriptors that are not connected in a chain structure, in units of 32-bit words.
1	DA	R/W	0	DMA arbitration mode: 1: DMA always prioritizes receive transfers. 0: Transfers are conducted in a cyclic manner according to the ratio set in [15:14] for reception and transmission.
0	RESET	R/W	0	Writing 1 to this bit will reset the entire EMAC. This bit automatically clears, and do not write to it again before it clears.

34.6.2 Transmit Register (DMA_TPD)

Offset address: 0x1004

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	TPD	W	0	Writing any value to this bit can command the DMA to check the current descriptor pointed to by the CHTD register. If it is not available, the DMA will enter a pause state and set bit 2 of the status register to 1; if it is available, the transfer will continue, which can be used to resume transfers that are in a paused state. Transmit transfers are usually paused due to data underflow or the inability of DMA to obtain the required descriptors.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.6.3 Receive Register (DMA_RPD)

Offset address: 0x1008

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RPD	W	0	Writing any value to this bit can command the DMA to check the current descriptor pointed to by the CHRD register. If it is not available, the DMA will enter a pause state and set DMA_STATUS[7] to 1; if it is available, the transfer will continue, which can be used to resume transfers that are in a paused state. Receive transfers are usually paused due to the inability of DMA to obtain the required descriptors.

34.6.4 Receive Descriptor Address Register (DMA_RLA)

Offset address: 0x100C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	RLA	R/W	0	Set the base address of the receive queue, aligned to a 32-bit address.

34.6.5 Transmit Descriptor Address Register (DMA_TLA)

Offset address: 0x1010

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	TLA	R/W	0	Set the base address of the transmit queue, aligned to a 32-bit address.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.6.6 DMA Status Register (DMA_STATUS)

Offset address: 0x1014

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	-
29	TSTIS	R	0	Timestamp interrupt status, cleared by reading the EMAC_INTSTATUS register.
28	RSV	-	-	-
27	MMCIS	R	0	MMC interrupt status, cleared in the MMC-related registers.
26	RSV	-	-	-
25:23	EB	R	0	Error types, valid only when bit 13 is 1: 000: Error writing received data 011: Error reading transmit data 100: Error writing receive descriptor 101: Error writing transmit descriptor 110: Error reading receive descriptor 111: Error reading transmit descriptor
22:20	TPS	R	0	Transmit status: 000: Stopped; after reset or upon receiving a stop command 001: Reading transmit descriptor 010: Waiting for transmit status 011: Transferring transmit data to the transmit FIFO 100: Writing timestamp 110: Paused; transmit descriptor unavailable or transmit data underflow 111: Closing transmit descriptor

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19:17	RPS	R	0	Receive status: 000: Stopped; after reset or upon receiving a stop command 001: Reading receive descriptor 011: Waiting for packet 100: Paused; receive descriptor unavailable 101: Closing receive descriptor 110: Writing timestamp 111: Transferring received data to external storage
16	NIS	R	0	At least one interrupt among bits 0, 2, 6, and 14 is pending.
15	AIS	R	0	At least one interrupt among bits 1, 3, 4, 5, 7, 8, 9, 10 and 13 is pending.
14	ER	R	0	Receive early interrupt, indicating that the DMA has filled the first received data into the buffer; writing 1 to this bit or bit 6 clears it.
13	FBE	R	0	Bus error interrupt, refer to bits [25:23] for determination; this will cause the DMA controller to stop the transfer; writing 1 clears it.
12:11	RSV	-	-	-
10	ET	R	0	Transmit early interrupt, indicating that the transmit data is fully in the transmit FIFO; writing 1 clears it.
9	RWT	R	0	Receive watchdog interrupt, indicating that the current received frame is too long; writing 1 clears it.
8	RPSI	R	0	Receive stop interrupt; writing 1 clears it.
7	RBU	R	0	Receive buffer unavailable interrupt, indicating that DMA cannot obtain the transfer descriptor; the DMA can be commanded to attempt to resume the transfer using the RPD register; the transfer will also resume automatically after the next frame is received; writing 1 clears it.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	RI	R	0	Receive complete interrupt, reception will continue; writing 1 clears it.
5	TU	R	0	Transmit data underflow interrupt, transmission will pause; writing 1 clears it.
4	RO	R	0	Receive data overflow interrupt, if an incomplete frame has been issued, bit 11 of the RDES0 descriptor will be set to 1; writing 1 clears it.
3	TJTO	R	0	Transmit Jabber timer timeout interrupt, usually occurs when the frame size is too large; will stop the transmission and set bit 14 of the TDES0 descriptor to 1; writing 1 clears it.
2	TBU	R	0	Transmit buffer unavailable interrupt, indicating that DMA cannot obtain the transfer descriptor; bits [22:20] will display the current transfer status; the DMA can be commanded to attempt to resume the transfer by setting bit 31 of TDES0 and using the TPD register; writing 1 clears it.
1	TSI	R	0	Transmit stop interrupt; writing 1 clears it.
0	TC	R	0	Transmit frame complete interrupt; writing 1 clears it.

34.6.7 DMA Operation Mode Register (DMA_OPMODE)

Offset address: 0x1018

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:27	RSV	-	-	-
26	DT	R/W	0	1: EMAC will not discard frames received with checksum errors. 0: If bit 7 is set to 0, the error frame will be discarded.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
25	RSF	R/W	0	1: DMA will only send data to external storage after a complete receive frame has been written to FIFO. 0: When the data in the receive FIFO exceeds the threshold set by bits [4:3], DMA will send the data to external storage.
24	DFF	R/W	0	1: The receive FIFO will not be cleared due to the unavailability of receive descriptors or buffers. 0: The FIFO will be cleared if receive descriptors or buffers are unavailable.
23:22	RSV	-	-	-
21	TSF	R/W	0	1: EMAC will send a frame only after it has been completely placed into the FIFO. 0: EMAC will start sending when the data in the transmit FIFO exceeds the threshold set by bits [16:14].
20	FTF	R/W	0	Writing 1 to this bit commands EMAC to clear the transmit FIFO. Do not write to this register again until this bit is automatically cleared.
19:17	RSV	-	-	-
16:14	TT	R/W	0	Set the transmit threshold: 000: 64 001: 128 010: 192 011: 256 100: 40 101: 32 110: 24 111: 16
13	ST	R/W	0	1: When this bit is set to 1, DMA begins to search for descriptors of frames to be transmitted, looking for the last incomplete transfer descriptor or searching from the address in the TLA register. If no descriptors are available, DMA will pause and set bit 2 of

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				DMASTATUS to 1. This bit is only effective when set to 1 while in the stopped state. 0: After transmitting the current frame, DMA will enter a stopped state, making the next descriptor the current descriptor.
12:8	RSV	-	-	-
7	FEF	R/W	0	1: All received frames, except for those that are too short, will enter the DMA. 0: The receive FIFO will discard error frames, but if bit 25 is 0, frames that have already been partially sent via DMA will not be discarded.
6	FUGF	R/W	0	1: DMA will forward error-free received frames shorter than 64 bytes to external storage. 0: DMA will not forward short frames unless a portion has already been sent due to the frame threshold setting.
5	DGF	R/W	0	1: Discard received jumbo frames. 0: Do not discard received jumbo frames.
4:3	RT	R/W	0	Set the receive threshold: 00: 64 01: 32 10: 96 11: 128
2	OSF	R/W	0	1: DMA will start transferring the next transmit frame in advance. 0: DMA will not start transferring the next transmit frame in advance.
1	SR	R/W	0	1: When this bit is set to 1, DMA will search for descriptors of received frames, starting from the address in the RLA register. If no descriptors are available, DMA will pause and set bit 7 of DMASTATUS to 1. This bit is only effective when set to 1 while in the stopped state.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: After transferring the current receiving frame, DMA will enter a stopped state, making the next descriptor the current descriptor.
0	RSV	-	-	-

34.6.8 DMA Interrupt Enable Register (DMA_INTEN)

Offset address: 0x101C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	-
16	NIE	R/W	0	1: Used in combination with bits 0, 2, 6, and 14, allowing them to be enabled. 0: Prohibit these interrupts from being enabled.
15	AIE	R/W	0	1: Used in combination with bits 1, 3, 4, 5, 7, 8, 9, 10, and 13, allowing them to be enabled. 0: Prohibit these interrupts from being enabled.
14	ERE	R/W	0	1: Enable early receive interrupt. 0: Disable interrupt.
13	FBEE	R/W	0	1: Enable bus error interrupt. 0: Disable interrupt.
12:11	RSV	-	-	-
10	ETE	R/W	0	1: Enable early transmit interrupt. 0: Disable interrupt.
9	RWTE	R/W	0	1: Enable receive watchdog interrupt. 0: Disable interrupt.
8	RPSIE	R/W	0	1: Enable receive stopped interrupt. 0: Disable interrupt.
7	RBUE	R/W	0	1: Enable receive buffer unavailable interrupt. 0: Disable interrupt.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	RIE	R/W	0	1: Enables receive complete interrupt. 0: Disable interrupt.
5	TUE	R/W	0	1: Enable transmit underflow interrupt. 0: Disable interrupt.
4	ROE	R/W	0	1: Enable receive overflow interrupt. 0: Disable interrupt.
3	TJTOE	R/W	0	1: Enable transmit jabber timeout register. 0: Disable interrupt.
2	TBUE	R/W	0	1: Enable transmit buffer unavailable interrupt. 0: Disable interrupt.
1	TSIE	R/W	0	1: Enable transmit stopped interrupt. 0: Disable interrupt.
0	TCE	R/W	0	1: Enable transmit frame complete interrupt. 0: Disable interrupt.

34.6.9 DMA Lost Frame and Buffer Overflow Counter Register (DMA_FMBOCNT)

Offset address: 0x1020

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	-
28	OVCNTOV	R	0	Indicate that the receive FIFO overflow counter has overflowed. Reading this register clears it.
27:17	OVCNT	R	0	Record the number of frames lost due to receive FIFO overflow. Reading this register clears it when mci_be_i[2] is 1.
16	MISCNTOV	R	0	Indicate that the lost frame counter has overflowed. Reading this register clears it.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	MISCNT	R	0	Record the number of frames discarded by the DMA. Reading this register clears it when mci_be_i[0] is 1.

34.6.10 Receive Interrupt Watchdog Timer Register (DMA_RIWDT)

Offset address: 0x1024

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	-
7:0	RIWDT	R/W	0	After the receive DMA completes a transfer, if RDES1[31] is 0, this counter will decrease with the system clock divided by 256. When it decreases to 0, RDES1[31] will be set.

34.6.11 DMA Current Transmit Descriptor Register (DMA_CTD)

Offset address: 0x1048

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CTD	R	0	Address of current descriptor: Cleared on reset and automatically updated during DMA operation.

34.6.12 DMA Current Receive Descriptor Register (DMA_CRD)

Offset address: 0x104C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CRD	R	0	Address of current descriptor: Cleared on reset and automatically updated during DMA operation.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.6.13 DMA Current Transmit Buffer Address Register (DMA_CTB)

Offset address: 0x1050

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CTB	R	0	Address of current DMA operation: Cleared on reset and automatically updated during DMA operation.

34.6.14 DMA Current Receive Buffer Address Register (DMA_CRB)

Offset address: 0x1054

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	CRB	R	0	Address of current DMA operation: Cleared on reset and automatically updated during DMA operation.

34.7 List of MMC Registers

EMAC register base address: 0x4010_0000

The registers are listed below:

Table 34-5: List of MMC Registers

Address Offset	Register Name	Description
0x100	MMC_CONTROL	Control register
0x104	MMC_RI	Receive interrupt register
0x108	MMC_TI	Transmit interrupt register
0x10C	MMC_RIM	Receive interrupt mask register
0x110	MMC_TIM	Transmit interrupt mask register
0x114	MMC_TXBYTE	Number of bytes transmitted, excluding preamble and retry bytes

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x118	MMC_TXF	Number of frames transmitted, excluding retry frames
0x11C	MMC_TXBCGF	Number of correctly transmitted playback frames
0x120	MMC_TXMCGF	Number of correctly transmitted multicast frames
0x124	MMC_TXF64	Number of frames transmitted with fewer than 64 bytes, excluding preamble and retry frames
0x128	MMC_TXF65T127	Number of frames transmitted with 65–127 bytes, excluding preamble and retry frames
0x12C	MMC_TXF128T255	Number of frames transmitted with 128–255 bytes, excluding preamble and retry frames
0x130	MMC_TXF256T511	Number of frames transmitted with 256–511 bytes, excluding preamble and retry frames
0x134	MMC_TXF512T1023	Number of frames transmitted with 512–1023 bytes, excluding preamble and retry frames
0x138	MMC_TXF1024	Number of frames transmitted with more than 1024 bytes, excluding preamble and retry frames
0x13C	MMC_TXUCF	Number of unicast frames transmitted
0x140	MMC_TXMCF	Number of multicast frames transmitted
0x144	MMC_TXBCF	Number of playback frames transmitted
0x148	MMC_TXUDF	Number of frames discarded due to underrun errors
0x14C	MMC_TXSGCOL	Number of frames successfully transmitted after a single collision in half-duplex mode
0x150	MMC_TXMTCOL	Number of frames successfully transmitted after multiple collisions in half-duplex mode
0x154	MMC_TXDEF	Number of frames successfully transmitted after a delay in half-duplex mode
0x158	MMC_TXLATECOL	Number of frames discarded due to late collisions
0x15C	MMC_TXEXSCOL	Number of frames discarded due to excessive collisions
0x160	MMC_TXCARERR	Number of frames discarded due to carrier errors

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x164	MMC_TXGBYTE	Number of bytes in correctly transmitted frames, excluding preamble
0x168	MMC_TXGF	Number of correctly transmitted frames
0x16C	MMC_TXEXSDEF	Number of frames discarded due to excessive deferral
0x170	MMC_TXPAUSEF	Number of correctly transmitted pause frames
0x174	MMC_TXVLANGF	Number of correctly transmitted VLAN frames
0x178	MMC_TXOVSIZEGF	Number of transmitted error-free oversize frames
0x180	MMC_RXF	Number of frames received
0x184	MMC_RXBYTE	Number of bytes received, excluding preamble
0x188	MMC_RXGBYTE	Number of bytes in correctly received frames, excluding preamble
0x18C	MMC_RXBCGF	Number of correctly received playback frames
0x190	MMC_RXMCGF	Number of correctly received multicast frames
0x194	MMC_RXCRCERR	Number of frames received with CRC errors
0x198	MMC_RXALGERR	Number of frames received with alignment errors (only in 10/100 mode)
0x19C	MMC_RXRUNTERR	Number of runt frames received with CRC errors
0x1A0	MMC_RXJABERR	Number of jumbo frames received with CRC errors
0x1A4	MMC_RXUDSIZEGF	Number of undersize frames received without errors
0x1A8	MMC_RXOVSIZEGF	Number of oversize frames received without errors
0x1AC	MMC_RXF64	Number of frames received with fewer than 64 bytes, excluding preamble
0x1B0	MMC_RXF65T127	Number of frames received with 65–127 bytes, excluding preamble
0x1B4	MMC_RXF128T255	Number of frames received with 128–255 bytes, excluding preamble
0x1B8	MMC_RXF256T511	Number of frames received with 256–511 bytes, excluding preamble
0x1BC	MMC_RXF512T1023	Number of frames received with 512–1023 bytes, excluding preamble

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x1C0	MMC_RXF1024	Number of frames received with more than 1024 bytes, excluding preamble
0x1C4	MMC_RXUCGF	Number of correctly received unicast frames
0x1C8	MMC_RXLERR	Number of frames received with length errors
0x1CC	MMC_OUTRANGE	Number of frames received with illegal length errors
0x1D0	MMC_RXPAUSEF	Number of pause frames received
0x1D4	MMC_RXFIFOOVF	Number of frames lost due to RX FIFO overflow
0x1D8	MMC_RXVLANF	Number of VLAN frames received
0x1DC	MMC_RXWDGERR	Number of frames received with watchdog timeout errors
0x1E0	MMC_RXRCVERR	Number of frames with receive errors and extended errors
0x1E4	MMC_RXCTRLGF	Number of correctly received control frames
0x200	MMC_IPCINTRMASK	Receive checksum interrupt mask register
0x208	MMC_IPCINTR	Receive checksum interrupt register
0x210	MMC_RXIPV4F	Number of correctly received IPV4 data frames
0x214	MMC_RXIPV4HDRERR	Number of IPV4 frames received with header errors (checksum, length, version errors)
0x218	MMC_RXIPV4NPF	Number of IPV4 frames received without payload
0x21C	MMC_RXIPV4FRAG	Number of IPV4 fragmented data frames received
0x220	MMC_RXIPV4UDSBL	Number of IPV4 frames received without checksum UDP
0x224	MMC_RXIPV6F	Number of correctly received IPV6 data frames
0x228	MMC_RXIPV6HDRERR	Number of IPV6 frames received with header errors (length, version errors)
0x22C	MMC_RXIPV6NPF	Number of IPV6 frames received without payload, including all IPV6 frames with fragmentation and security extension headers
0x230	MMC_RXUDPF	Number of UDP frames received (which does not increment when RXIPV4UDSBL increases)
0x234	MMC_RXUDPERRF	Number of UDP frames received with checksum errors
0x238	MMC_RXTCPF	Number of TCP frames received

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Address Offset	Register Name	Description
0x23C	MMC_RXTCPERRF	Number of TCP frames received with checksum errors
0x240	MMC_RXICMPF	Number of ICMP frames received
0x244	MMC_RXICMPERRF	Number of ICMP frames received with checksum errors
0x250	MMC_RXIPV4BYTE	Number of bytes in correctly received IPV4 data frames
0x254	MMC_RXIPV4HDRERB	Number of bytes in IPV4 frames received with header errors
0x258	MMC_RXIPV4NPB	Number of bytes in IPv4 frames received without payload
0x25C	MMC_RXIPV4FRAGB	Number of bytes in IPV4 fragmented data frames received
0x260	MMC_RXIPV4UDSBLB	Number of bytes in IPV4 frames received without checksum UDP
0x264	MMC_RXIPV6BYTE	Number of bytes in correctly received IPV6 data frames
0x268	MMC_RXIPV6HDRERB	Number of bytes in IPV6 frames received with header errors
0x26C	MMC_RXIPV6NPB	Number of bytes in IPV6 frames received without payload
0x270	MMC_RXUDPBYTE	Number of bytes in correctly received UDP frames
0x274	MMC_RXUDPERRB	Number of bytes in UDP frames received with checksum errors
0x278	MMC_RXTCPBYTE	Number of bytes in correctly received TCP frames
0x27C	MMC_RXTCPERRB	Number of bytes in TCP frames received with checksum errors
0x280	MMC_RXICMPBYTE	Number of bytes in correctly received ICMP frames
0x284	MMC_RXICMPERRB	Number of bytes in ICMP frames received with checksum errors

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.7.1 MMC Control Register (MMC_CONTROL)

Offset address: 0x100

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	-
8	UCDBC	R/W	0	1: Filtered broadcast frames will update the MMC counter. 0: Filtered broadcast frames will not update the MMC counter.
7:6	RSV	-	-	-
5	PRSTVL	R/W	0	1: The preset value is counter full. 0: The preset value is counter half-full.
4	PRST	W	0	Writing 1 sets the counter value to the preset value.
3	CNTFREEZ	R/W	0	1: The counter value will not be updated due to the transmitted and received frames. 0: The counter value will be updated.
2	RSTONRD	R/W	0	1: Reading the counter will reset its value to zero. 0: Reading will not reset the value to zero.
1	NRO	R/W	0	1: The counter value will not return to 0 when it reaches full. 0: The counter value will return to 0 when it reaches full.
0	RST	W	0	Writing 1 resets the counter.

34.7.2 MMC Receive Interrupt Register (MMC_RI)

Offset address: 0x104

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
25	RXCTRFIS	R	0	This bit is set when the receive control frame counter RXCTRLGF reaches half of the maximum value or the maximum value, and is cleared on read.
24	RXERRFIS	R	0	This bit is set when the receive error frame counter RXRCVERR reaches half of the maximum value or the maximum value, and is cleared on read.
23	RXWDERRFIS	R	0	This bit is set when the receive timeout error frame counter RXWDGERR reaches half of the maximum value or the maximum value, and is cleared on read.
22	RXVLANIS	R	0	This bit is set when the receive VLAN frame counter RXVLANF reaches half of the maximum value or the maximum value, and is cleared on read.
21	RXFOVFIS	R	0	This bit is set when the receive FIFO overflow counter RXFIFOOVF reaches half of the maximum value or the maximum value, and is cleared on read.
20	RXPAUSFIS	R	0	This bit is set when the receive pause frame counter RXPAUSEF reaches half of the maximum value or the maximum value, and is cleared on read.
19	RXORANGFIS	R	0	This bit is set when the receive out-of-range error frame counter OUTRANGE reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
18	RXLERRFIS	R	0	This bit is set when the receive length error frame counter RXLERR reaches half of the maximum value or the maximum value, and is cleared on read.
17	RXUCGFIS	R	0	This bit is set when the receive unicast good frame counter RXUCGF reaches half of the maximum value or the maximum value, and is cleared on read.
16	RX1024TMAXFIS	R	0	This bit is set when the receive over 1024 frame counter RXF1024 reaches half of the maximum value or the maximum value, and is cleared on read.
15	RX512T1023FIS	R	0	This bit is set when the receive 512 to 1023 frame counter RXF512T1023 reaches half of the maximum value or the maximum value, and is cleared on read.
14	RX256T511FIS	R	0	This bit is set when the receive 256 to 511 frame counter RXF256T511 reaches half of the maximum value or the maximum value, and is cleared on read.
13	RX128T255FIS	R	0	This bit is set when the receive 128 to 255 frame counter RXF128T255 reaches half of the maximum value or the maximum value, and is cleared on read.
12	RX65T127FIS	R	0	This bit is set when the receive 65 to 127 frame counter RXF65T127 reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
11	RX64FIS	R	0	This bit is set when the receive below 64 frame counter RXF64 reaches half of the maximum value or the maximum value, and is cleared on read.
10	RXOSIZEGFIS	R	0	This bit is set when the receive oversize good frame counter RXOVSIZEGF reaches half of the maximum value or the maximum value, and is cleared on read.
9	RXUSIZEGFIS	R	0	This bit is set when the receive undersize good frame counter RXUDSIZEGF reaches half of the maximum value or the maximum value, and is cleared on read.
8	RXJABERFIS	R	0	This bit is set when the receive jabber error frame counter RXJABERR reaches half of the maximum value or the maximum value, and is cleared on read.
7	RXRUNTFIS	R	0	This bit is set when the receive runt error frame counter RXRUNTERR reaches half of the maximum value or the maximum value, and is cleared on read.
6	RXALIGNFIS	R	0	This bit is set when the receive alignment error frame counter RXALGERR reaches half of the maximum value or the maximum value, and is cleared on read.
5	RXCRCERFIS	R	0	This bit is set when the receive CRC error frame counter RXCRCERR reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4	RXMCGFIS	R	0	This bit is set when the receive multicast good frame counter RXMCGF reaches half of the maximum value or the maximum value, and is cleared on read.
3	RXBCGFIS	R	0	This bit is set when the receive broadcast good frame counter RXBCGF reaches half of the maximum value or the maximum value, and is cleared on read.
2	RXGBOIS	R	0	This bit is set when the receive good byte counter RXGBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
1	RXOIS	R	0	This bit is set when the receive byte counter RXBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
0	RXFIS	R	0	This bit is set when the receive frame counter RXF reaches half of the maximum value or the maximum value, and is cleared on read.

34.7.3 MMC Transmit Interrupt Register (MMC_TI)

Offset address: 0x108

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25	TXOSIZEGFIS	R	0	This bit is set when the transmit oversize good frame counter TXOVSIZEGF reaches half of the maximum value or the maximum value, and is cleared on read.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
24	TXVLANGFIS	R	0	This bit is set when the transmit VLAN good frame counter TXVLANGF reaches half of the maximum value or the maximum value, and is cleared on read.
23	TXPAUSFIS	R	0	This bit is set when the transmit pause frame counter TXPAUSEF reaches half of the maximum value or the maximum value, and is cleared on read.
22	TXEXDEFFIS	R	0	This bit is set when the transmit excessive deferral frame counter TXEXSDEF reaches half of the maximum value or the maximum value, and is cleared on read.
21	TXGFIS	R	0	This bit is set when the transmit good frame counter TXGF reaches half of the maximum value or the maximum value, and is cleared on read.
20	TXGOIS	R	0	This bit is set when the transmit good byte counter TXGBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
19	TXCARERFIS	R	0	This bit is set when the transmit carrier error frame counter TXCARERR reaches half of the maximum value or the maximum value, and is cleared on read.
18	TXEXCOLFIS	R	0	This bit is set when the transmit excessive collision frame counter TXEXSCOL reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
17	TXLACOLFIS	R	0	This bit is set when the transmit late collision frame counter TXLATECOL reaches half of the maximum value or the maximum value, and is cleared on read.
16	TXDEFFIS	R	0	This bit is set when the transmit deferred frame counter TXDEF reaches half of the maximum value or the maximum value, and is cleared on read.
15	TXMCOLGFIS	R	0	This bit is set when the transmit multiple collision good frame counter TXMTCOL reaches half of the maximum value or the maximum value, and is cleared on read.
14	TXSCOLGFIS	R	0	This bit is set when the transmit single collision good frame counter TXSGCOL reaches half of the maximum value or the maximum value, and is cleared on read.
13	TXUFERFIS	R	0	This bit is set when the transmit underflow frame counter TXUDF reaches half of the maximum value or the maximum value, and is cleared on read.
12	TXBCFIS	R	0	This bit is set when the transmit broadcast frame counter TXBCF reaches half of the maximum value or the maximum value, and is cleared on read.
11	TXMCFIS	R	0	This bit is set when the transmit multicast frame counter TXMCF reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	TXUCFIS	R	0	This bit is set when the transmit unicast frame counter TXUCF reaches half of the maximum value or the maximum value, and is cleared on read.
9	TX1024TMAXFIS	R	0	This bit is set when the transmit over 1024 frame counter TXF1024 reaches half of the maximum value or the maximum value, and is cleared on read.
8	TX512T1023FIS	R	0	This bit is set when the transmit 512 to 1023 frame counter TXF512T123 reaches half of the maximum value or the maximum value, and is cleared on read.
7	TX256T511FIS	R	0	This bit is set when the transmit 256 to 511 frame counter TXF256T511 reaches half of the maximum value or the maximum value, and is cleared on read.
6	TX128T255FIS	R	0	This bit is set when the transmit 128 to 255 frame counter TXF128T255 reaches half of the maximum value or the maximum value, and is cleared on read.
5	TX65T127FIS	R	0	This bit is set when the transmit 65 to 127 frame counter TXF65T127 reaches half of the maximum value or the maximum value, and is cleared on read.
4	TX64FIS	R	0	This bit is set when the transmit below 64 frame counter TXF64 reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	TXMCGFIS	R	0	This bit is set when the transmit multicast good frame counter TXMCGF reaches half of the maximum value or the maximum value, and is cleared on read.
2	TXBCGFIS	R	0	This bit is set when the transmit broadcast good frame counter TXBCGF reaches half of the maximum value or the maximum value, and is cleared on read.
1	TXFIS	R	0	This bit is set when the transmit frame counter TXF reaches half of the maximum value or the maximum value, and is cleared on read.
0	TXOIS	R	0	This bit is set when the transmit byte counter TXBYTE reaches half of the maximum value or the maximum value, and is cleared on read.

34.7.4 MMC Receive Interrupt Mask Register (MMC_RIM)

Offset address: 0x10C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25	RXCTRFIM	R/W	0	Interrupt mask setting bit for the receive control frame counter RXCTRLGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
24	RXERRFIM	R/W	0	Interrupt mask setting bit for the receive error frame counter RXRCVERR: 1: Interrupt will not be triggered.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Interrupt can be triggered.
23	RXWDERRFIM	R/W	0	Interrupt mask setting bit for the receive timeout error frame counter RXWDGERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
22	RXVLANIM	R/W	0	Interrupt mask setting bit for the receive VLAN frame counter RXVLANF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
21	RXFOVFIM	R/W	0	Interrupt mask setting bit for the receive FIFO overflow counter RXFIFOOVF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
20	RXPAUSFIM	R/W	0	Interrupt mask setting bit for the receive pause frame counter RXPAUSEF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
19	RXORANGFIM	R/W	0	Interrupt mask setting bit for the receive address out-of-range counter OUTRANGE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
18	RXLERRFIM	R/W	0	Interrupt mask setting bit for the receive frame length error counter RXLERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
17	RXUCGFIM	R/W	0	Interrupt mask setting bit for the receive unicast good frame counter RXUCGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
16	RX1024TMAXFIM	R/W	0	Interrupt mask setting bit for the receive over 1024 frame counter RXF1024: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
15	RX512T1023FIM	R/W	0	Interrupt mask setting bit for the receive 512 to 1023 frame counter RXF512T1023: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
14	RX256T511FIM	R/W	0	Interrupt mask setting bit for the receive 256 to 511 frame counter RXF256T511: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
13	RX128T255FIM	R/W	0	Interrupt mask setting bit for the receive 128 to 255 frame counter RXF128T255: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
12	RX65T127FIM	R/W	0	Interrupt mask setting bit for the receive 65 to 127 frame counter RXF65T127: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
11	RX64FIM	R/W	0	Interrupt mask setting bit for the receive below 64 frame counter RXF64: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
10	RXOSIZEGFIM	R/W	0	Interrupt mask setting bit for the receive oversize good frame counter RXOVSIZEGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
9	RXUSIZEGFIM	R/W	0	Interrupt mask setting bit for the receive undersize good frame counter RXOVSIZEGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
8	RXJABERFIM	R/W	0	Interrupt mask setting bit for the receive jabber error frame counter RXJABERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
7	RXRUNTFIM	R/W	0	Interrupt mask setting bit for the receive runt error frame counter RXRUNTERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
6	RXALIGNFIM	R/W	0	Interrupt mask setting bit for the receive alignment error frame counter RXALGERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
5	RXCRCERFIM	R/W	0	Interrupt mask setting bit for the receive CRC error frame counter RXRCVERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
4	RXMCGFIM	R/W	0	Interrupt mask setting bit for the receive multicast good frame counter RXMCGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
3	RXBCGFIM	R/W	0	Interrupt mask setting bit for the receive broadcast good frame counter RXBCGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	RXGBOIM	R/W	0	Interrupt mask setting bit for the receive good byte counter RXGBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
1	RXOIM	R/W	0	Interrupt mask setting bit for the receive byte counter RXBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
0	RXFIM	R/W	0	Interrupt mask setting bit for the receive frame counter RXF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

34.7.5 MMC Transmit Interrupt Mask Register (MMC_TIM)

Offset address: 0x110

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	-
25	TXOSIZEGFIM	R/W	0	Interrupt mask setting bit for the transmit oversize good frame counter TXOVSIZEGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
24	TXVLANGFIM	R/W	0	Interrupt mask setting bit for the transmit VLAN good frame counter TXVLANGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
23	TXPAUSFIM	R/W	0	Interrupt mask setting bit for the transmit pause frame counter TXPAUSEF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
22	TXEXDEFFIM	R/W	0	Interrupt mask setting bit for the transmit excessive deferral frame counter TXEXSDEF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
21	TXGFIM	R/W	0	Interrupt mask setting bit for the transmit good frame counter TXGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
20	TXGOIM	R/W	0	Interrupt mask setting bit for the transmit good byte counter TXGBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
19	TXCARERFIM	R/W	0	Interrupt mask setting bit for the transmit carrier error frame counter TXCARERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
18	TXEXCOLFIM	R/W	0	Interrupt mask setting bit for the transmit excessive collision frame counter TXEXSCOL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
17	TXLACOLFIM	R/W	0	Interrupt mask setting bit for the transmit late collision frame counter TXLATECOL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
16	TXDEFFIM	R/W	0	Interrupt mask setting bit for the transmit deferred frame counter TXDEF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15	TXMCOLGFIM	R/W	0	Interrupt mask setting bit for the transmit multiple collision good frame counter TXMTCOL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
14	TXSCOLGFIM	R/W	0	Interrupt mask setting bit for the transmit single collision good frame counter TXSGCOL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
13	TXUFERFIM	R/W	0	Interrupt mask setting bit for the transmit underflow error frame counter TXUDF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
12	TXBCFIM	R/W	0	Interrupt mask setting bit for the transmit broadcast frame counter TXBCF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
11	TXMCFIM	R/W	0	Interrupt mask setting bit for the transmit multicast frame counter TXMCF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
10	TXUCFIM	R/W	0	Interrupt mask setting bit for the transmit unicast frame counter TXUCF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
9	TX1024TMAXFIM	R/W	0	Interrupt mask setting bit for the transmit over 1024 frame counter TXF1024: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
8	TX512T1023FIM	R/W	0	Interrupt mask setting bit for the transmit 512 to 1023 frame counter TXF512T1023: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
7	TX256T511FIM	R/W	0	Interrupt mask setting bit for the transmit 256 to 511 frame counter TXF256T511: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
6	TX128T255FIM	R/W	0	Interrupt mask setting bit for the transmit 128 to 255 frame counter TXF128T255: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
5	TX65T127FIM	R/W	0	Interrupt mask setting bit for the transmit 65 to 127 frame counter TXF65T127: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
4	TX64FIM	R/W	0	Interrupt mask setting bit for the transmit below 64 frame counter TXF64: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
3	TXMCGFIM	R/W	0	Interrupt mask setting bit for the transmit multicast good frame counter TXMCGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
2	TXBCGFIM	R/W	0	Interrupt mask setting bit for the transmit broadcast good frame counter TXBCGF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	TXFIM	R/W	0	Interrupt mask setting bit for the transmit frame counter TXF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
0	TXOIM	R/W	0	Interrupt mask setting bit for the transmit byte counter TXBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

34.7.6 MMC Receive Checksum Interrupt Mask Register (MMC_IPCINTRMASK)

Offset address: 0x200

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	-
29	RXICMPEROIM	R/W	0	Interrupt mask setting bit for the received ICMP checksum error frame byte counter RXICMPERRB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
28	RXICMPGOIM	R/W	0	Interrupt mask setting bit for the received good ICMP frame byte counter RXICMPBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
27	RXTCPEROIM	R/W	0	Interrupt mask setting bit for the received TCP checksum error frame byte counter RXTCPERRB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
26	RXTCPGOIM	R/W	0	Interrupt mask setting bit for the received good TCP frame byte counter RXTCPBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
25	RXUDPEROIM	R/W	0	Interrupt mask setting bit for the received UDP checksum error frame byte counter RXUDPERRB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
24	RXUDPGOIM	R/W	0	Interrupt mask setting bit for the received good UDP frame byte counter RXUDPBYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
23	RXIPV6NPOIM	R/W	0	Interrupt mask setting bit for the received IPV6 no payload frame byte counter RXIPV6NPB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
22	RXIPV6HEROIM	R/W	0	Interrupt mask setting bit for the received IPV6 header error frame byte counter RXIPV6HDRERB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
21	RXIPV6GOIM	R/W	0	Interrupt mask setting bit for the received IPV6 good data frame byte counter RXIPV6BYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
20	RXIPV4UDSBOIM	R/W	0	Interrupt mask setting bit for the received IPV4 UDP checksum disabled frame byte counter RXIPV4UDSBL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19	RXIPV4FRAOIM	R/W	0	Interrupt mask setting bit for the received IPV4 fragmented data frame byte counter RXIPV4FRAGB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
18	RXIPV4NPOIM	R/W	0	Interrupt mask setting bit for the received IPV4 no payload frame byte counter RXIPV4NPB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
17	RXIPV4HEROIM	R/W	0	Interrupt mask setting bit for the received IPV4 header error frame byte counter RXIPV4HDRERB: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
16	RXIPV4GOIM	R/W	0	Interrupt mask setting bit for the received IPV4 good data frame byte counter RXIPV4BYTE: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
15:14	RSV	-	-	-
13	RXICMPERFIM	R/W	0	Interrupt mask setting bit for the received ICMP checksum error frame counter RXICMPERRF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
12	RXICMPGFIM	R/W	0	Interrupt mask setting bit for the received ICMP frame counter RXICMPF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
11	RXTCPERFIM	R/W	0	Interrupt mask setting bit for the received TCP checksum error frame counter RXTCPERRF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
10	RXTCPGFIM	R/W	0	Interrupt mask setting bit for the received TCP frame counter RXTCPF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
9	RXUDPERFIM	R/W	0	Interrupt mask setting bit for the received UDP checksum error frame counter RXUDPERRF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
8	RXUDPGFIM	R/W	0	Interrupt mask setting bit for the received UDP frame counter RXUDPF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
7	RXIPV6NPFIM	R/W	0	Interrupt mask setting bit for the received IPV6 no payload frame counter RXIPV6NPF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
6	RXIPV6HERFIM	R/W	0	Interrupt mask setting bit for the received IPV6 header error frame counter RXIPV6HDRERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
5	RXIPV6GFIM	R/W	0	Interrupt mask setting bit for the received IPV6 good data frame counter RXIPV6F: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
4	RXIPV4UDSBFIM	R/W	0	Interrupt mask setting bit for the received IPV4 UDP checksum disabled frame counter RXIPV4UDSBL: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	RXIPV4FRAFIM	R/W	0	Interrupt mask setting bit for the received IPV4 fragmented data frame counter RXIPV4FRAG: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
2	RXIPV4NPFIM	R/W	0	Interrupt mask setting bit for the received IPV4 no payload frame counter RXIPV4NPF: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
1	RXIPV4HERFIM	R/W	0	Interrupt mask setting bit for the received IPV4 header error frame counter RXIPV4HDRERR: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.
0	RXIPV4GFIM	R/W	0	Interrupt mask setting bit for the received IPV4 good data frame counter RXIPV4F: 1: Interrupt will not be triggered. 0: Interrupt can be triggered.

34.7.7 MMC Receive Checksum Interrupt Register (MMC_IPCINTR)

Offset address: 0x208

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	-
29	RXICMPEROIS	R	0	This bit is set when the received ICMP checksum error frame byte counter RXICMPERRB reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
28	RXICMPGOIS	R	0	This bit is set when the received good ICMP frame byte counter RXICMPBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
27	RXTCPEROIS	R	0	This bit is set when the received TCP checksum error frame byte counter RXTCPERRB reaches half of the maximum value or the maximum value, and is cleared on read.
26	RXTCPGOIS	R	0	This bit is set when the received good TCP frame byte counter RXTCPBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
25	RXUDPEROIS	R	0	This bit is set when the received UDP checksum error frame byte counter RXUDPERRB reaches half of the maximum value or the maximum value, and is cleared on read.
24	RXUDPGOIS	R	0	This bit is set when the received good UDP frame byte counter RXUDPBYTE reaches half of the maximum value or the maximum value, and is cleared on read.
23	RXIPV6NPOIS	R	0	This bit is set when the received IPV6 no payload frame byte counter RXIPV6NPB reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
22	RXIPV6HEROIS	R	0	This bit is set when the received IPV6 header error frame byte counter RXIPV6HDRERB reaches half of the maximum value or the maximum value, and is cleared on read.
21	RXIPV6GOIS	R	0	This bit is set when the receive IPV6 good data frame byte counter RXIPV6BYTE reaches half of the maximum value or the maximum value, and is cleared on read.
20	RXIPV4UDSBOIS	R	0	This bit is set when the received IPV4 UDP checksum disabled frame byte counter RXIPV4UDSBL reaches half of the maximum value or the maximum value, and is cleared on read.
19	RXIPV4FRAOIS	R	0	This bit is set when the received IPV4 fragmented data frame byte counter RXIPV4FRAGB reaches half of the maximum value or the maximum value, and is cleared on read.
18	RXIPV4NPOIS	R	0	This bit is set when the received IPV4 no payload frame byte counter RXIPV4NPB reaches half of the maximum value or the maximum value, and is cleared on read.
17	RXIPV4HEROIS	R	0	This bit is set when the received IPV4 header error frame byte counter RXIPV4HDRERB reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
16	RXIPV4GOIS	R	0	This bit is set when the received IPV4 good data frame byte counter RXIPV4BYTE reaches half of the maximum value or the maximum value, and is cleared on read.
15:14	RSV	-	-	-
13	RXICMPERFIS	R	0	This bit is set when the received ICMP checksum error frame counter RXICMPERRF reaches half of the maximum value or the maximum value, and is cleared on read.
12	RXICMPGFIS	R	0	This bit is set when the received ICMP frame counter RXICMPF reaches half of the maximum value or the maximum value, and is cleared on read.
11	RXTCPERFIS	R	0	This bit is set when the received TCP checksum error frame counter RXTCPERRF reaches half of the maximum value or the maximum value, and is cleared on read.
10	RXTCPGFIS	R	0	This bit is set when the received TCP frame counter RXTCPF reaches half of the maximum value or the maximum value, and is cleared on read.
9	RXUDPERFIS	R	0	This bit is set when the received UDP checksum error frame counter RXUDPERRF reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
8	RXUDPGFIS	R	0	This bit is set when the received UDP frame counter RXUDPF reaches half of the maximum value or the maximum value, and is cleared on read.
7	RXIPV6NPFIS	R	0	This bit is set when the received IPV6 no payload frame counter RXIPV6NPF reaches half of the maximum value or the maximum value, and is cleared on read.
6	RXIPV6HERFIS	R	0	This bit is set when the received IPV6 header error frame counter RXIPV6HDRERR reaches half of the maximum value or the maximum value, and is cleared on read.
5	RXIPV6GFIS	R	0	This bit is set when the receive IPV6 good data frame counter RXIPV6F reaches half of the maximum value or the maximum value, and is cleared on read.
4	RXIPV4UDSBFIS	R	0	This bit is set when the received IPV4 UDP checksum disabled frame counter RXIPV4UDSBL reaches half of the maximum value or the maximum value, and is cleared on read.
3	RXIPV4FRAFIS	R	0	This bit is set when the received IPV4 fragmented data frame counter RXIPV4FRAG reaches half of the maximum value or the maximum value, and is cleared on read.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
2	RXIPV4NPFIS	R	0	This bit is set when the received IPV4 no payload frame counter RXIPV4NPF reaches half of the maximum value or the maximum value, and is cleared on read.
1	RXIPV4HERFIS	R	0	This bit is set when the received IPV4 header error frame counter RXIPV4HDRERR reaches half of the maximum value or the maximum value, and is cleared on read.
0	RXIPV4GFIS	R	0	This bit is set when the receive IPV4 good data frame counter RXIPV4F reaches half of the maximum value or the maximum value, and is cleared on read.

34.8 DMA Descriptors

34.8.1 Transmit Descriptor 0 (TDES0)

Bit	Name	Description
31	OWN	1: The descriptor is owned by the DMA. 0: The descriptor is owned by the CPU. The DMA clears this bit either when it completes the frame transmission or when the buffers allocated in the descriptor are empty. This ownership bit of the frame shall be set after all subsequent descriptors belonging to the same frame have been set.
30:18	RSV	-
17	TTS	This status bit indicates that a timestamp has been captured and placed in TDES2 and TDES3, which is valid only when bit 30 of TDES1 is set.
16	IPHE	This bit indicates an IP header error, valid only when the checksum is enabled. If IPV4 content is used, the checksum will still be inserted.
15	ES	This bit indicates that at least one error has occurred among bits 16, 14, 13, 12, 11, 10, 9, 8, 2 and 1.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Description
14	JTO	This bit indicates a jabber timer timeout.
13	FF	This bit indicates that the CPU has commanded the DMA to clear the frame data in the FIFO.
12	PLE	This bit indicates a checksum error, usually caused by incorrect data length or a frame length exceeding the FIFO. In this case, the checksum will not be inserted, and this bit is valid only when the checksum is enabled.
11	LC	This bit indicates a carrier loss error, valid only in half-duplex mode and for the frames transmitted without collision.
10	NC	This bit indicates that there is no carrier.
9	LCOL	This bit indicates that a collision occurs after the collision window.
8	ECOL	This bit indicates that the transmission is aborted after 16 successive collisions while attempting to transmit the current frame. If retries are disabled, this bit is set after the first collision and the transmission of the frame is aborted.
7	VF	This bit indicates that the transmitted frame is a VLAN frame.
6:3	COLCNT	This 4-bit counter value indicates the number of collisions occurring before the frame is transmitted, valid when bit 8 of TDES0 is 0.
2	ED	This bit indicates that the transmission will be terminated because of excessive deferral of over 24,288 bit times (155,680 bits times in 1000-Mbps mode, or in jumbo frame mode) if bit 4 (deferral check) is set high in EMAC configuration register.
1	UF	This bit indicates a TX FIFO data underflow, which will pause the transmission.
0	DB	This bit indicates that a deferral has occurred due to the loss of carrier in half-duplex mode.

34.8.2 Transmit Descriptor 1 (TDES1)

Bit	Name	Description
31	IC	1: This bit sets the transmit interrupt (bit 0 of DMASTATUS register) after the present frame has been transmitted. This bit is valid only when the last segment bit (TDES1[30) is set.
30	LS	1: This bit indicates that the buffer contains the last segment of the current frame, and the TBS1 or TBS2 field shall have a non-zero value.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Description
29	FS	1: This bit indicates that the buffer contains the first segment of the current frame.
28:27	CIC	Set the checksum insertion: 00: No checksum insertion 01: Insert IPV4 header checksum 10: Insert TCP/UDP/ICMP checksum without pseudo-header 11: Insert TCP/UDP/ICMP checksum with pseudo-header These bits are valid only when TDES1[29] is set.
26	DC	1: Do not append CRC to the end of the transmitted frame. This bit is valid only when TDES1[29] is set.
25	TER	1: This bit indicates this is the last descriptor in the descriptor list, after which it will return to the initial descriptor address.
24	TCH	1: This bit indicates that the second address in the descriptor is the next descriptor address rather than the second buffer address. This bit is valid only when TDES1[25] is set.
23	DP	1: Do not automatically add padding to a frame shorter than 64 bytes. 0: Automatically add padding to a frame shorter than 64 bytes. This bit is valid only when TDES1[29] is set.
22	TTSE	1: Enable IEEE1588 hardware timestamping, valid only when TDES1[29] is set.
22:11	TBS2	Set the second data buffer byte size, valid only when TDES1[24] is set.
10:0	TBS1	Set the first data buffer byte size.

34.8.3 Transmit Descriptor 2 (TDES2)

Bit	Name	Description
31:0	B1A	Set the address of the first buffer or the lower bits of the timestamp.

34.8.4 Transmit Descriptor 3 (TDES3)

Bit	Name	Description
31:0	B2A	Set the address of the second buffer (or the address of the next descriptor) or the higher bits of the timestamp.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

34.8.5 Receive Descriptor 0 (RDES0)

Bit	Name	Description
31	OWN	1: The descriptor is owned by the DMA. 0: The descriptor is owned by the CPU. The DMA clears this bit either when it completes the frame transmission or when the buffers allocated in the descriptor are full.
30	DAFE	This bit indicates a frame that failed in the destination address filter.
29:16	FL	These bits indicate the byte length of the received frame, including the CRC length. This field is valid when RDES0[8] is 1 and RDES0[14] is 0. RDES0[8] being 0 indicates the number of bytes that have been transmitted in the current frame.
15	ES	This bit indicates that at least one error has occurred among bits 14, 11, 7, 6, 4, 3, 1 and 0.
14	DE	This bit indicates that the current frame is truncated because the current frame does not fit within the current descriptor buffers, and the DMA does not own the next descriptor. This field is valid only when RDES0[8] is 1.
13	SAFE	This bit indicates a frame that failed in the source address filter.
12	LE	This bit indicates that the actual length of the frame received and the length/ type field do not match.
11	OVF	This bit indicates that a FIFO overflow has occurred.
10	VF	This bit indicates that the frame pointed to by this descriptor is a VLAN frame.
9	FS	This bit indicates that this descriptor is the first descriptor of the frame, and the buffer of this descriptor is used to load the initial part of the frame.
8	LS	This bit indicates that this descriptor is the last descriptor of the frame, and the buffer of this descriptor is used to load the final part of the frame.
7	IPCSE	This bit indicates that a header checksum error has occurred.
6	LC	This bit indicates that a late collision has occurred while receiving the frame in half-duplex mode.
5	FT	1: The received frame is an Ethernet-type frame. 0: The received frame is an IEEE802.3 frame.
4	RWDTO	This bit indicates that a receive watchdog timeout has occurred.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Description
3	RE	This bit indicates that the RXER signal is asserted while the RXDV signal is asserted during frame reception.
2	DBE	This bit indicates that the received frame has a non-integer multiple of bytes (odd nibbles). This bit is valid only in MII Mode.
1	CRCE	This bit indicates that a CRC checksum error has occurred, valid only when RDES0[8] is 1.
0	PLCSE	This bit indicates that a data checksum error has occurred.

The types and statuses of the received frame indicated by bits 0, 5, and 7 in RDES0 are shown in the table below.

Bit5: FT	Bit7: IPCSE	Bit0: PLCSE	Frame Type and Status
0	0	0	IEEE 802.3 frame, with the length less than 1,536
1	0	0	When bit 10 of EMAC Config is 1: IPv4/IPv6 frame, no checksum error When bit 10 of EMAC Config is 0: IEEE 802.3 frame, with the length greater than or equal to 1,536
1	0	1	IPv4/IPv6 frame, with payload checksum error
1	1	0	IPv4/IPv6 frame, with header checksum error
1	1	1	IPv4/IPv6 frame, with both payload and header checksum error
0	0	1	IPv4/IPv6 frame, with payload check bypassed due to format mismatch
0	1	1	Neither IPv4 nor IPv6 frame, no checksum performed

34.8.6 Receive Descriptor 1 (RDES1)

Bit	Name	Description
31	DIC	1: Disable receive interrupt, valid only when bit 8 of RDES0 is 1.
30:26	RSV	Reserved
25	RER	1: This bit indicates this is the last descriptor in the descriptor list, after which it will return to the initial descriptor address.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Description
24	RCH	1: This bit indicates that the second address in the descriptor is the next descriptor address rather than the second buffer address. This bit is valid only when bit 25 is 0.
23:22	RSV	Reserved
22:11	RBS2	These bits indicate the second data buffer byte size, valid only when bit 24 is 0. The buffer size must be a multiple of 4.
10:0	RBS1	These bits indicate the first data buffer byte size, which must be a multiple of 4.

34.8.7 Receive Descriptor 2 (RDES2)

Bit	Name	Description
31:0	B1A	Set the address of the first buffer or the lower bits of the timestamp.

34.8.8 Receive Descriptor 3 (RDES3)

Bit	Name	Description
31:0	B2A	Set the address of the second buffer (or the address of the next descriptor) or the higher bits of the timestamp.

34.9 Operation Procedure

34.9.1 DMA Initialization

1. Operate the EMAC_CONFIG register to perform a software reset on the module.
2. Query the DMA_BUSMODE register until the reset operation is complete.
3. Configure the necessary bits in the DMA_BUSMODE register.
4. Create transmission descriptors in memory, ensuring that the receive descriptors are owned by EMAC-DMA. It is preferable to have multiple different descriptors before reusing the same descriptor.
5. Initialize the descriptor addresses in the DMA_TPD and DMA_RPD registers.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Configure the DMA_OPMODE register.
7. Configure the DMA_INTEN register to enable interrupts.
8. Start the transmission using the DMA_OPMODE register.

34.9.2 EMAC Initialization

1. Configure the PHY.
2. Configure the EMAC_ADDR0H and EMAC_ADDR0L registers. If additional addresses are required, configure the remaining address registers.
3. Configure the EMAC_HASHTABLEHIGH, EMAC_HASHTABLELOW, and EMAC_FRAMEFILTER registers to set the filtering method.
4. Configure the EMAC_FLOWCONTROL register.
5. Configure the EMAC_INTMASK register to select the required interrupts.
6. Configure the EMAC_CONFIG register and start the transmission.

34.9.3 Performing Transmission

1. Set the interrupt handler: check the interrupt status and the descriptor status.
2. Set the appropriate descriptor, ensuring that the descriptor is owned by DMA.
3. If the descriptor is not owned by DMA, the DMA will enter a suspended state. The transmission can be started using the TPD and RPD registers.

34.9.4 Stopping and Restarting Transmission

If it is required to pause the transmission for a period of time, follow these steps:

1. Clear DMA_OPMODE[13] to disable DMA transmit operations.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

2. Wait for the previous frame transmission to complete, then the debug register can be queried to check the status.
3. Clear EMAC_CONFIG[3:2].
4. Query the EMAC_DEBUG register; after all data in the RX FIFO has been transferred to the system memory, disable the receive DMA.
5. To restart the transmission, first enable the DMA, and then enable the EMAC transmission.

34.9.5 Guidelines for Using EEE Features

Entering and exiting LPI mode:

1. Check if the PHY supports EEE features.
2. Configure the PHY to use the EEE function.
3. Configure the EMAC_LPIT register.
4. Read the PHY status and update EMAC_LPICS[17].
5. Configure EMAC_LPICS[16] to make the EMAC enter LPI mode.
6. Clear EMAC_LPICS[16] to exit LPI mode.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35 Universal Serial Bus Full-speed Device Interface (USB)

35.1 Overview

The USB device controller is an interface compatible with the USB 2.0 full-speed protocol, which works in conjunction with USB PHY to realize communication between the chip and the USB HOST.

35.2 Main Features

- USB1.1 and USB2.0 full-speed protocols compliant
- 4 universal bidirectional transmission endpoints (EP1, EP2, EP3, EP4)
- Up to 64 bytes packet length of endpoint, supporting Memory and FIFO access modes
- FIFO mode supports 32-bit access
- 8-/16-/32-bit access in Memory access mode
- Suspend, wake-up and remote wake-up functions
- Toggle hardware comparison and software control functions
- Interrupt enabled for data transmission on each endpoint
- Interrupt enabled for bus reset, suspend and resume operations
- Optional NAK handshake packet for CRC error
- Data packet exceeding the maximum packet length (64 bytes) automatically answered with NAK
- NAK response from USB device for subsequent IN operations if the host does not return ACK

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

- Detection of lost token packets and data packets EOP, with optional automatic NAK response for lost EOPs

35.3 Functional Description

35.3.1 Interrupt Status and Control Registers

For each interrupt of the USB device, there are interrupt status register, interrupt enable register, and interrupt clear register, as detailed below:

- USB_INTSTATRAW: Raw interrupt status register that displays the interrupt occurrence status regardless of whether the interrupt is enabled.
- USB_INTEN: Interrupt enable register that enables a specific interrupt.
- USB_INTCLR: Interrupt clear register that is used to clear the interrupt occurrence flag. Just write 1 to clear the interrupt, and the register will be cleared automatically.

35.3.2 Address Setting

The Set Address command in control transfers can be entirely handled by hardware. Software can use the interrupt bit SETADDR to know that the Set Address command has occurred.

35.3.3 Remote Wakeup

The remote wakeup function is implemented through USB_WORKINGMODE[20]. Writing 1 to USB_WORKINGMODE[20] will send a K state on the USB port, with the duration controlled by software. Writing 0 to USB_WORKINGMODE[20] will stop the USB port from sending K states.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Before initiating remote wakeup, it is necessary to confirm whether the current USB PHY is enabled and whether the CLK48M clock generated by PHY is activated. Without the CLK48M clock, the controller will not be able to initiate remote wakeup.

35.3.4 Selective NAK Response for CRC Errors in Token and Data Packets

When CRC errors occur in token packets and data packets, the controller can selectively respond with NAK, controlled by USB_WORKINGMODE[10]. When USB_WORKINGMODE[10] is set to 1, the USB device will respond with an NAK handshake packet during the handshake phase in the event of a CRC error; otherwise, no handshake packet will be returned.

35.3.5 Packet Length Exceeding 64 Bytes

When the endpoint of the USB controller receives a data packet with a length exceeding 64 bytes, the status register USB_INTSTATRAW[24] will be set to 1. If USB_INTEN[24] is set to 1, an interrupt will be generated.

35.3.6 Packet Loss EOP

When both USB_WORKINGMODE[23] and USB_WORKINGMODE[24] are set to 1, if a token packet or data packet does not receive the EOP (end of packet) within the specified packet length, the status register USB_INTSTATRAW[30] will be set to 1. If USB_INTEN[30] is set to 1, an interrupt will be generated. The controller detects the token packet based on a length of 32 full-speed bits. If the token packet exceeds 32 full-speed bits without receiving an EOP (end of packet), this error will be triggered, and the device will reset its internal state machine, ensuring that the reception of the next packet is not affected. The data packet is detected based on the valid data length of 64 bytes + 8 bytes. If the actual length of the valid data in the packet exceeds

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

this value, this error will also be triggered.

When such errors occur in the token packet and data packet, it is possible to choose whether to respond to the USB host with an NAK handshake packet. This can be configured by setting USB_WORKINGMODE[21] and USB_WORKINGMODE[22] respectively. When USB_WORKINGMODE[21] is set to 1, upon the occurrence of such an error in the token packet, the USB controller will respond with an NAK handshake packet after the $(32 + 11)^{th}$ full-speed bit of the token packet. When USB_WORKINGMODE[22] is set to 1, upon the occurrence of such an error in the data packet, the USB controller will respond with an NAK handshake packet after the $(64\ bytes + 8\ bytes + 11\ bits)^{th}$ full-speed bit of the data packet.

35.3.7 IN Operation ACK Timeout, Responding NAK for Next IN Operation

When the host initiates an IN operation, if the ACK handshake packet that the host needs to respond with times out during the handshake phase, the EPx_IN_HANDSHAKE_ERR_RAW in the status register USB_INTSTATRAW will be set to 1 for different endpoints. When EPx_IN_HANDSHAKE_ERR_RAW is set to 1 and the host initiates another IN operation for that endpoint, the USB controller will respond with an NAK handshake packet until the software clears the EPx_IN_HANDSHAKE_ERR_RAW bit.

35.3.8 FIFO Access and Memory Access

This USB controller supports both FIFO access and memory access. FIFO access only supports 32-bit access, while memory access supports 8-, 16-, and 32-bit access.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.4 Register Description

USB register base address: 0x4090_0000

The registers are listed below:

Table 35-1: List of USB Registers

Offset	Name	Description
0X00	USB_WORKINGMODE	Working mode register
0X04	USB_EP0CSR	EP0 transfer control register
0X08	USB_EP1CSR	EP1 transfer control register
0X0C	USB_EP2CSR	EP2 transfer control register
0X10	USB_EP3CSR	EP3 transfer control register
0X14	USB_EP4CSR	EP4 transfer control register
0X18	USB_ADDR	USB address register
0X1C	USB_SETUP03DATA	SETUP data packet register 0
0X20	USB_SETUP47DATA	SETUP data packet register 1
0X24	USB_EPADDR	Endpoint address configuration register
0X28	USB_CURRENTPID	Current USB bus packet PID register
0X2C	USB_CURRENTFRAMENUMBER	Frame number register
0X30	USB_CRCERRORCNT	CRC error counter register
0X34	USB_STATUSDETECTCNT	Suspend/resume/reset detection time register
0X40	USB_EP0SENDBN	EP0 transmit data number register
0X44	USB_EP1SENDBN	EP1 transmit data number register
0X48	USB_EP2SENDBN	EP2 transmit data number register
0X4C	USB_EP3SENDBN	EP2 transmit data number register
0X50	USB_EP4SENDBN	EP4 transmit data number register
0X100	USB_EP0FIFO	EP0 FIFO access entry
0X104	USB_EP1FIFO	EP1 FIFO access entry
0X108	USB_EP2FIFO	EP2 FIFO access entry
0X10C	USB_EP3FIFO	EP3 FIFO access entry
0X110	USB_EP4FIFO	EP4 FIFO access entry
0X200–23C	USB_EP0MEM	EP0 memory access entry
0X240–27C	USB_EP1MEM	EP1 memory access entry
0X280–2BC	USB_EP2MEM	EP2 memory access entry
0X2C0–2FC	USB_EP3MEM	EP3 memory access entry

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset	Name	Description
0X300–33C	USB_EP4MEM	EP4 memory access entry
0XFFE4	USB_INTSTATRAW	Status register
0XFFE8	USB_INTEN	Interrupt enable register
0XFFF0	USB_INTCLR	Interrupt clear register

35.4.1 Working Mode Register (USB_WORKINGMODE)

Offset address: 0x00

Reset value: 0x0000 0009

Bit	Name	Attribute	Reset Value	Description
31:26	RSV	-	-	Reserved
25	USB_EP0_ZOD_INTR_EN	R/W	0x0	Interrupt generation enable for 0-length data packet in USB OUT operation: 1: Enable this interrupt. 0: Disable this interrupt.
24	USB_DATA_NOEOP_EN	R/W	0x0	When the USB controller receives a data packet longer than 64 + 8 bytes, it will consider this data packet as losing the EOP, and the controller will reset the state machine internally and generate an error state. 1: Enable this function. 0: Disable this function.
23	USB_TOKEN_NOEOP_EN	R/W	0x0	When the USB controller receives a token packet that exceeds the specified length, it will consider this token packet as losing the EOP, and the controller will reset the state machine internally and generate an error state. 1: Enable this function. 0: Disable this function.
22	USB_DATA_NOEOP_NAK_EN	R/W	0x0	When the USB controller receives a data packet longer than 64 + 8 bytes,

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				it will consider this data packet as losing the EOP. When this bit is set to 1, the controller will reply with an NAK handshake packet to the host. 1: Reply with NAK. 0: Do not reply with NAK. This function is valid only when USB_DATA_NOEOP_EN is set to 1.
21	USB_TOKEN_NOEOP_NAK_EN	R/W	0x0	When the USB controller receives a token packet that exceeds the specified length, it will consider this token packet as losing the EOP. When this bit is set to 1, the controller will reply with an NAK handshake packet to the host. 1: Reply with NAK. 0: Do not reply with NAK. This function is valid only when USB_TOKEN_NOEOP_EN is set to 1.
20	USB_REMOTE_WAKEUP	R/W	0x0	USB remote wakeup control bit: 1: The controller sends K state. 0: The controller does not send K state. Upon completion of the wakeup, this bit shall be cleared by software writing 0.
19:13	RSV	-	-	Reserved
12	USB_MORETHAN64_NAK_EN	R/W	0x0	Enable reply with NAK handshake packet for receiving a data packet longer than 64 bytes: 1: Enable this function. 0: Disable this function.
11	USB_IN_TIMEOUT_NAK_EN	R/W	0x0	After the IN operation data packet is sent successfully, if the controller does not receive the ACK handshake packet from the host, the device will reply with NAK for the next IN

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				operation on that endpoint, until the software clears the error status bit. 1: Enable this function. 0: Disable this function.
10	USB_CRCERR_NAK_EN	R/W	0x0	Enable NAK response for CRC errors in token and data packets: 1: Enable this function. 0: Disable this function.
9:8	USB_LINE_STATE	R	0x0	USB DP/DM signal status bit: Bit 8: DM Bit 9: DP
7	RSV	-	-	Reserved
6	USB_DPPU_LO	R/W	0x0	USB controller DP signal 2.8k pull-up resistor control bit: 1: Pull-up enabled 0: Pull-up disabled
5	RSV	-	-	Reserved
4	USB_DPPU	R/W	0x0	USB controller DP signal 2.1k pull-up resistor control bit: 1: Enable pull-up. 0: Disable pull-up.
3	USB_BUS_AUTO_RST_EN	R/W	0x1	Enable USB bus reset of controller address, transmit/receive state machine and TX/RX FIFO: 1: Enable the reset. 0: Disable the reset.
2	USB_FORCE_RST	R/W	0x0	Reset controller address, transmit/receive state machine and TX/RX FIFO: 1: Enable the reset. 0: Disable the reset. After each reset operation, the software shall write 0 to clear this bit.
1	USB_SUSPEND	R/W	0x0	1: Set USB PHY to be in suspended state. 0: Clear the suspended state of USB PHY.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	SPEED_MODE	R/W	0x1	USB operation mode selection bit: 1: Full-speed mode 0: Low-speed mode This chip only supports full-speed mode.

35.4.2 EP0 Transfer Control Register (USB_EP0CSR)

Offset address: 0x04

Reset value: 0x0000 0100

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	EP0_RECEIVED_ACK	R	0x0	Flag of EP0 receiving the ACK handshake packet from the host: 1: Handshake packet received 0: Handshake packet not received This bit is read-only and can be cleared by USB reset or writing 1 to EP0_DATA_START.
23:20	RSV	-	-	Reserved
19	EP0_OUT_TOGGLE_STATE	R	0x0	EP0 out/setup status error flag: 1: Toggle received is not as expected. 0: Toggle is normal.
18	EP0_OUT_TOGGLE_CTRL_En	W	0x0	Enable the value in EP0_OUT_TOGGLE_WANT to take effect: 1: The value written to EP0_OUT_TOGGLE_WANT is effective.
17	EP0_OUT_TOGGLE_WANT	R/W	0x0	Comparison value of toggle for EP0 out/setup packets: 1: Data 1 0: Data 0
16	EP0_OUT_TOGGLE_VALUE	R	0x0	Toggle value of packets received by EP0: 1: Data 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Data 0
15	EP0_IN_TOGGLE_CTRL_EN	W	0x0	Enable the value in EP0_IN_TOGGLE_VALUE to take effect: 1: The value written to EP0_IN_TOGGLE_VALUE is effective.
14	EP0_IN_TOGGLE_VALUE	R/W	0x0	IN operation toggle control bit: 1: Data 1 0: Data 0 Writing this bit will change the comparison register value, reading this bit returns the current IN toggle value.
13	EP0_SEND_STALL_DONE	R/W	0x0	EP0 STALL transmission complete flag: 1: STALL transmission is completed. 0: STALL transmission is not completed. Writing 1 clears this bit.
12	EP0_SEND_STALL	W	0x0	STALL transmission control bit: 1: STALL is transmitted. 0: STALL is not transmitted. A single write operation transmits STALL only once.
11	EP0_RECEIVED_DONE	W	0x0	Reception complete control bit: 1: Reception is completed. 0: Reception is not completed. Writing 1 to this bit will set the FIFO to ready state, and it is necessary to write a 1 to this bit each time the data is received and read, otherwise, the device will respond with NAK during the next OUT/SETUP operation.
10	EP0_DATA_START	W	0x0	Transmit START: 1: Transmit the data in FIFO. 0: No operation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
9	EP0_FIFOCLR	W	0x0	EP0 FIFO pointer reset control bit: 1: Reset FIFO pointer. 0: Do not reset FIFO pointer.
8	EP0_EN	R/W	0x1	EP0 enable: 1: Enabled EP0. 0: Disable EP0.
7:0	EP0_RECEIVED_BYTE	R	0x0	Number of data bytes received by EP0: This value must be less than the maximum packet length.

35.4.3 EP1 Transfer Control Register (USB_EP1CSR)

Offset address: 0x08

Reset value: 0x0000 0100

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	EP1_RECEIVED_ACK	R	0x0	Flag of EP1 receiving the ACK handshake packet from the host: 1: Handshake packet received 0: Handshake packet not received This bit is read-only and can be cleared by USB reset or writing 1 to EP1_DATA_START.
23:20	RSV	-	-	Reserved
19	EP1_OUT_TOGGLE_STATE	R	0x0	EP1 out/setup status error flag: 1: Toggle received is not as expected. 0: Toggle is normal.
18	EP1_OUT_TOGGLE_CTRL_En	W	0x0	Enable the value in EP1_OUT_TOGGLE_WANT to take effect: 1: The value written to EP1_OUT_TOGGLE_WANT is effective.
17	EP1_OUT_TOGGLE_WANT	R/W	0x0	Comparison value of toggle for EP1 out/setup packets:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Data 1 0: Data 0
16	EP1_OUT_TOGGLE_VALUE	R	0x0	Toggle value of packets received by EP1: 1: Data 1 0: Data 0
15	EP1_IN_TOGGLE_CTRL_EN	W	0x0	Enable the value in EP1_IN_TOGGLE_VALUE to take effect: 1: The value written to EP1_IN_TOGGLE_VALUE is effective.
14	EP1_IN_TOGGLE_VALUE	R/W	0x0	IN operation toggle control bit: 1: Data 1 0: Data 0 Writing this bit will change the comparison register value, reading this bit returns the current IN toggle value.
13	EP1_SEND_STALL_DONE	R	0x0	EP1 STALL transmission complete flag: 1: STALL transmission is completed. 0: STALL transmission is not completed. Writing 1 clears this bit.
12	EP1_SEND_STALL	R/W	0x0	STALL transmission control bit: 1: STALL is transmitted. 0: STALL is not transmitted. A single write operation transmits STALL multiple times until this bit is cleared.
11	EP1_RECEIVED_DONE	W	0x0	Reception complete control bit: 1: Reception is completed. 0: Reception is not completed. Writing 1 to this bit will set the FIFO to ready state, and it is necessary to write a 1 to this bit each time the data is received and read, otherwise, the

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				device will respond with NAK during the next OUT/SETUP operation.
10	EP1_DATA_START	W	0x0	Transmit START: 1: Transmit the data in FIFO. 0: No operation.
9	EP1_FIFOCLR	W	0x0	EP1 FIFO pointer reset control bit: 1: Reset FIFO pointer. 0: Do not reset FIFO pointer.
8	EP1_EN	R/W	0x1	EP1 enable: 1: Enable EP1. 0: Disable EP1.
7:0	EP1_RECEIVED_BYTE	R	0x0	Number of data bytes received by EP1: This value must be less than the maximum packet length.

35.4.4 EP2 Transfer Control Register (USB_EP2CSR)

Offset address: 0x0C

Reset value: 0x0000 0100

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	EP2_RECEIVED_ACK	R	0x0	Flag of EP2 receiving the ACK handshake packet from the host: 1: Handshake packet is received. 0: Handshake packet is not received. This bit is read-only and can be cleared by USB reset or writing 1 to EP2_DATA_START.
23:20	-	-	-	-
19	EP2_OUT_TOGGLE_STATE	R	0x0	EP2 out/setup status error flag: 1: Toggle received is not as expected. 0: Toggle is normal.

Unicmicro

Bit	Name	Attribute	Reset Value	Description
18	EP2_OUT_TOGGLE_CTRL_En	W	0x0	Enable the value in EP2_OUT_TOGGLE_WANT to take effect: 1: The value written to EP2_OUT_TOGGLE_WANT is effective.
17	EP2_OUT_TOGGLE_WANT	R/W	0x0	Comparison value of toggle for EP2 out/setup packets: 1: Data 1 0: Data 0
16	EP2_OUT_TOGGLE_VALUE	R	0x0	Toggle value of packets received by EP2: 1: Data 1 0: Data 0
15	EP2_IN_TOGGLE_CTRL_EN	W	0x0	Enable the value in EP2_IN_TOGGLE_VALUE to take effect: 1: The value written to EP2_IN_TOGGLE_VALUE is effective.
14	EP2_IN_TOGGLE_VALUE	R/W	0x0	IN operation toggle control bit: 1: Data 1 0: Data 0 Writing this bit will change the comparison register value, reading this bit returns the current IN toggle value.
13	EP2_SEND_STALL_DONE	R	0x0	EP2 STALL transmission complete flag: 1: STALL transmission is completed. 0: STALL transmission is not completed. Writing 1 clears this bit.
12	EP2_SEND_STALL	R/W	0x0	STALL transmission control bit: 1: STALL is transmitted. 0: STALL is not transmitted. A single write operation transmits STALL multiple times until this bit is

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				cleared.
11	EP2_RECEIVED_DONE	W	0x0	Reception complete control bit: 1: Reception is completed. 0: Reception is not completed. Writing 1 to this bit will set the FIFO to ready state, and it is necessary to write a 1 to this bit each time the data is received and read, otherwise, the device will respond with NAK during the next OUT/SETUP operation.
10	EP2_DATA_START	W	0x0	Transmit START: 1: Transmit the data in FIFO. 0: No operation.
9	EP2_FIFOCLR	W	0x0	EP2 FIFO pointer reset control bit: 1: Reset FIFO pointer. 0: Do not reset FIFO pointer.
8	EP2_EN	R/W	0x1	EP2 enable: 1: Enable EP2. 0: Disable EP2.
7:0	EP2_RECEIVED_BYTE	R	0x0	Number of data bytes received by EP2: This value must be less than the maximum packet length.

35.4.5 EP3 Transfer Control Register (USB_EP3CSR)

Offset address: 0x10

Reset value: 0x0000 0100

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	EP3_RECEIVED_ACK	R	0x0	Flag of EP3 receiving the ACK handshake packet from the host: 1: Handshake packet is received. 0: Handshake packet is not received. This bit is read-only and can be cleared by USB reset or writing 1 to

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				EP3_DATA_START.
23:20	-	-	-	-
19	EP3_OUT_TOGGLE_STATE	R	0x0	EP3 out/setup status error flag: 1: Toggle received is not as expected. 0: Toggle is normal.
18	EP3_OUT_TOGGLE_CTRL_En	W	0x0	Enable the value in EP3_OUT_TOGGLE_WANT to take effect: 1: The value written to EP3_OUT_TOGGLE_WANT is effective.
17	EP3_OUT_TOGGLE_WANT	R/W	0x0	Comparison value of toggle for EP3 out/setup packets: 1: Data 1 0: Data 0
16	EP3_OUT_TOGGLE_VALUE	R	0x0	Toggle value of packets received by EP3: 1: Data 1 0: Data 0
15	EP3_IN_TOGGLE_CTRL_EN	W	0x0	Enable the value in EP3_IN_TOGGLE_VALUE to take effect: 1: The value written to EP3_IN_TOGGLE_VALUE is effective.
14	EP3_IN_TOGGLE_VALUE	R/W	0x0	IN operation toggle control bit: 1: Data 1 0: Data 0 Writing this bit will change the comparison register value, reading this bit returns the current IN toggle value.
13	EP3_SEND_STALL_DONE	R	0x0	EP3 STALL transmission complete flag: 1: STALL transmission is completed. 0: STALL transmission is not completed. Writing 1 clears this bit.
12	EP3_SEND_ST	R/W	0x0	STALL transmission control bit:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
	ALL			1: STALL is transmitted. 0: STALL is not transmitted. A single write operation transmits STALL multiple times until this bit is cleared.
11	EP3_RECEIVED_DONE	W	0x0	Reception complete control bit: 1: Reception is completed. 0: Reception is not completed. Writing 1 to this bit will set the FIFO to ready state, and it is necessary to write a 1 to this bit each time the data is received and read, otherwise, the device will respond with NAK during the next OUT/SETUP operation.
10	EP3_DATA_START	W	0x0	Transmit START: 1: Transmit the data in FIFO. 0: No operation.
9	EP3_FIFOCLR	W	0x0	EP3 FIFO pointer reset control bit: 1: Reset FIFO pointer. 0: Do not reset FIFO pointer.
8	EP3_EN	R/W	0x1	EP3 enable: 1: Enable EP3. 0: Disable EP3.
7:0	EP3_RECEIVED_BYTE	R	0x0	Number of data bytes received by EP3: This value must be less than the maximum packet length.

35.4.6 EP4 Transfer Control Register (USB_EP4CSR)

Offset address: 0x14

Reset value: 0x0000 0100

Bit	Name	Attribute	Reset Value	Description
31:25	RSV	-	-	Reserved
24	EP4_RECEIVED_ACK	R	0x0	Flag of EP4 receiving the ACK handshake packet from the host: 1: Handshake packet is received.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Handshake packet is not received. This bit is read-only and can be cleared by USB reset or writing 1 to EP4_DATA_START.
23:20	-	-	-	-
19	EP4_OUT_TOGGLE_STATE	R	0x0	EP4 out/setup status error flag: 1: Toggle received is not as expected. 0: Toggle is normal.
18	EP4_OUT_TOGGLE_CTRL_En	W	0x0	Enable the value in EP4_OUT_TOGGLE_WANT to take effect: 1: The value written to EP4_OUT_TOGGLE_WANT is effective.
17	EP4_OUT_TOGGLE_WANT	R/W	0x0	Comparison value of toggle for EP4 out/setup packets: 1: Data 1 0: Data 0
16	EP4_OUT_TOGGLE_VALUE	R	0x0	Toggle value of packets received by EP4: 1: Data 1 0: Data 0
15	EP4_IN_TOGGLE_CTRL_EN	W	0x0	Enable the value in EP4_IN_TOGGLE_VALUE to take effect: 1: The value written to EP4_IN_TOGGLE_VALUE is effective.
14	EP4_IN_TOGGLE_VALUE	R/W	0x0	IN operation toggle control bit: 1: Data 1 0: Data 0 Writing this bit will change the comparison register value, reading this bit returns the current IN toggle value.
13	EP4_SEND_STALL_DONE	R	0x0	EP4 STALL transmission complete flag: 1: STALL transmission is completed. 0: STALL transmission is not

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				completed. Writing 1 clears this bit.
12	EP4_SEND_STALL	R/W	0x0	STALL transmission control bit: 1: STALL is transmitted. 0: STALL is not transmitted. A single write operation transmits STALL multiple times until this bit is cleared.
11	EP4_RECEIVED_DONE	W	0x0	Reception complete control bit: 1: Reception is completed. 0: Reception is not completed. Writing 1 to this bit will set the FIFO to ready state, and it is necessary to write a 1 to this bit each time the data is received and read, otherwise, the device will respond with NAK during the next OUT/SETUP operation.
10	EP4_DATA_START	W	0x0	Transmit START: 1: Transmit the data in FIFO. 0: No operation.
9	EP4_FIFOCLR	W	0x0	EP4 FIFO pointer reset control bit: 1: Reset FIFO pointer. 0: Do not reset FIFO pointer.
8	EP4_EN	R/W	0x1	EP4 enable: 1: Enable EP4. 0: Disable EP4.
7:0	EP4_RECEIVED_BYTE	R	0x0	Number of data bytes received by EP4: This value must be less than the maximum packet length.

35.4.7 USB Address Register (USB_ADDR)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6:0	USB_ADDR	R/W	0x0	USB address register (read-only)

35.4.8 SETUP Data Packet Register (USB_SETUP03DATA)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SETUP_0_3_DATA	R	0x0	SETUP data packet byte 0–byte 3 register

35.4.9 SETUP Data Packet Register (USB_SETUP47DATA)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	SETUP_4_7_DATA	R	0x0	SETUP data packet byte 4–byte 7 register

35.4.10 Endpoint Address Configuration Register (USB_EPADDR)

Offset address: 0x24

Reset value: 0x0000 4321

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	EP4_ADDR	R/W	0x4	EP4 address configuration
11:8	EP3_ADDR	R/W	0x3	EP3 address configuration
7:4	EP2_ADDR	R/W	0x2	EP2 address configuration
3:0	EP1_ADDR	R/W	0x1	EP1 address configuration

35.4.11 Bus Packet PID Register (USB_CURRENTPID)

Offset address: 0x28

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:4	RSV	-	-	Reserved
3:0	CURRENT_PID	R	0x0	Current PID value of the received USB packet

35.4.12 Frame Number Register (USB_CURRENTFRAMENUMBER)

Offset address: 0x2C

Reset value: 0x0000 003F

Bit	Name	Attribute	Reset Value	Description
31:11	RSV	-	-	Reserved
10:0	CURRENT_FRAME_NUMBER	R	0x3F	Current frame number

35.4.13 CRC Error Counter Register (USB_CRCERRORCNT)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	CRC_ERROR_CNT	R	0x0	Number of CRC error packets, reset during USB reset

35.4.14 Detection Time Register (USB_STATUSDETECTCNT)

Offset address: 0x34

Reset value: 0x0000 00FF

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	USB_STATUS_DETECT_CNT	R/W	0xFF	Reset/resume/suspend detection threshold setting: Each unit is 2.5 μs (120 clock cycles at 48MHz). 0x0: The threshold is 2.5 μs. 0x1: The threshold is 5.0 μs. …..

35.4.15 EP0 Transmit Data Byte Number Register (USB_EP0SENDBN)

Offset address: 0x40

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	EP0_SEND_BYTE	R/W	0x0	EP0 transmit data byte number register

35.4.16 EP1 Transmit Data Byte Number Register (USB_EP1SENDBN)

Offset address: 0x44

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	EP1_SEND_BYTE	R/W	0x0	EP1 transmit data byte number register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.4.17 EP2 Transmit Data Byte Number Register (USB_EP2SENDBN)

Offset address: 0x48

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	EP2_SEND_BYTE	R/W	0x0	EP2 transmit data byte number register

35.4.18 EP3 Transmit Data Byte Number Register (USB_EP3SENDBN)

Offset address: 0x4C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	EP3_SEND_BYTE	R/W	0x0	EP3 transmit data byte number register

35.4.19 EP4 Transmit Data Byte Number Register (USB_EP4SENDBN)

Offset address: 0x50

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	EP4_SEND_BYTE	R/W	0x0	EP4 transmit data byte number register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.4.20 EP0 FIFO Access Entry (USB_EP0FIFO)

Offset address: 0x100

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	EP0FIFO	R/W	0x0	EP0 FIFO entry address, only 32-bit access is supported.

35.4.21 EP1 FIFO Access Entry (USB_EP1FIFO)

Offset address: 0x104

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	EP1FIFO	R/W	0x0	EP1 FIFO entry address, only 32-bit access is supported.

35.4.22 EP2 FIFO Access Entry (USB_EP2FIFO)

Offset address: 0x108

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	EP2FIFO	R/W	0x0	EP2 FIFO entry address, only 32-bit access is supported.

35.4.23 EP3 FIFO Access Entry (USB_EP3FIFO)

Offset address: 0x10C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	EP3FIFO	R/W	0x0	EP3 FIFO entry address, only 32-bit access is supported.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.4.24 EP4 FIFO Access Entry (USB_EP4FIFO)

Offset address: 0x110

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	EP4FIFO	R/W	0x0	EP4 FIFO entry address, only 32-bit access is supported.

35.4.25 Status Register (USB_INTSTATRAW)

Offset address: 0xFFE4

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	TOGGLE_STATE_ERR_RAW	R	0x0	When a toggle error occurs during IN/OUT/setup operations, this status bit is set to 1.
30	NOEOP_ERR_RAW	R	0x0	If the length of the token packet received by the device exceeds the specified value in the protocol, or if the data packet exceeds (64 + 8) bytes, this status bit will be set to 1.
29	EP4_IN_HANDSHAKE_ERR_RAW	R	0x0	For EP4 IN operations, if the host does not successfully return an ACK signal, this bit will be set to 1.
28	EP3_IN_HANDSHAKE_ERR_RAW	R	0x0	For EP3 IN operations, if the host does not successfully return an ACK signal, this bit will be set to 1.
27	EP2_IN_HANDSHAKE_ERR_RAW	R	0x0	For EP2 IN operations, if the host does not successfully return an ACK signal, this bit will be set to 1.
26	EP1_IN_HANDSHAKE_ERR_RAW	R	0x0	For EP1 IN operations, if the host does not successfully return an ACK signal, this bit will be set to 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
25	EP0_IN_HANDSHAKE_ERR_RAW	R	0x0	For EP0 IN operations, if the host does not successfully return an ACK signal, this bit will be set to 1.
24	DATA_BYTE_MORETHAN_64_RAW	R	0x0	If the length of the received data packet exceeds 64 bytes, this bit will be set to 1.
23	CRC_ERR_RAW	R	0x0	If there is a CRC error in the received token packet or data packet, this bit will be set to 1.
22	SETADDR_RAW	R	0x0	When the host successfully sets the USB device address, this bit will be set to 1.
21	TURNAROUND_ERROR_RAW	R	0x0	A timeout interrupt occurs when the host replies with an ACK packet.
20	EP4_ACK_RAW	R	0x0	EP4 ACK status: When transmitting or receiving an ACK packet, this bit will be set to 1.
19	EP4_OUT_RAW	R	0x0	EP4 OUT interrupt: When the valid data enters the FIFO, this bit will be set to 1.
18	EP4_IN_RAW	R	0x0	When EP4 receives an IN token packet, this bit will be set to 1.
17	EP3_ACK_RAW	R	0x0	EP3 ACK status: When transmitting or receiving an ACK packet, this bit will be set to 1.
16	EP3_OUT_RAW	R	0x0	EP3 OUT interrupt: When the valid data enters the FIFO, this bit will be set to 1.
15	EP3_IN_RAW	R	0x0	When EP3 receives an IN token packet, this bit will be set to 1.
14	EP2_ACK_RAW	R	0x0	EP2 ACK status: When transmitting or receiving an ACK packet, this bit will be set to 1.
13	EP2_OUT_RAW	R	0x0	EP2 OUT interrupt: When the valid data enters the FIFO, this bit will be set to 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	EP2_IN_RAW	R	0x0	When EP2 receives an IN token packet, this bit will be set to 1.
11	EP1_ACK_RAW	R	0x0	EP1 ACK status: When transmitting or receiving an ACK packet, this bit will be set to 1.
10	EP1_OUT_RAW	R	0x0	EP1 OUT interrupt: When the valid data enters the FIFO, this bit will be set to 1.
9	EP1_IN_RAW	R	0x0	When EP1 receives an IN token packet, this bit will be set to 1.
8	EP0_ACK_RAW	R	0x0	EP0 ACK status: When transmitting or receiving an ACK packet, this bit will be set to 1.
7	EP0_OUT_RAW	R	0x0	EP0 OUT interrupt: When the valid data enters the FIFO, this bit will be set to 1.
6	EP0_IN_RAW	R	0x0	When EP0 receives an IN token packet, this bit will be set to 1.
5	SUDAV_RAW	R	0x0	When a setup data packet is received, this bit will be set to 1.
4	SETUPTOK_RAW	R	0x0	When a setup token packet is received, this bit will be set to 1.
3	SOF_RAW	R	0x0	When a SOF packet is received, this bit will be set to 1.
2	RESUME_RAW	R	0x0	When the host resumes, this bit will be set to 1.
1	SUSPEND_RAW	R	0x0	When the host suspends, this bit will be set to 1.
0	BUS_RESET_RAW	R	0x0	When the host resets, this bit will be set to 1.

35.4.26 Interrupt Enable Register (USB_INTEN)

Offset address: 0xFFE8

Reset value: 0x0000 0000

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31	TOGGLE_STATE_ERR_EN	R/W	0x0	TOGGLE_STATE_ERR interrupt enable
30	NOEOP_ERR_EN	R/W	0x0	NOEOP_ERR interrupt enable
29	EP4_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP4_IN_HANDSHAKE_ERR interrupt enable
28	EP3_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP3_IN_HANDSHAK _ERR interrupt enable
27	EP2_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP2_IN_HANDSHAKE_ERR interrupt enable
26	EP1_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP1_IN_HANDSHAKE_ERR interrupt enable
25	EP0_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP0_IN_HANDSHAKE_ERR interrupt enable
24	DATA_BYTE_MORETHAN_64_EN	R/W	0x0	Enable interrupt for DATA packet length exceeding 64 bytes
23	CRC_ERR_EN	R/W	0x0	CRC error interrupt enable
22	SETADDR_EN	R/W	0x0	Enable interrupt when the host sets the USB device address successfully
21	TURNAROUND_ERROR_EN	R/W	0x0	Enable interrupt for host ACK packet timeout
20	EP4_ACK_EN	R/W	0x0	Enable interrupt for EP4 ACK status: Transmitting or receiving ACK packets
19	EP4_OUT_EN	R/W	0x0	EP4 OUT interrupt is triggered when the valid data enters the FIFO.
18	EP4_IN_EN	R/W	0x0	Enable interrupt for EP4 receiving an IN token packet
17	EP3_ACK_EN	R/W	0x0	Enable interrupt for EP3 ACK status: Transmitting or receiving ACK packets
16	EP3_OUT_EN	R/W	0x0	EP3 OUT interrupt is triggered when the valid data enters the FIFO.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15	EP3_IN_EN	R/W	0x0	Enable interrupt for EP3 receiving an IN token packet
14	EP2_ACK_EN	R/W	0x0	Enable interrupt for EP2 ACK status: Transmitting or receiving ACK packets
13	EP2_OUT_EN	R/W	0x0	EP2 OUT interrupt is triggered when the valid data enters the FIFO.
12	EP2_IN_EN	R/W	0x0	Enable interrupt for EP2 receiving an IN token packet
11	EP1_ACK_EN	R/W	0x0	Enable interrupt for EP1 ACK status: Transmitting or receiving ACK packets
10	EP1_OUT_EN	R/W	0x0	EP1 OUT interrupt is triggered when the valid data enters the FIFO.
9	EP1_IN_EN	R/W	0x0	Enable interrupt for EP1 receiving an IN token packet
8	EP0_ACK_EN	R/W	0x0	Enable interrupt for EP0 ACK status: Transmitting or receiving ACK packets
7	EP0_OUT_EN	R/W	0x0	EP0 OUT interrupt is triggered when the valid data enters the FIFO.
6	EP0_IN_EN	R/W	0x0	Enable interrupt for EP0 receiving an IN token packet
5	SUDAV_EN	R/W	0x0	Enable interrupt for receiving a setup data packet
4	SETUPTOK_EN	R/W	0x0	Enable interrupt for receiving a setup token packet
3	SOF_EN	R/W	0x0	Enable interrupt for receiving a SOF packet
2	RESUME_EN	R/W	0x0	Host resume interrupt enable

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	SUSPEND_EN	R/W	0x0	Host suspend interrupt enable
0	BUS_RESET_EN	R/W	0x0	Host reset interrupt enable

35.4.27 Interrupt Clear Register (USB_INTCLR)

Offset address: 0xFFF0

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31	TOGGLE_STATE_ERR_CLR	R/W	0x0	TOGGLE_STATE_ERR _RAW interrupt register clear bit: 1: Clear TOGGLE_STATE_ERR_RAW to 0. 0: Keep TOGGLE_STATE_ERR_RAW unchanged.
30	NOEOP_ERR_CLR	R/W	0x0	NOEOP_ERR_RAW interrupt register clear bit: 1: Clear NOEOP_ERR_RAW to 0. 0: Keep NOEOP_ERR_RAW unchanged.
29	EP4_IN_HANDSHAKE_ERR_CLR	R/W	0x0	EP4_IN_HANDSHAKE_ERR_RAW interrupt register clear bit: 1: Clear EP4_IN_HANDSHAKE_ERR_RAW to 0. 0: Keep EP4_IN_HANDSHAKE_ERR_RAW unchanged.
28	EP3_IN_HANDSHAKE_ERR_CLR	R/W	0x0	EP3_IN_HANDSHAKE_ERR_RAW interrupt register clear bit: 1: Clear EP3_IN_HANDSHAKE_ERR_RAW to 0. 0: Keep EP3_IN_HANDSHAKE_ERR_RAW unchanged.
27	EP2_IN_HANDSHAKE_ERR_CLR	R/W	0x0	EP2_IN_HANDSHAKE_ERR_RAW interrupt register clear bit:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Clear EP2_IN_HANDSHAKE_ERR_RAW to 0. 0: Keep EP2_IN_HANDSHAKE_ERR_RAW unchanged.
26	EP1_IN_HANDSHAKE_ERR_CLR	R/W	0x0	EP1_IN_HANDSHAKE_ERR_RAW interrupt register clear bit: 1: Clear EP1_IN_HANDSHAKE_ERR_RAW to 0. 0: Keep EP1_IN_HANDSHAKE_ERR_RAW unchanged.
25	EP0_IN_HANDSHAKE_ERR_CLR	R/W	0x0	EP0_IN_HANDSHAKE_ERR_RAW interrupt register clear bit: 1: Clear EP0_IN_HANDSHAKE_ERR_RAW to 0. 0: Keep EP0_IN_HANDSHAKE_ERR_RAW unchanged.
24	DATA_BYTE_MORETHAN_64_CLR	R/W	0x0	DATA_BYTE_MORETHAN_64_RAW interrupt register clear bit: 1: Clear DATA_BYTE_MORETHAN_64_RAW to 0. 0: Keep DATA_BYTE_MORETHAN_64_RAW unchanged.
23	CRC_ERR_CLR	R/W	0x0	CRC_ERR_RAW interrupt register clear bit: 1: Clear CRC_ERR_RAW to 0. 0: Keep CRC_ERR_RAW unchanged.
22	SETADDR_CLR	R/W	0x0	SETADDR_RAW interrupt register clear bit: 1: Clear SETADDR_RAW to 0. 0: Keep SETADDR_RAW unchanged.
21	TURNAROUND_ERROR_CLR	R/W	0x0	TURNAROUND_ERROR_RAW interrupt register clear bit:

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Clear TURNAROUND_ERROR_RAW to 0. 0: Keep TURNAROUND_ERROR_RAW unchanged.
20	EP4_ACK_CLR	R/W	0x0	EP4_ACK_RAW interrupt register clear bit: 1: Clear EP4_ACK_RAW to 0. 0: Keep EP4_ACK_RAW unchanged.
19	EP4_OUT_CLR	R/W	0x0	EP4_OUT_RAW interrupt register clear bit: 1: Clear EP4_OUT_RAW to 0. 0: Keep EP4_OUT_RAW unchanged.
18	EP4_IN_CLR	R/W	0x0	EP4_IN_RAW interrupt register clear bit: 1: Clear EP4_IN_RAW to 0. 0: Keep EP4_IN_RAW unchanged.
17	EP3_ACK_CLR	R/W	0x0	EP3_ACK_RAW interrupt register clear bit: 1: Clear EP3_ACK_RAW to 0. 0: Keep EP3_ACK_RAW unchanged.
16	EP3_OUT_CLR	R/W	0x0	EP3_OUT_RAW interrupt register clear bit: 1: Clear EP3_OUT_RAW to 0. 0: Keep EP3_OUT_RAW unchanged.
15	EP3_IN_CLR	R/W	0x0	EP3_IN_RAW interrupt register clear bit: 1: Clear EP3_IN_RAW to 0. 0: Keep EP3_IN_RAW unchanged.
14	EP2_ACK_CLR	R/W	0x0	EP2_ACK_RAW interrupt register clear bit: 1: Clear EP2_ACK_RAW to 0. 0: Keep EP2_ACK_RAW unchanged.
13	EP2_OUT_CLR	R/W	0x0	EP2_OUT_RAW interrupt register clear bit: 1: Clear EP2_OUT_RAW to 0. 0: Keep EP2_OUT_RAW unchanged.
12	EP2_IN_CLR	R/W	0x0	EP2_IN_RAW interrupt register clear

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				bit: 1: Clear EP2_IN_RAW to 0. 0: Keep EP2_IN_RAW unchanged.
11	EP1_ACK_CLR	R/W	0x0	EP1_ACK_RAW interrupt register clear bit: 1: Clear EP1_ACK_RAW to 0. 0: Keep EP1_ACK_RAW unchanged.
10	EP1_OUT_CLR	R/W	0x0	EP1_OUT_RAW interrupt register clear bit: 1: Clear EP1_OUT_RAW to 0. 0: Keep EP1_OUT_RAW unchanged.
9	EP1_IN_CLR	R/W	0x0	EP1_IN_RAW interrupt register clear bit: 1: Clear EP1_IN_RAW to 0. 0: Keep EP1_IN_RAW unchanged.
8	EP0_ACK_CLR	R/W	0x0	EP0_ACK_RAW interrupt register clear bit: 1: Clear EP0_ACK_RAW to 0. 0: Keep EP0_ACK_RAW unchanged.
7	EP0_OUT_CLR	R/W	0x0	EP0_OUT_RAW interrupt register clear bit: 1: Clear EP0_OUT_RAW to 0. 0: Keep EP0_OUT_RAW unchanged.
6	EP0_IN_CLR	R/W	0x0	EP0_IN_RAW interrupt register clear bit: 1: Clear EP0_IN_RAW to 0. 0: Keep EP0_IN_RAW unchanged.
5	SUDAV_CLR	R/W	0x0	Receiving setup data packet interrupt register clear bit: 1: Clear SUDAV_RAW to 0. 0: Keep SUDAV_RAW unchanged.
4	SETUPTOK_CLR	R/W	0x0	Receiving setup token packet interrupt register clear bit: 1: Clear SETUPTOK_RAW to 0. 0: Keep SETUPTOK_RAW unchanged.
3	SOF_CLR	R/W	0x0	Receiving SOF packet interrupt register clear bit:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Clear SOF_RAW to 0. 0: Keep SOF_RAW unchanged.
2	RESUME_CLR	R/W	0x0	Host resume interrupt register clear bit: 1: Clear RESUME_RAW to 0. 0: Keep RESUME_RAW unchanged.
1	SUSPEND_CLR	R/W	0x0	Host suspend interrupt register clear bit: 1: Clear SUSPEND_RAW to 0. 0: Keep SUSPEND_RAW unchanged.
0	BUS_RESET_CLR	R/W	0x0	Host reset interrupt register clear bit: 1: Clear BUS_RESET_RAW to 0. 0: Keep BUS_RESET_RAW unchanged.

35.5 Operation Procedure

35.5.1 USB Connection

By default, USB is not connected. After completing the initialization, the USB connection can be established by setting USB_WORKINGMODE[4] and USB_WORKINGMODE[6]. Conversely, the USB connection can be disconnected by clearing these bits.

35.5.2 SETUP Data and EP0 Control Transfer Data

The set address command in control transfers is fully handled by the hardware, and the software can use the last interrupt bit SETADDR if it needs to know that the set address command has occurred.

Each control transfer over USB goes through the SETUP phase, DATA phase (optional), and STATUS phase.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

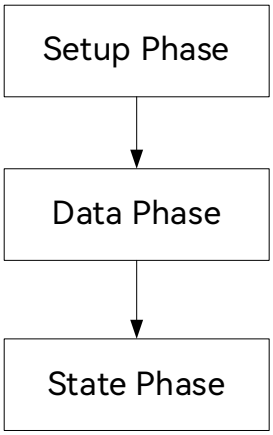

Figure 35-1: USB Transfer Diagram

The STATUS phase is completed with a packet of length 0.

- Setup phase: When the control transfer setup data has been received, a SUDAV_INT interrupt will be generated by the USB, and the received setup data will be stored in the registers USB_SETUP03DATA and USB_SETUP47DATA.
- Data phase: If the direction of the data phase is OUT, then the EP0OUT_INT interrupt will occur when the data transfer is complete, and the results will be stored in the EP0 FIFO, while the data transfer status will be preserved in EP0CSR. If the direction of the data phase is IN, when the host sends an IN token for EP0, the EP0IN_INT interrupt will be triggered, and the data in the EP0 FIFO will be sent back to the host according to the indications in EP0CSR. If the data is not ready, USB will automatically respond with an NAK packet.
- Status phase: The status of the status phase may be sent from the host to the device or from the device back to the host, depending on the direction of the data phase. This requires the software to maintain this status. If it is required to send from the device to the host, the user needs to prepare a 0-length data using the data phase method. USB will send the data back to the host when the EN0IN_INT interrupt occurs, completing the status phase of the entire control transfer.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

35.5.3 Endpoint In Transfer

When the host sends the EPx IN token, the EPxIN_INT interrupt will be triggered, and the data in the EPx FIFO will be sent back to the host according to the indications in EPxCSR, including the length of the data transmitted. The flowchart for the endpoint in transfer is shown in the following figure:

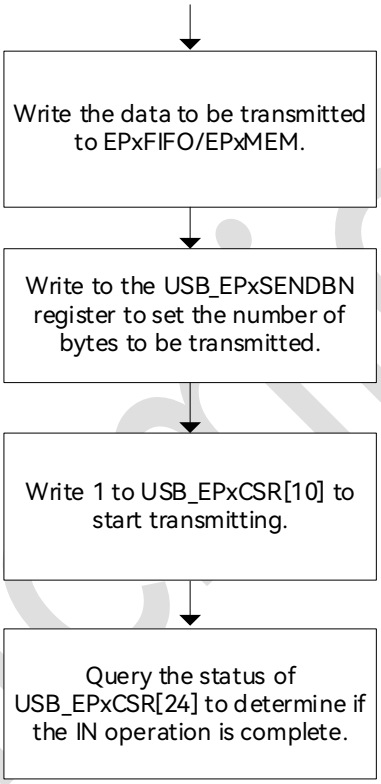

Figure 35-2: USB Endpoint In Transfer Flowchart

Unicmicro

35.5.4 Endpoint Out Transfer

When the EPxOUT_INT occurs, the USB_EPxCSR[7:0] field records the length of the received data, according to which the user can read data from EPxFIFO or EPxMEM. After the data from EPxOUT has been read, the user needs to write USB_EPxCSR[11] to indicate that the next data is allowed to be received. The flowchart for the endpoint out transfer is shown in the following figure:

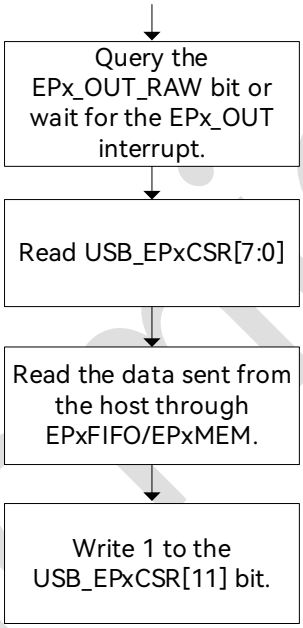

Figure 35-3: USB Endpoint Out Transfer Flowchart

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36 Analog-to-digital Converter (ADC)

36.1 Overview

The analog-to-digital converter (ADC) samples the channel voltage, receives the analog-to-digital conversion results, and transmits the simply processed results to the CPU or other modules in the system.

This ADC controller manages a 12-bit successive approximation ADC with 16 external channels, where two adjacent external channels can form a pair of differential input signals. The ADC controller supports regular sequences of up to 16 positions and injection sequences of up to 4 positions, supports dual-ADC cooperative scanning, and features an analog watchdog function. It has a 32-level deep RX FIFO, and each channel has its own independent data register.

36.2 Main Features

- Input voltage range: $V_{REFN} \leq V_{IN} \leq V_{REFP}$, where V_{REFN} is connected to the ground (V_{SSA}) of the analog circuit, and V_{REFP} can be the power supply (V_{DDA}) of the analog circuit or an externally input reference voltage (V_{REF}).
- The two ADCs can be used with the internal operational amplifier (OPA) to amplify the signal before sampling or as a buffer.
- Configurable ADCCLK clock
- Resolution: 12 bits
- Power-on stabilization time: configurable from 32 to 63 ADCCLK cycles
- Input channels: 16 single-ended external input channels are provided, where adjacent pairs of channels can serve as differential external input channels. The regular sequence provides

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

up to 8 positions for channel conversion, each of which can select any channel.

- Trigger mode: software configuring register, or internal timer events (rising edge, falling edge or both edges), or GPIOA interrupts
- Conversion modes: single-shot mode, continuous mode, discontinuous mode, dual-ADC cooperative mode
- Regular and injection sequences: The regular sequence contains up to 20 positions for channel conversion, while the injection sequence contains up to 4 positions for channel conversion. Both groups consist of a sequence of conversions that can be done on any channel and in any order. The injection sequence has higher priority over the regular sequence.
- It is able to continuously handle multiple times (2, 4, 8, 16, 32, 64 or 128 times) sampling conversions on one or some channels and calculate the average
- It is provided with an analog watchdog for monitoring the conversion result.
- Each channel has an independent data register, which can be set uniformly to clear data automatically after reading.
- Each of the two ADC controllers is provided with an RX FIFO that is 32-level deep and 16-bit wide, for storing the conversion results of the regular and injection sequences.
- DMA mode supported

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.3 System Block Diagram

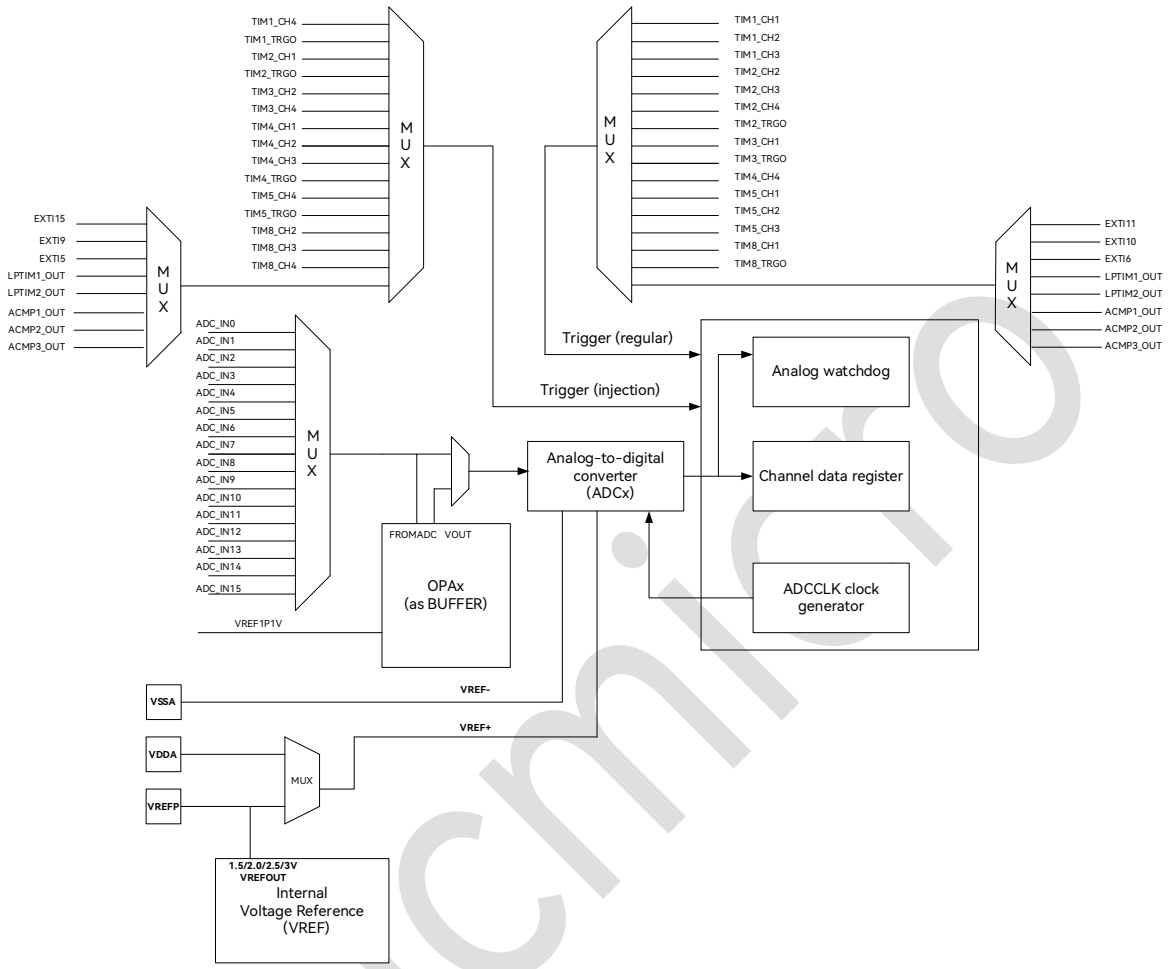

Figure 36-1: ADC System Block Diagram

Notes:

- OPA1 is used as the buffer for ADC1.
- OPA2 is used as the buffer for ADC2.

36.4 Pin Description

Table 36-1: ADC Pin Description

Function Pin	Alternate Function Pin	Direction	Functional Description
ADC_IN0	PA0	AI	ADC analog input channel
ADC_IN1	PA1	AI	ADC analog input channel

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Function Pin	Alternate Function Pin	Direction	Functional Description
ADC_IN2	PA2	AI	ADC analog input channel
ADC_IN3	PA3	AI	ADC analog input channel
ADC_IN6	PA6	AI	ADC analog input channel
ADC_IN7	PA7	AI	ADC analog input channel
ADC_IN8	PB0	AI	ADC analog input channel
ADC_IN9	PB1	AI	ADC analog input channel
ADC_IN10	PC0	AI	ADC analog input channel
ADC_IN11	PC1	AI	ADC analog input channel
ADC_IN12	PC2	AI	ADC analog input channel
ADC_IN13	PC3	AI	ADC analog input channel
ADC_IN14	PC4	AI	ADC analog input channel
ADC_IN15	PC5	AI	ADC analog input channel
VDDA	-	AP	Analog power 3.3 V
VSSA	-	AG	Analog ground
VREFP	-	AP	Analog positive reference voltage

36.5 Functional Description

36.5.1 Analog Watchdog

The analog watchdog can compare the channel data against the upper and lower threshold values. If the channel data exceeds the range defined by these thresholds, the watchdog will generate an interrupt flag. The channels to be compared can be independently enabled, both the upper and lower thresholds can be freely configured, and the comparison conditions can also be selected. There are three interrupt flags: channel data watchdog alarm interrupt flag, regular sequence watchdog alarm interrupt flag, and injection sequence watchdog alarm interrupt flag. The channel data watchdog alarm interrupt flag does not distinguish which sequence the data belongs to. All three interrupt flags shall be cleared by software writing 1, and will not be cleared by hardware automatically.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Note: In addition to setting up the analog watchdog before starting the conversion, the settings of the analog watchdog can also be modified during the conversion. Whenever the analog watchdog finishes comparing the latest channel data, the 27th bit of the current status register (STAT), the watchdog comparison channel data completion flag (Wdg_cmplt), will be set to 1. At this point, the settings of the analog watchdog can be changed, and the changes will take effect immediately. The Wdg_cmplt flag will be cleared by hardware when software writes to the watchdog comparison condition register, or it can also be cleared by software writing 1 directly to this flag. However, due to the high frequency of channel switching and the relatively long time taken for software to read the register, analyze data, and then write back to the register, there is a risk that the comparison conditions may not be updated in a timely manner.

36.5.2 ADC Clock Generator

The ADC clock generator generates a clock (ADCCLK) specifically for the analog ADC by dividing the APB clock (PCLK) according to the division factor PCLK_DIV set in the CLKCTRL register. The clock used by the ADC controller is the system clock (PCLK).

In dual-ADC cooperative operation mode, whether for regular sequence cooperation or injection sequence cooperation, ADC2 always uses the same clock as ADC1. This means that the clock for ADC2 is selected and generated by the CLKCTRL register of ADC controller 1, while the clock selection and clock division settings in the CLKCTRL register of ADC controller 2 have no effect. When ADC2 operates independently, its clock is still generated by ADC controller 2.

ADCCLK comes from two sources: An ADC clock generated by the internal clock generator of the ADC controller, or an external input clock.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.5.2.1 Internal Clock Generator

The formula for the internally generated ADC clock frequency is as follows:

$$f_{ADCCLK} = f_{PCLK}/(Pclk_div + 1)$$

Wherein, f_{ADCCLK} is the internal ADC clock frequency, f_{PCLK} is the APB clock frequency, and Pclk_div is the frequency division factor stored in the CLKCTRL register, with a valid range of 0 ≤ Pclk_div ≤ 65535.

In terms of the timing logic for the ADC controller itself, the fastest division frequency allowed is 1 for the ADC single mode or the regular sequence delayed continuous scanning mode. For the dual-ADC collaborative mode, the fastest division frequency allowed is 2. When ADCC is in single scanning or delayed continuous scanning mode, the ADCCLK is allowed to operate at a maximum of 1 division frequency. In parallel scanning, delayed single scanning, or delayed intermittent scanning modes, the ADCCLK is allowed to operate at a maximum of 2 division frequency.

Note: The analog ADC supports a maximum sampling rate of 5.25 MSPS.

36.5.2.2 External Clock Synchronization Circuit

The external input ADC clock is first synchronized by the PCLK clock domain (sampled by PCLK) before being input to the analog ADC. Therefore, the frequency of the external input ADC clock cannot exceed half of the PCLK frequency, i.e.:

$$f_{ADCCLK} \leq f_{PCLK} / 2$$

36.5.3 Data Receiver

The data receiver is responsible for receiving the conversion results from the ADC, calculating the sum and average, and passing the average value to the register and FIFO.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.5.4 RX FIFO

The RX FIFO has a depth of 32 levels (capable of storing 32 data entries), with each entry consisting of a 4-bit channel number and a 12-bit conversion result, as shown in the table below. The data in the RX FIFO is read from the register logic module and transmitted to the CPU or DMA.

Table 36-2: Data Format in RX FIFO

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Low 4 bits of channel number				Conversion result											

When the RX FIFO has received enough data (as specified by the watermark in the DGCTRL register), the RX FIFO data available interrupt flag (FIFO_AVL) will be set to 1; if DMA transfer is enabled, a DMA request will also be issued.

If the RX FIFO is full, new data cannot be written.

If an overflow occurs in the RX FIFO, the RX FIFO overflow interrupt flag (FIFO_OVF) will be set to 1.

36.6 Conversion Mode Description

36.6.1 Overview of Conversion Sequence

The ADC controller has regular sequence and injection sequence. The regular sequence contains up to 20 positions for channel conversion, while the injection sequence contains up to 4 positions for channel conversion. Both groups consist of a sequence of conversions that can be done on any channel and in any order.

The regular trigger signal can be received when the ADC is in the stabilization state or idle state; the injection trigger signal can be received when the ADC is in the stabilization state, idle state

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

or regular state. If the ADC receives a regular or injection trigger signal while in a stabilization state, it will begin the conversion only after stabilization. If the ADC receives a regular or injection trigger signal while in an idle state, it will start the conversion immediately.

The injection sequence always takes priority over the regular sequence. However, new incoming injection trigger signals cannot interrupt a conversion that is currently ongoing for an injection sequence. If an injection sequence conversion is triggered while a regular sequence is being converted, it must be checked whether the number of ADC_SOC signal occurrences for the interrupted regular sequence channel equals the number of conversions set for that channel (i.e., whether the interrupted channel is on its last conversion). If the set number of conversions has been reached, the interrupted channel can proceed with the conversion and write data to the data register or FIFO. If the set number of conversions has not been reached, the ADC will wait for the most recent conversion to complete, discard the data, and then start the injection sequence conversion. If an injection sequence is currently converting and a regular trigger signal or a new injection trigger signal arrives, the new trigger signal will be ignored, and the ongoing injection sequence conversion will continue. If both a regular sequence conversion and an injection sequence conversion are triggered within the same PCLK clock cycle while in an idle state, the regular trigger signal will be ignored and the injection sequence will start converting. If a regular sequence conversion is interrupted by an injection sequence, after the injection sequence conversion ends, the ADC will require 16 ADCCLK cycles (during which the ADC_SOC signal will not be sent) to resume the regular sequence conversion according to the selection of the RGL_RCV_SEL bit in the DGCTRL register. It will either continue from the interrupted position or restart from the first position.

For independently operating ADC, there are three conversion modes for the regular sequence: single scan mode, continuous scan mode and intermittent scan mode, and only one conversion mode for the injection sequence: single-trigger scan.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

For cooperatively operating ADC, there are three cooperative modes for the regular sequence: Independent scan mode, synchronous scan mode, and delayed scan mode; and three cooperative modes for the injection sequence: Independent scan mode, parallel scan mode, and alternating trigger scan mode. The options for cooperative operation mode shall be used in combination with the options for independent operation mode, for example, adopting the mode of “synchronous continuous scan for regular sequence + alternating trigger scan for injection sequence”.

Specifically, if the cooperative mode of the regular sequence is set to synchronous scan mode, the ADC with “short” sequence will wait for the ADC with “long” sequence to ensure that each round of regular sequence conversions starts synchronously. Since the number of positions in the regular sequences of the two ADCs may differ, and the sampling times of each channel (multiple samples averaged) may also differ, it is possible for the regular sequence conversion times of the two ADCs to vary. Similarly, the conversion times of the injection sequences may also differ.

It is worth noting that the parallel conversion mode of the injection sequence does not guarantee a synchronous start, but it ensures synchronization upon completion through mutual waiting. If the regular sequence operates in independent scan mode, the clock frequency, phase and the time point for starting the regular sequence scanning of the two ADCs can be different. When a start signal of injection sequence conversion arrives, based on the clock generated by the ADC controller 1, the two ADCs will immediately start the injection sequence conversion after completing the current ongoing regular sequence conversion lasting 16 ADCCLK cycles, without waiting. Therefore, the start time points of the injection sequence conversion are likely to be different. However, when both ADCs complete the injection sequence conversion, they need to wait for each other to finish the injection sequence conversion before the state machine

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

can return to the state it was in prior to entering the injection state (which could be the regular state or idle state).

In summary, if the cooperative mode of the regular sequence is set to synchronous scan mode, there are two scenarios of "waiting": one where an ADC that has completed the regular sequence conversion waits for another ADC that is still converting the regular sequence; and another where an ADC that has completed the injection sequence conversion waits for another ADC that is still converting the injection sequence, and then they synchronously resume the regular sequence conversion. Note that the ADC in the waiting state is not idle, i.e., the idle state flag remains 0.

36.6.2 Basic Conversion Mode of Regular Sequence

The basic conversion modes for regular sequences include single scan mode, continuous scan mode and intermittent scan mode, which can be used in combination with the dual-ADC cooperative method.

36.6.2.1 Single Scan Mode for Regular Sequence

In single scan mode, when the regular sequence conversion is triggered, the regular sequence is only converted once and then returns to the idle state.

36.6.2.2 Continuous Scan Mode for Regular Sequence

In continuous scan mode, when the regular sequence conversion is triggered, the regular sequence is converted continuously. If it is necessary to stop the conversion of the regular sequence, then the sampling controller and data receiver of the ADC controller can be briefly reset by software writing a 1 to bit 2 Adc_stop of the analog circuit control register (ANCTRL), allowing the state machine to return to the stabilization state.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.6.2.3 Intermittent Scan Mode for Regular Sequence

In intermittent scan mode, each time the regular sequence conversion is triggered, it will convert the channels corresponding to (Short_leng + 1) positions in the regular sequence, and then return to the idle state. If there are less than (Short_leng + 1) unconverted positions remaining in the regular sequence, all the remaining unconverted positions will be converted before returning to the idle state. For example, if there are 8 positions in the regular sequence and 3 positions in the regular short sequence, then the first trigger will convert the channels corresponding to positions 0, 1 and 2, the second trigger will convert the channels corresponding to positions 3, 4, and 5, and the third trigger will convert the channels corresponding to positions 6 and 7.

36.6.3 Basic Conversion Mode of Injection Sequence

There is only one basic conversion mode for injection sequences, the single scan mode. In single scan mode, when the injection sequence conversion is triggered, the injection sequence is only converted once and then returns to the idle state.

36.6.4 Dual-ADC Cooperative Mode for Regular Sequence

The dual-ADC cooperative mode for regular sequences includes synchronous scan mode (also known as simultaneous mode or parallel mode) and delayed scan mode (also known as alternating mode or following mode). By default, the two ADCs operate independently.

When one of the dual-ADC cooperative modes is selected for the regular sequence, both ADC controllers will use the regular trigger signal from ADC controller 1, and both ADCs will utilize the clock generated by ADC controller 1.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.6.4.1 Synchronous Scan Mode for Dual-ADC Regular Sequence

In synchronous scan mode for regular sequences, if both ADCs are in the idle state, both ADCs will start synchronous conversion when ADC controller 1 receives the regular trigger signal.

If the regular sequences or (in intermittent scan mode) the regular short sequences of the two ADC controllers do not finish at the same time, the two ADC controllers will wait for each other to finish the current sequence to ensure that the next conversion of the regular sequence or regular short sequence still starts synchronously.

Note: If both ADCs are not idle at the arrival of the regular trigger signal from ADC controller 1, the two ADCs will start the first round of conversions sequentially, and the synchronous start of conversion can only be realized after a round of waiting process.

36.6.4.2 Delayed Scan Mode for Dual-ADC Regular Sequence

In delayed scan mode for regular sequences, when ADC controller 1 receives the regular trigger signal, ADC1 will start conversion first. After a delay of 6−21 ADCCLK cycles, ADC2 will then start conversion. If the delay time does not exceed 10 ADCCLK cycles, then the conversion of ADC2 appears to be delayed relative to the conversion of ADC1.

If the delay time exceeds 10 ADCCLK cycles, then, apart from the first conversion of ADC1, subsequent start of conversion for its own ADC will have a delay relative to the start of conversion of the other ADC.

In addition, the delayed continuous scan mode for regular sequences is the only cooperative mode that does not involve a waiting process at the end of the regular sequence conversion, whereas the delayed single or intermittent scan modes for regular sequences still have an end waiting process.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.6.5 Dual-ADC Cooperative Mode for Injection Sequence

The dual-ADC cooperative mode for injection sequences includes parallel scan mode and alternating trigger scan mode. By default, the two ADCs operate independently.

When one of the dual-ADC cooperative modes is selected for the injection sequence, both ADC controllers will use the injection trigger signal from ADC controller 1, and both ADCs will utilize the clock generated by ADC controller 1.

36.6.5.1 Parallel Scan Mode for Dual-ADC Injection Sequence

In the injection sequence parallel scan mode, when ADC controller 1 receives the injection trigger signal, if both ADCs are performing regular conversions, they will each start the injection parallel conversion after completing the most recent regular conversion; if both ADCs are idle, they will directly enter the injection state and start the injection parallel conversion.

Suppose that in the partial process of independent continuous conversion of regular sequence and parallel conversion of injection sequence, because the regular sequence conversions of the two ADCs are independent, their ADCCLK counters may step differently, resulting in different starting times for the parallel conversion of injection sequence. However, they will still have a waiting process at the end of the injection sequence conversion. The situation is similar for delayed continuous conversion of regular sequence and parallel conversion of injection sequence.

However, if the regular sequence is synchronously converted and the injection sequence is converted in parallel, the ADCCLK counters of the two ADCs step in unison, leading to the parallel conversion of injection sequence starting at the same time.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.6.5.2 Alternating Trigger Scan Mode for Dual-ADC Injection Sequence

In the alternating trigger scan mode for injection sequence, when ADC controller 1 receives the injection trigger signal, only one of the two ADCs will start the injection sequence conversion, while the other will enter the injection state and wait, and then both ADCs will simultaneously resume their previous states. The 1st, 3rd, 5th, … (odd-numbered) injection trigger signals will cause ADC1 to convert the injection sequence; the 2nd, 4th, 6th, … (even-numbered) injection trigger signals will cause ADC2 to convert the injection sequence.

Note: In alternating trigger mode, the injection conversion of the next ADC can only be triggered after the previous ADC has completed its injection conversion. If a new trigger signal arrives while the previous ADC injection conversion is not yet completed, the incoming trigger signal will be ignored.

36.7 Register Description

ADCC1 register base address: 0x4700_6000

ADCC2 register base address: 0x4700_7000

The registers are listed below:

Table 36-3: List of ADC Memory-mapped Registers

Offset Address	Name	Reset Value	Description
0x0	ADC_ANCTRL	0x0	Analog circuit control register
0x4	ADC_DGCTRL	0x0	Digital circuit control register
0x8	ADC_CLKCTRL	0xA400003	ADC clock control register
0xC	ADC_CHAVGCFG0	0x0	Multiple average setting register 0
0x10	ADC_CHAVGCFG1	0x0	Multiple average setting register 1
0x14	ADC_RGLCHCFG0	0x0	Regular sequence channel setting register 0

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Reset Value	Description
0x18	ADC_RGLCHCFG1	0x0	Regular sequence channel setting register 1
0x1C	ADC_RGLCHCFG2	0x0	Regular sequence channel setting register 2
0x20	ADC_RGLCHCFG3	0x0	Regular sequence channel setting register 3
0x24	ADC_INJCHCFG	0x0	Injection sequence channel setting register
0x28	ADC_CHDAT0	0x0	Channel data register 0
0x2C	ADC_CHDAT1	0x0	Channel data register 1
0x30	ADC_CHDAT2	0x0	Channel data register 2
0x34	ADC_CHDAT3	0x0	Channel data register 3
0x38	ADC_CHDAT4	0x0	Channel data register 4
0x3C	ADC_CHDAT5	0x0	Channel data register 5
0x40	ADC_CHDAT6	0x0	Channel data register 6
0x44	ADC_CHDAT7	0x0	Channel data register 7
0x48	ADC_CHDAT8	0x0	Channel data register 8
0x4C	ADC_CHDAT9	0x0	Channel data register 9
0x50	ADC_CHDAT10	0x0	Channel data register 10
0x54	ADC_CHDAT11	0x0	Channel data register 11
0x58	ADC_CHDAT12	0x0	Channel data register 12
0x5C	ADC_CHDAT13	0x0	Channel data register 13
0x60	ADC_CHDAT14	0x0	Channel data register 14
0x64	ADC_CHDAT15	0x0	Channel data register 15
0x68	ADC_CHDAT16	0x0	Channel data register 16
0x6C	ADC_DUALDAT	0x0	Dual data shadow register
0x70	ADC_FIFO_OUT	0x0	RX FIFO data register
0x74	ADC_WDGEN	0x0	Watchdog enable register
0x78	ADC_WDGCOND	0xFFF0000	Watchdog comparison condition register
0x7C	ADC_STAT	ADC1: 0x3000000 ADC2: 0x2000000	Current status register
0x80	ADC_INTSTAT	0x0	Interrupt status/clear register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Name	Reset Value	Description
0x84	ADC_INTEN	0x0	Interrupt enable register
0x88	ADC_MINTSTAT	0x0	Enabled interrupt status shadow register

Registers are detailed in the following sections.

36.7.1 Analog Circuit Control Register (ADC_ANCTRL)

Offset address: 0x00

Reset value: 0x000 0000

Bit	Name	Attribute	Reset Value	Description
32:15	RSV	-	-	Reserved
14	OPAMP_EN	R/W	0x0	ADC and OPA connection enable: Note: This register for ADC Controller 1 corresponds to OPA1, while the register for ADC Controller 2 corresponds to OPA2. (The OPA positive channel selection signal (SELP[2:0]) shall select the connection from the ADC MUX output. For detailed usage, please refer to the process "ADC Sampling via OPA Buffering".) 0: Disconnected 1: Connected
13	DIFF_IN14_EN	R/W	0x0	Differential input enable for channels 14 and 15: 0: Single-ended input 1: Differential input
12	DIFF_IN12_EN	R/W	0x0	Differential input enable for channels 12 and 13: 0: Single-ended input 1: Differential input
11	DIFF_IN10_EN	R/W	0x0	Differential input enable for channels 10 and 11:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Single-ended input 1: Differential input
10	DIFF_IN8_EN	R/W	0x0	Differential input enable for channels 8 and 9: 0: Single-ended input 1: Differential input
9	DIFF_IN6_EN	R/W	0x0	Differential input enable for channels 6 and 7: 0: Single-ended input 1: Differential input
8	DIFF_IIN4_EN	R/W	0x0	Differential input enable for channels 4 and 5: 0: Single-ended input 1: Differential input
7	DIFF_IN2_EN	R/W	0x0	Differential input enable for channels 2 and 3: 0: Single-ended input 1: Differential input
6	DIFF_IN0_EN	R/W	0x0	Differential input enable for channels 0 and 1: 0: Single-ended input 1: Differential input
5:4	VREFP_SEL	R/W	0x0	ADC positive reference voltage selection: 0 or 1: Power supply V_{DDA} of analog circuit 2: External input reference voltage V_{REFP} or provided by the internal VREF module 3: Reserved
3	POWER_ON	R/W	0x0	ADC power-up control: The analog ADC requires at least 32 ADCCLK cycles for stabilization after powering up. 0: Power down 1: Power up
2	ADC_STOP	W1C	0x0	Conversion stop control:

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				Writing 1 in any state will reset the sampler and receiver of the ADC controller to their initial values and place them into the idle state. The sampler will stop sending adc_soc signals to the analog ADC, and the receiver will no longer receive new incoming data unless a new sequence conversion is initiated by software or hardware. This bit does not affect the generation of ADCCLK, nor does it affect the register or RX FIFO. 0: Write invalid 1: Stop conversion
1	INJ_START	W1C	0x0	Injection sequence conversion start control: In the stable state, idle state or normal state, writing 1 will start the injection sequence conversion after the current conversion ends; writing 1 during the injection state is invalid. 0: Write invalid 1: Software starts the injection sequence conversion.
0	RGL_START	W1C	0x0	Regular sequence conversion start control: In the stable state or idle state, writing 1 will immediately start the regular sequence conversion; writing 1 during the regular or injection state is invalid. 0: Write invalid 1: Software starts the regular sequence conversion.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.7.2 Digital Circuit Control Register (ADC_DGCTRL)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:29	RSV	-	-	Reserved
28	COMPLEMNT_EN	R/W	0x0	Enable the ADC conversion data to be stored in the register and RX FIFO in two's complement format: The watchdog compares the sign-magnitude representation with the threshold and is not affected by this bit. 0: Save data in sign-magnitude format. 1: Save data in two's complement format.
27:26	DMA_MOD	R/W	0x0	DMA receive mode selection: 0 or 3: DMA receive disabled 1: DMA receive mode 1, where a DMA request is issued when the number of data stored in FIFO (mixed regular sequence and injection sequence) reaches the number set by the watermark. 2: DMA receive mode 2, where a DMA request is issued when the ADC controller itself receives data, allowing reading the data of channel data registers 0–15 (CHDAT0–15) or dual data shadow registers (DUALDAT).
25:24	WATERMARK	R/W	0x0	RX FIFO data available trigger count selection: When a certain amount of data is available in RX FIFO, FIFO_AVL will be set to 1, triggering a DMA transfer. 0: At least 1 data in RX FIFO

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: At least 4 data in RX FIFO 2: At least 8 data in RX FIFO 3: At least 16 data in RX FIFO
23	FIFO_EN	R/W	0x0	RX FIFO enable: 0: RX FIFO disabled, data cleared 1: RX FIFO enabled
22	DATA_R_CLR	R/W	0x0	Enable automatic clearing of channel data after reading: 0: Data will not be cleared after reading the channel data register. 1: Data will be automatically cleared after reading the channel data register.
21:20	INJ_TRIG_EDG	R/W	0x0	Trigger signal edge selection for injection sequence conversion: 0: Signal trigger disabled 1: Rising- edge trigger 2: Falling-edge trigger 3: Double-edge trigger
19:16	INJ_TRIG_SEL	R/W	0x0	Trigger signal source selection for injection sequence conversion: 0x0: CC4 event of timer 1 0x1: TRGO event of timer 1 0x2: CC1 event of timer 2 0x3: TRGO event of timer 2 0x4: CC2 event of timer 3 0x5: CC4 event of timer 3 0x6: CC1 event of timer 4 0x7: CC2 event of timer 4 0x8: CC3 event of timer 4 0x9: TRGO event of timer 4 0xA: CC4 event of timer 5 0xB: TRGO event of timer 5 0xC: CC2 event of timer 8 0xD: CC3 event of timer 8 0xE: CC4 event of timer 8 0xF: External input port 15

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:14	RGL_TRIG_EDG	R/W	0x0	Trigger signal edge selection for regular sequence conversion: 0: Signal trigger disabled 1: Rising- edge trigger 2: Falling-edge trigger 3: Double-edge trigger
13:10	RGL_TRIG_SEL	R/W	0x0	Trigger signal source selection for regular sequence conversion: 0x0: CC1 event of timer 1 0x1: CC2 event of timer 1 0x2: CC3 event of timer 1 0x3: CC2 event of timer 2 0x4: CC3 event of timer 2 0x5: CC4 event of timer 2 0x6: TRGO event of timer 2 0x7: CC1 event of timer 3 0x8: TRGO event of timer 3 0x9: CC4 event of timer 4 0xA: CC1 event of timer 5 0xB: CC2 event of timer 5 0xC: CC3 event of timer 5 0xD: CC1 event of timer 8 0xE: TRGO event of timer 8 0xF: External input port 11
9	RGL_RCV_SEL	R/W	0x0	Position selection for resuming scanning after a regular sequence is interrupted by an injection sequence: 0: Resume scanning from the interrupted position. 1: Restart scanning from the beginning.
8:7	DUAL_INJ_MOD	R/W	0x0	Cooperation mode selection for injection sequence conversion in dual-ADC cooperative mode: 0 or 3: Independent scan mode for injection sequence

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Parallel scan mode for injection sequence 2: Alternating trigger scan mode for injection sequence Note: If it is configured to the parallel scan mode or alternating trigger scan mode for the injection sequence, ADC2 will always use the same clock as ADC1. The clock for ADC2 is selected and generated by the CLKCTRL register of ADC controller 1, while the clock selection and clock division settings in the CLKCTRL register of ADC controller 2 will not take effect.
6:5	DUAL_RGL_MOD	R/W	0x0	Cooperation mode selection for regular sequence conversion in dual-ADC cooperative mode: 0 or 3: Independent scan mode for regular sequence 1: Synchronous scan mode for regular sequence 2: Delayed scan mode for regular sequence Note: If it is configured to the synchronous scan mode or delayed scan mode for the regular sequence, ADC2 will always use the same clock as ADC1. The clock for ADC2 is selected and generated by the CLKCTRL register of ADC controller 1, while the clock selection and clock division settings in the CLKCTRL register of ADC controller 2 will not take effect.
4	INJ_MOD	R/W	0x0	Injection sequence conversion mode selection: 0: No conversion 1: Single trigger scan mode

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3:2	RGL_MOD	R/W	0x0	Regular sequence conversion mode selection: 0: No conversion 1: Single scan mode 2: Continuous scan mode 3: Intermittent scan mode
1	RD_EDGE	R/W	0x0	Trigger edge selection for ADC controller to read conversion results: 0: ADC_EOC rising edge 1: ADC_EOC falling edge
0	ADCC_EN	R/W	0x0	ADC controller enable: This bit controls the clock generator, sampling controller and data receiver. Note: The RX FIFO data is cleared through the FIFO_EN bit (bit 23) of this register. 0: Disable ADC controller. 1: Enable ADC controller.

36.7.3 ADCx Clock Control Register (ADC_CLKCTRL)

Offset address: 0x08

Reset value: 0x0A40 0003

Bit	Name	Attribute	Reset Value	Description
31:28	SYNC_DELAY	R/W	0x0	In the regular sequence delay mode of the dual-ADC cooperative mode, the delay time setting for the current ADC to start conversion relative to the previous ADC is applicable only in this mode and does not take effect in other operation modes of the regular sequence. For example, this bit in ADC controller 1 indicates the delay time of the ADC_SOC signal set of ADC1 relative to ADC2; while this bit in ADC controller 2

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				indicates the delay time of ADC_SOC signal set of ADC2 relative to ADC1. To stagger the sampling time by 4 ADCCLK cycles, the delay time is set to (Sync_delay + 6) ADCCLK cycles, with a range of 6 to 21 ADCCLK cycles.
27:24	CH_SWITCH	R/W	0xA	Channel switching timing selection: These bits decide when to switch channels based on ADCCLK counter value. Note: 1) Do not set it to 0–3, as the ADC is sampling when the ADCCLK counter reaches 0–3, and channel switching is prohibited at this time. 2) For differential inputs, it is recommended to set this register to 0xE or 0xF.
23:22	SOC_WIDTH	R/W	0x1	ADC_SOC signal width: The duration of the ADC_SOC signal being high is from the falling edge of the ADCCLK to the rising edge of the (Soc_width + 2)th ADCCLK. 0: 1.5 ADCCLK clock cycles 1: 2.5 ADCCLK clock cycles 2: 3.5 ADCCLK clock cycles 3: 4.5 ADCCLK clock cycles
21:17	STABL_TIME	R/W	0x0	ADC stabilization time configuration: ADC stabilization time = (Stabl_time + 32) ADCCLK cycles, with a range of 32 to 63.
16	ADCCLK_SEL	R/W	0x0	ADC clock source selection: 0: PCLK divided clock 1: Synchronized external clock (XTH clock only)

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:0	PCLK_DIV	R/W	0x3	PCLK division factor setting: The effective division factor is (PCLK_DIV + 1), with a range of 1 to 65536. Note: 1. If f_{PCLK} = 168 MHz, the actual division factor shall be greater than or equal to 2, that is, PCLK_DIV ≥ 1 shall be set. 2. For different PCLK frequencies, the ADC independent scan mode and the delayed continuous scan mode for the regular sequence of dual ADCs without injection sequence support a minimum division of 1; other dual-ADC cooperative modes support a minimum division of 2, that is, PCLK_DIV ≥ 1 shall be set.

Notes: ADC1 always uses the ADCCLK clock generated by ADC controller 1. ADC2 uses the ADCCLK clock generated by ADC controller 2 in independent operation mode, but uses the ADCCLK clock generated by ADC controller 1 in dual-ADC cooperative mode.

Furthermore, in dual-ADC cooperative mode, ADC controller 2 still needs to properly configure the bits [31:17] of this register.

36.7.4 Multiple Average Setting Register 0 (ADC_CHAVGCFG0)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29:27	CH9_AVG_SEL	R/W	0x0	Channel 9 conversion number setting (Refer to channel 0 conversion number setting)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
26:24	CH8_AVG_SEL	R/W	0x0	Channel 8 conversion number setting (Refer to channel 0 conversion number setting)
23:21	CH7_AVG_SEL	R/W	0x0	Channel 7 conversion number setting (Refer to channel 0 conversion number setting)
20:18	CH6_AVG_SEL	R/W	0x0	Channel 6 conversion number setting (Refer to channel 0 conversion number setting)
17:15	CH5_AVG_SEL	R/W	0x0	Channel 5 conversion number setting (Refer to channel 0 conversion number setting)
14:12	CH4_AVG_SEL	R/W	0x0	Channel 4 conversion number setting (Refer to channel 0 conversion number setting)
11:9	CH3_AVG_SEL	R/W	0x0	Channel 3 conversion number setting (Refer to channel 0 conversion number setting)
8:6	CH2_AVG_SEL	R/W	0x0	Channel 2 conversion number setting (Refer to channel 0 conversion number setting)
5:3	CH1_AVG_SEL	R/W	0x0	Channel 1 conversion number setting (Refer to channel 0 conversion number setting)
2:0	CH0_AVG_SEL	R/W	0x0	Channel 0 conversion number setting: 0: 1 conversion 1: 2 conversions with results averaged 2: 4 conversions with results averaged 3: 8 conversions with results averaged 4: 16 conversions with results averaged 5: 32 conversions with results averaged 6: 64 conversions with results averaged 7: 128 conversions with results averaged

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.7.5 Multiple Average Setting Register 1 (ADC_CHAVGCFG1)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:21	RSV	-	-	Reserved
20:18	CH16_AVG_SEL	R/W	0x0	Channel 16 conversion number setting (Refer to channel 0 conversion number setting)
17:15	CH15_AVG_SEL	R/W	0x0	Channel 15 conversion number setting (Refer to channel 0 conversion number setting)
14:12	CH14_AVG_SEL	R/W	0x0	Channel 14 conversion number setting (Refer to channel 0 conversion number setting)
11:9	CH13_AVG_SEL	R/W	0x0	Channel 13 conversion number setting (Refer to channel 0 conversion number setting)
8:6	CH12_AVG_SEL	R/W	0x0	Channel 12 conversion number setting (Refer to channel 0 conversion number setting)
5:3	CH11_AVG_SEL	R/W	0x0	Channel 11 conversion number setting (Refer to channel 0 conversion number setting)
2:0	CH10_AVG_SEL	R/W	0x0	Channel 10 conversion number setting (Refer to channel 0 conversion number setting)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.7.6 Regular Sequence Channel Setting Register 0 (ADC_RGLCHCFG0)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29:25	RGL_CH6_SEL	R/W	0x0	Selection of the 6th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
24:20	RGL_CH5_SEL	R/W	0x0	Selection of the 5th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
19:15	RGL_CH4_SEL	R/W	0x0	Selection of the 4th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
14:10	RGL_CH3_SEL	R/W	0x0	Selection of the 3rd conversion channel for regular sequence (Refer to the 1st conversion channel selection)
9:5	RGL_CH2_SEL	R/W	0x0	Selection of the 2nd conversion channel for regular sequence (Refer to the 1st conversion channel selection)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
4:0	RGL_CH1_SEL	R/W	0x0	Selection of the 1st conversion channel for regular sequence: 0x0: External channel 0 0x1: External channel 1 0x2: External channel 2 0x3: External channel 3 0x4: External channel 4 0x5: External channel 5 0x6: External channel 6 0x7: External channel 7 0x8: External channel 8 0x9: External channel 9 0xA: External channel 10 0xB: External channel 11 0xC: External channel 12 0xD: External channel 13 0xE: External channel 14 0xF: External channel 15 0x10−0x1F: Reserved

36.7.7 Regular Sequence Channel Setting Register 1 (ADC_RGLCHCFG1)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29:25	RGL_CH12_SEL	R/W	0x0	Selection of the 12th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
24:20	RGL_CH11_SEL	R/W	0x0	Selection of the 11th conversion channel for regular sequence (Refer to the 1st conversion channel selection)

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19:15	RGL_CH10_SEL	R/W	0x0	Selection of the 10th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
14:10	RGL_CH9_SEL	R/W	0x0	Selection of the 9th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
9:5	RGL_CH8_SEL	R/W	0x0	Selection of the 8th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
4:0	RGL_CH7_SEL	R/W	0x0	Selection of the 7th conversion channel for regular sequence (Refer to the 1st conversion channel selection)

36.7.8 Regular Sequence Channel Setting Register 2 (ADC_RGLCHCFG2)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29:25	RGL_CH18_SEL	R/W	0x0	Selection of the 18th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
24:20	RGL_CH17_SEL	R/W	0x0	Selection of the 17th conversion channel for regular sequence (Refer to the 1st conversion channel selection)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
19:15	RGL_CH16_SEL	R/W	0x0	Selection of the 16th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
14:10	RGL_CH15_SEL	R/W	0x0	Selection of the 15th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
9:5	RGL_CH14_SEL	R/W	0x0	Selection of the 14th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
4:0	RGL_CH13_SEL	R/W	0x0	Selection of the 13th conversion channel for regular sequence (Refer to the 1st conversion channel selection)

36.7.9 Regular Sequence Channel Setting Register 3 (ADC_RGLCHCFG3)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:18	RSV	-	-	Reserved
17:15	SHORT_LENG	R/W	0x0	Channel conversion position number setting in regular short sequence in intermittent scan mode: The number of channel positions in a regular short sequence is set to (Short_leng + 1). The regular short sequence can contain a maximum of 8 channel conversion positions, i.e., the range of Short_leng is 0–7.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
14:10	RGL_LENG	R/W	0x0	Channel conversion position number setting in regular sequence: The number of channel positions in a regular sequence is set to (Rgl_leng + 1). The regular sequence can contain a maximum of 20 channel conversion positions, i.e., the valid range of Rgl_leng is 0–19. Note: The software shall avoid configuring Rgl_leng > 19, as any positions beyond this will be assigned to channel 0 and will only be converted once.
9:5	RGL_CH20_SEL	R/W	0x0	Selection of the 20th conversion channel for regular sequence (Refer to the 1st conversion channel selection)
4:0	RGL_CH19_SEL	R/W	0x0	Selection of the 19th conversion channel for regular sequence (Refer to the 1st conversion channel selection)

36.7.10 Injection Sequence Channel Setting Register (ADC_INJCHCFG)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:22	RSV	-	-	Reserved
21:20	INJ_LENG	R/W	0x0	Channel conversion position number setting in injection sequence: The number of channel positions in an injection sequence is set to (Inj_leng + 1).

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				The injection sequence can contain a maximum of 4 channel conversion positions.
19:15	INJ_CH4_SEL	R/W	0x0	Selection of the 4th conversion channel for injection sequence (Refer to the 1st conversion channel selection)
14:10	INJ_CH3_SEL	R/W	0x0	Selection of the 3rd conversion channel for injection sequence (Refer to the 1st conversion channel selection)
9:5	INJ_CH2_SEL	R/W	0x0	Selection of the 2nd conversion channel for injection sequence (Refer to the 1st conversion channel selection)
4:0	INJ_CH1_SEL	R/W	0x0	Selection of the 1st conversion channel for injection sequence 0x0: External channel 0 0x1: External channel 1 0x2: External channel 2 0x3: External channel 3 0x4: External channel 4 0x5: External channel 5 0x6: External channel 6 0x7: External channel 7 0x8: External channel 8 0x9: External channel 9 0xA: External channel 10 0xB: External channel 11 0xC: External channel 12 0xD: External channel 13 0xE: External channel 14 0xF: External channel 15 0x10–0x1F: Reserved

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.7.11 Channel Data Registers 0–16 (ADC_CHDAT0–19)

Offset address: 0x28–0x68

Reset value: 0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	-	-	Reserved
16	DATA_VALID	R	0x0	Data valid signal: This signal is activated upon obtaining valid data. It is cleared by hardware when Adc_en = 0 or after the software reads this register. 0: Data is invalid. 1: Data is valid.
15:12	RSV	-	-	Reserved
11:0	CH_DATA	R	0x0	Channel data: If Dat_r_clr = 1, it will be cleared by hardware after the software reads this register.

Note: Registers 0 to 15 correspond to the data from 16 external input channels. For dual-port differential inputs, the differential data from ports 0 and 1 will be stored in register 0, the differential data from ports 2 and 3 will be stored in register 2, and so on.

36.7.12 Dual-data Shadow Register (ADC_DUALDAT)

Offset address: 0x6C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	ADC2_LAST_CH_NUM	R	0x0	ADC2 last converted channel number
27:16	ADC2_DATA	R	0x0	ADC2 last conversion result
15:12	ADC1_LAST_CH_NUM	R	0x0	ADC1 last converted channel number
11:0	ADC1_DATA	R	0x0	ADC1 last conversion result

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Notes:

- The channel numbers and results of the most recent conversions from both ADCs are mapped to this register address simultaneously, which facilitates data reading in dual-ADC cooperative mode.
- In DMA mode 2, a DMA request is issued as soon as the ADC itself has valid data ready. However, if the other ADC does not have data ready at that moment, the channel number of the other ADC indicates the channel currently being prepared, and the data is the previous data from that channel (invalid). Therefore, to ensure that both ADCs are ready for data simultaneously, the conversion counts for each channel of both ADCs shall be set to the same value, and conversions shall be triggered only after the IDLE_STATE status flag is set, allowing both ADCs to start conversions at the same time. Of course, since each ADC controller issues its own DMA request, it is also feasible to read only the valid data from each ADC during each request.

36.7.13 RX FIFO Data Register (ADC_FIFO_OUT)

Offset address: 0x70

Reset value: 0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:12	LAST_CH_NUM	R	0x0	Channel number
11:0	FIFO_OUT	R	0x0	RX FIFO data

36.7.14 Watchdog Enable Register (ADC_WDGEN)

Offset address: 0x74

Reset value: 0x0000 0000

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15	WDG_EN_15	R/W	0x0	Channel 15 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
14	WDG_EN_14	R/W	0x0	Channel 14 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
13	WDG_EN_13	R/W	0x0	Channel 13 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
12	WDG_EN_12	R/W	0x0	Channel 12 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
11	WDG_EN_11	R/W	0x0	Channel 11 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
10	WDG_EN_10	R/W	0x0	Channel 10 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
9	WDG_EN_9	R/W	0x0	Channel 9 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
8	WDG_EN_8	R/W	0x0	Channel 8 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
7	WDG_EN_7	R/W	0x0	Channel 7 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
6	WDG_EN_6	R/W	0x0	Channel 6 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
5	WDG_EN_5	R/W	0x0	Channel 5 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
4	WDG_EN_4	R/W	0x0	Channel 4 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	WDG_EN_3	R/W	0x0	Channel 3 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
2	WDG_EN_2	R/W	0x0	Channel 2 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
1	WDG_EN_1	R/W	0x0	Channel 1 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.
0	WDG_EN_0	R/W	0x0	Channel 0 watchdog enable: 0: Disable watchdog function. 1: Enable watchdog function.

36.7.15 Watchdog Comparison Condition Register (ADC_WDGCOND)

Offset address: 0x78

Reset value: 0x0FFF 0000

Bit	Name	Attribute	Reset Value	Description
31	CONDITION	R/W	0x0	Selection of trigger condition for watchdog event: 0: The watchdog will trigger an alarm when the sign-magnitude of converted data exceeds the threshold range, i.e., the sign-magnitude of converted data > H_threshold, or the sign-magnitude of converted data < L_threshold. 1: The watchdog will trigger an alarm when the sign-magnitude of converted data is within the threshold range, i.e., L_threshold ≤ the sign-magnitude of converted data ≤ H_threshold.
30:28	RSV	R	0x0	Reserved
27:16	H_THRESHOLD	R/W	0xFFF	Watchdog threshold upper limit

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
15:12	RSV	R	0x0	Reserved
11:0	L_THRESHOLD	R/W	0x0	Watchdog threshold lower limit

Notes:

1. If multiple-sampling mode is used, the watchdog only compares the average value of the converted data with the thresholds.
2. In addition to setting the watchdog comparison conditions before ADC startup, the comparison conditions can also be changed during ADC conversion whenever the WDG_CMPLT flag is set to 1, so that the comparison conditions for each channel data can vary. However, this approach carries the risk of delayed updates to the comparison conditions.
3. If it is set to the watchdog reset value, the watchdog will not trigger an alarm.

36.7.16 Current Status Register (ADC_STAT)

Offset address: 0x7C

Reset value: ADC1: 0x3000000

ADC2: 0x2000000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	Reserved
27	WDG_CMPLT	R/W1C	0x0	Watchdog channel data comparison completion flag, can be cleared by writing 1. When the watchdog finishes comparing the data of a channel, this flag will be set to 1. When the flag is 1, the watchdog comparison conditions can be changed, and the new comparison conditions is used for the comparison of the next channel data (regardless of regular sequence or injection sequence). After writing to the watchdog comparison

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				condition register (WDGCOND), this flag will also be cleared. During ADC operation, when this flag is 0, do not change the watchdog comparison conditions to avoid erroneous comparison results. 0: Watchdog channel data comparison is completed. 1: Watchdog channel data comparison is not completed.
26	FIFO_FULL	R	0x0	RX FIFO full flag: 0: Less than 32 data entries in RX FIFO 1: 32 data entries in RX FIFO
25	FIFO_EMPTY	R	0x1	RX FIFO empty flag: 0: There is data in RX FIFO. 1: There is no data in RX FIFO.
24	INJ_IN_TURN	R	ADC1: 0x1 ADC2: 0x0	Alternating trigger injection cycle: This register indicates which ADC is currently executing the injection conversion in the alternating trigger scan mode, or which ADC will perform the conversion the next time the injection conversion is triggered. 0: Current ADC is not in turn. 1: Current ADC is in turn.
23:21	INJ_NUM	R	0x0	Position number of the channel currently being converted in the injection sequence: 1–4 indicates the position number of the channel being converted in the injection sequence, while 0 indicates that no channel in the injection sequence is currently being converted.
20:16	RGL_NUM	R	0x0	Position number of the channel currently being converted in the regular sequence: 1–20 indicates the position number of the channel being converted in the regular sequence, while 0 indicates that no

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				channel in the regular sequence is currently being converted (including when the injection sequence is being converted).
15:11	CH_NUM	R	0x0	Number of channels being converted: Valid range: 0–15; when idle, it displays the channel number of the first position in the regular sequence (set by Rgl_ch1_sel).
10:4	CONV_CNT	R	0x0	Number of completed conversions at the current conversion position: Since each channel can independently set the number of conversions used to calculate the average, the valid range of this register value is no greater than (the selected value of Chn_avg_sel - 1).
3:1	RSV	-	-	Reserved
0	IDLE_STATE	R	0x0	Idle status flag: When this flag is 1, the ADC can respond to the trigger signals for the regular sequence. After ADC reset, it first enters the stabilization state, and then enters the idle state after 32 ADC clock cycles. In dual-ADC cooperative mode, if the conversion time length of the two ADCs is different, this flag will remain 0 when the ADC that has completed the conversion is waiting for the other ADC that is still converting.

36.7.17 Interrupt Status / Clear Register (ADC_INTSTAT)

Offset address: 0x80

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
6	WDG_INJ	R/W1C	0x0	Injection sequence watchdog alarm interrupt flag, can be cleared by writing 1 During the conversion of the injection sequence, if the watchdog function is enabled for the most recently converted channel and its conversion result meets the watchdog alarm condition, an interrupt will be triggered.
5	WDG_RGL	R/W1C	0x0	Regular sequence watchdog alarm interrupt flag, can be cleared by writing 1 During the conversion of the regular sequence, if the watchdog function is enabled for the most recently converted channel and its conversion result meets the watchdog alarm condition, an interrupt will be triggered.
4	WDG_CH	R/W1C	0x0	Channel data watchdog alarm interrupt flag, can be cleared by writing 1 If the watchdog function is enabled for the most recently converted channel and its conversion result meets the watchdog alarm condition, an interrupt will be triggered.
3	FIFO_OVF	R/W1C	0x0	RX FIFO overflow interrupt flag, can be cleared by writing 1
2	FIFO_AVL	R/W1C	0x0	RX FIFO data available interrupt flag, can be cleared by writing 1 When the amount of data in the RX FIFO reaches the value set by WATERMARK in the DGCTRL register, this flag will be set to 1.
1	EOIC	R/W1C	0x0	Injection sequence conversion complete interrupt flag, can be cleared by writing 1
0	EORC	R/W1C	0x0	Regular sequence conversion complete interrupt flag, can be cleared by writing 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

36.7.18 Interrupt Enable Register (ADC_INTEN)

Offset address: 0x84

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	R	0x0	Reserved
6	WDG_INJ_EN	R/W	0x0	Injection sequence watchdog alarm interrupt enable
5	WDG_RGL_EN	R/W	0x0	Regular sequence watchdog alarm interrupt enable
4	WDG_CH_EN	R/W	0x0	Channel data watchdog alarm interrupt enable
3	FIFO_OVF_EN	R/W	0x0	RX FIFO overflow interrupt enable
2	FIFO_AVL_EN	R/W	0x0	RX FIFO data available interrupt enable
1	EOIC_EN	R/W	0x0	Injection sequence conversion complete interrupt enable
0	EORC_EN	R/W	0x0	Regular sequence conversion complete interrupt enable

36.7.19 Enabled Interrupt Status Shadow Register (ADC_MINTSTAT)

Offset address: 0x88

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:7	RSV	-	-	Reserved
6	WDG_INJ_M	R	0x0	Enabled injection sequence watchdog alarm interrupt flag
5	WDG_RGL_M	R	0x0	Enabled regular sequence watchdog alarm interrupt flag
4	WDG_CH_M	R	0x0	Enabled channel data watchdog alarm interrupt flag
3	FIFO_OVF_M	R	0x0	Enabled RX FIFO overflow interrupt flag
2	FIFO_AVL_M	R	0x0	Enabled RX FIFO data available interrupt flag

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
1	EOIC_M	R	0x0	Enabled injection sequence conversion complete interrupt flag
0	EORC_M	R	0x0	Enabled regular sequence conversion complete interrupt flag

Note: This register maps the result of the logical "AND" operation between the interrupt status/clear register (INTSTAT) and the interrupt enable register (INTEN). To clear an interrupt flag, write "1" to the (original) interrupt flag in the interrupt status/clear register (INTSTAT).

36.8 Operation Procedure

36.8.1 Multi-channel A/D Conversion in Single Scan Mode of Regular Sequence

In single scan mode, ADC performs only one conversion after starting the conversion.

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[3:2] to 1 to set the regular sequence conversion mode to single scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
11. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
12. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software.
13. Wait for the ADC regular sequence conversion to complete and read the data from the data register of the corresponding channel.
14. If multiple conversions are required, repeat steps 12 and 13.

36.8.2 Multi-channel A/D Conversion in Continuous Scan Mode of Regular Sequence

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[3:2] to 2 to set the regular sequence conversion mode to continuous scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
11. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
12. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software.
13. Wait for the ADC regular sequence conversion to complete and read the data from the data register of the corresponding channel.

36.8.3 Multi-channel A/D Conversion in Intermittent Scan Mode of Regular Short Sequence

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[3:2] to 3 to set the regular sequence conversion mode to intermittent scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.
9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

10. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
11. Configure ADC_RGLCHCFG3[17:15] to set the number of positions for channel conversions in the regular short sequence.
12. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
13. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software for one regular short sequence at a time.
14. Wait for the ADC regular sequence conversion to complete and read the data from the data register of the corresponding channel.
15. If multiple conversions are required, repeat steps 13 and 14. Once all channel positions have been converted, start conversion again from channel position 0.

36.8.4 Multi-channel A/D Conversion in Single Scan Mode of Injection Sequence

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[4] to 1 to set the injection sequence conversion mode to single trigger scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.
9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the injection sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the injection sequence.
11. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
12. Set ADC_ANCTRL[1] to 1 to start the injection sequence conversion via software.
13. Wait for the ADC injection sequence conversion to complete and read the data from the data register of the corresponding channel.
14. If multiple conversions are required, repeat steps 12 and 13.

36.8.5 Analog Watchdog

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[3:2] to 1 to set the regular sequence conversion mode to single scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
11. Configure the watchdog comparison condition register to set the upper and lower limits for the watchdog threshold.
12. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
13. Set ADC_INTEN[5] to 1 to enable the regular sequence watchdog alarm interrupt.
14. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software.
15. Wait for the ADC regular sequence conversion to complete; a watchdog interrupt will be generated if the conversion result exceeds the watchdog limits.
16. Write 1 to clear the ADC_INTSTAT[5] regular sequence watchdog alarm interrupt flag and continue the next conversion.

36.8.6 Differential Channel Input

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC.
4. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[3:2] to 1 to set the regular sequence conversion mode to single scan mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.
9. Configure the ADC_ANCTRL register to enable channel differential input.
10. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
11. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
12. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
13. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software.
14. Wait for the ADC regular sequence conversion to complete and read the differential data of the corresponding channels; for example, the differential data of channels 0 and 1 will be stored in the channel data register 0, and so on.
15. If multiple conversions are required, repeat steps 12 and 13.

36.8.7 Dual-ADC Cooperative Mode for Regular Sequence

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pins corresponding to the ADC1/2 input channels.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC1/2 clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC1/2.
4. Configure ADC_ANCTRL[5:4] to set the ADC1/2 positive reference voltage as the analog circuit power supply V_{DDA}.
5. Configure ADC_DGCTRL[3:2] to set the regular sequence conversion mode.
6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC1 sampling rate.
9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
11. Set ADC_DGCTRL[0] to 1 to enable the ADC1/2 controller.
12. Configure ADC_DGCTRL[6:5] to set the cooperative mode for regular sequence conversion when ADC1/2 work cooperatively.
13. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion for ADC1 via software.
14. Wait for the ADC1/2 regular sequence conversion to complete and read the data from the data register of the corresponding ADC1/2 channels.
15. If multiple conversions are required, repeat steps 13 and 14.

36.8.8 Dual-ADC Cooperative Mode for Injection Sequence

1. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pins corresponding to the ADC1/2 input channels.
2. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC1/2 clock.
3. Set ADC_ANCTRL[3] to 1 to power on the ADC1/2.
4. Configure ADC_ANCTRL[5:4] to set the ADC1/2 positive reference voltage as the analog circuit power supply V_{DDA}.
5. Set ADC_DGCTRL[4] to 1 to set the injection sequence conversion mode to single trigger scan mode.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
7. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
8. Configure ADC_CLKCTRL[15:0] to set the ADC1 sampling rate.
9. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
10. Configure the injection sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the injection sequence.
11. Set ADC_DGCTRL[0] to 1 to enable the ADC1/2 controller.
12. Configure ADC_DGCTRL[8:7] to set the cooperative mode for regular sequence conversion when ADC1/2 work cooperatively.
13. Set ADC_ANCTRL[1] to 1 to start the injection sequence conversion for ADC1 via software.
14. Wait for the ADC injection sequence conversion to complete and read the data from the data register of the corresponding channel.
15. If multiple conversions are required, repeat steps 13 and 14.

36.8.9 ADC Sampling via OPA Buffering

1. Set the OPA to unit buffer mode, and select the signal for the OPA positive channel (SELP) from the output of the ADC MUX.
2. Configure the channel to be converted as an analog interface (GPIOx_MODE) based on the GPIO pin corresponding to the ADC input channel.
3. Configure the APB2 peripheral clock enable register (APB2CKENR) to enable the ADC clock.
4. Set ADC_ANCTRL[3] to 1 to power on the ADC.
5. Configure ADC_ANCTRL[5:4] to set the ADC positive reference voltage as the analog circuit power supply V_{DDA}.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

6. Set ADC_DGCTRL[3:2] to 1 to set the regular sequence conversion mode to single scan mode.
7. Set ADC_DGCTRL[22] to 1 to automatically clear the channel data after reading.
8. Configure ADC_CLKCTRL[27:24] to set the timing at which the channel selects to switch.
9. Configure ADC_CLKCTRL[15:0] to set the ADC sampling rate.
10. Configure the multiple average setting register to set the number of conversions for each channel and average the results.
11. Configure the regular sequence channel setting register to set the channels for conversion at each position and the number of positions for channel conversions in the regular sequence.
12. Set ADC_ANCTRL[14] to 1 to enable the connection between the ADC and OPA.
13. Set ADC_DGCTRL[0] to 1 to enable the ADC controller.
14. Set ADC_ANCTRL[0] to 1 to start the regular sequence conversion via software.
15. Wait for the ADC regular sequence conversion to complete and read the data from the data register of the corresponding channel.
16. If multiple conversions are required, repeat steps 14 and 15.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37 Digital-to-analog Converter (DAC)

37.1 Overview

The digital-to-analog converter (DAC) can convert 12-bit digital signals into analog voltage output.

37.2 Main Features

- 8-bit or 12-bit digital input
- Simultaneous update of conversion data for two converters
- Generation of triangle waves and noise waves
- DMA operation
- External trigger for updating conversion data

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.3 System Block Diagram

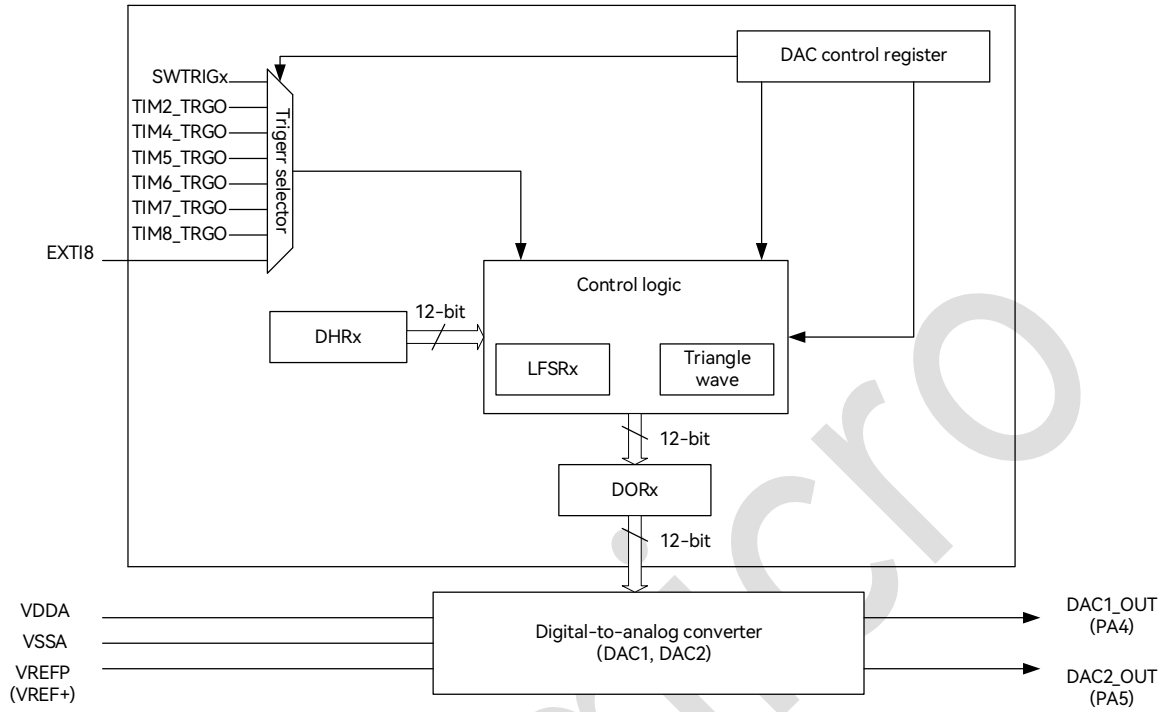

Figure 37-1: DAC System Block Diagram

37.4 Functional Description

37.4.1 DAC Conversion Control

Two DAC channels can be independently enabled or disabled using the DACENx bit in the DAC_CTRL register. Once enabled, the DAC converts the data in DORx to an analog voltage output. The output voltage is given by V_{REF} * (DOR / 4095).

The DAC output buffer can be enabled or disabled using the BUFx bit in the DAC_CTRL register, which can reduce the output impedance and directly drive certain external loads.

The DAC supports 8-bit or 12-bit data; when using 8-bit data, the lower 4 bits of DORx will automatically be loaded with 0.

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.4.2 Trigger Mode

The DAC supports 8 different trigger sources. When the trigger mode is enabled, the data in DHR will be loaded into DORx upon receiving a trigger event.

37.4.3 Noise Wave

The DAC can generate noise waves. When the noise wave mode is enabled, the LFSR will generate noise wave amplitudes in a specific manner. Each time a trigger event is detected, the LFSR value will be added to the value in DHR and then loaded into DORx. The high bits in the LFSR can be masked to control the noise amplitude.

37.4.4 Triangle Wave

The DAC can generate triangle waves. When the triangle wave mode is enabled, the triangle wave counter will generate triangle wave amplitudes within the range specified by MAMPx in the DAC_CTRL register. Each time a trigger event is detected, the triangle wave counter value will be added to the value in DHR and then loaded into DORx.

37.4.5 DMA

When the DMA mode is enabled, the DAC will generate a DMA request upon receiving a hardware trigger. The value written to the DHR register will be loaded into DORx after one cycle. If a hardware trigger is received while the previous DMA request has not been acknowledged, an underflow interrupt will be generated, and the DMA mode will be disabled.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.4.6 Dual DAC Coordination

By using the DAC_DHR12RD, DAC_DHR12LD, or DAC_DHR8RD registers, two DACs can load data simultaneously. When using trigger mode, TENx and TSELx are only effective independently for the corresponding DAC channel, but the same TSEL can be used to synchronize their updates. When using DMA mode, the DHR values of the corresponding channels will be loaded immediately. To synchronize updates via DMA, the trigger mode of the other channel must be disabled.

37.5 Register Description

DAC register base address: 0x4600_C000

The registers are listed below:

Table 37-1: List of DAC Registers

Offset Address	Name	Description
0x00	DAC_CTRL	Control register
0x04	DAC_SWTRG	Software trigger register
0x08	DAC_DHR12R0	DAC1 12-bit right-aligned data register
0x0C	DAC_DHR12L0	DAC1 12-bit left-aligned data register
0x10	DAC_DHR8R0	DAC1 8-bit right-aligned data register
0x14	DAC_DHR12R1	DAC2 12-bit right-aligned data register
0x18	DAC_DHR12L1	DAC2 12-bit left-aligned data register
0x1C	DAC_DHR8R1	DAC2 8-bit right-aligned data register
0x20	DAC_DHR12RD	Dual-DAC 12-bit right-aligned data register
0x24	DAC_DHR12LD	Dual-DAC 12-bit left-aligned data register
0x28	DAC_DHR8RD	Dual-DAC 8-bit right-aligned data register
0x2C	DAC_DOR0	DAC1 output data register
0x30	DAC_DOR1	DAC2 output data register
0x34	DAC_IS	Interrupt status register
0x38	DAC_CLK	DAC clock setting register

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.5.1 Control Register (DAC_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	DAC2_VREF_MODE	R/W	0x0	DAC2 reference voltage selection: 00/01: VREFP_AVDD (VDDA) 10: External input reference voltage V_{VREFP} or provided by the internal VREF module 11: Reserved
29	DMAIE1	R/W	0x0	DAC2 DMA underflow interrupt enable: 0: Interrupt disabled 1: Interrupt enabled
28	DMAEN1	R/W	0x0	DAC2 DMA mode enable: 0: DMA disabled 1: DMA enabled
27:24	MAMP1	R/W	0x0	Selection of mask in DAC2 noise mode / amplitude in triangle wave mode: 0000: No masking of LSFR bit [0] / triangle wave amplitude is 0 0001: No masking of LSFR bit [1:0] / triangle wave amplitude is 3 0010: No masking of LSFR bit [2:0] / triangle wave amplitude is 7 0011: No masking of LSFR bit [3:0] / triangle wave amplitude is 15 0100: No masking of LSFR bit [4:0] / triangle wave amplitude is 31 0101: No masking of LSFR bit [5:0] / triangle wave amplitude is 63 0110: No masking of LSFR bit [6:0] / triangle wave amplitude is 127 0111: No masking of LSFR bit [7:0] / triangle wave amplitude is 255

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1000: No masking of LSFR bit [8:0] / triangle wave amplitude is 511 1001: No masking of LSFR bit [9:0] / triangle wave amplitude is 1023 1001: No masking of LSFR bit [10:0] / triangle wave amplitude is 2047 ≥ 1011: No masking of LSFR bit [11:0] / triangle wave amplitude is 4095
23:22	WAVE1	R/W	0x0	Select DAC2 generation of noise wave/triangle wave: 00: Neither noise wave nor triangle wave generated 01: Noise wave generated 10/11: Triangle wave generated
21:19	TSEL1	R/W	0x0	DAC2 trigger source selection: 000: TIM6 TRGO 001: TIM8 TRGO 010: TIM7 TRGO 011: TIM5 TRGO 100: TIM2 TRGO 101: TIM4 TRGO 110: External interrupt pin 111: Software trigger
18	TEN1	R/W	0x0	DAC2 trigger mode enable: 0: Trigger disabled; data written to DACDHR starts conversion and output after one cycle. 1: Trigger enabled; data written to DACDHR starts conversion and output after three cycles (for hardware trigger) / one cycle (for software trigger).
17	BUF1	R/W	0x0	DAC2 output buffer enable: 0: Disabled 1: Enabled
16	DACEN1	R/W	0	DAC2 enable: 0: Disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Enabled; a power-up delay is required when switching to the enabled state.
15:14	DAC0_VREF_MODE	R/W	0	DAC1 reference voltage selection: 00/01: VREFP_AVDD (VDDA) 10: External input reference voltage V_{REFP} or provided by the internal V_{REF} module 11: Reserved
13	DMAIE0	R/W	0	DAC1 DMA underflow interrupt enable: 0: Disabled 1: Enabled
12	DMAEN0	R/W	0	DAC1 DMA mode enable: 0: Disabled 1: Enabled
11:8	MAMP0	R/W	0	Selection of DAC1 noise mode mask / triangle wave mode amplitude: 0000: No masking of LSFR bit [0] / triangle wave amplitude is 0 0001: No masking of LSFR bit [1:0] / triangle wave amplitude is 3 0010: No masking of LSFR bit [2:0] / triangle wave amplitude is 7 0011: No masking of LSFR bit [3:0] / triangle wave amplitude is 15 0100: No masking of LSFR bit [4:0] / triangle wave amplitude is 31 0101: No masking of LSFR bit [5:0] / triangle wave amplitude is 63 0110: No masking of LSFR bit [6:0] / triangle wave amplitude is 127 0111: No masking of LSFR bit [7:0] / triangle wave amplitude is 255 1000: No masking of LSFR bit [8:0] / triangle wave amplitude is 511 1001: No masking of LSFR bit [9:0] / triangle wave amplitude is 1023

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1001: No masking of LSFR bit [10:0] / triangle wave amplitude is 2047 ≥ 1011: No masking of LSFR bit [11:0] / triangle wave amplitude is 4095
7:6	WAVE0	R/W	0	Select DAC1 generation of noise wave/triangle wave: 00: Neither noise wave nor triangle wave generated 01: Noise wave generated 10/11: Triangle wave generated
5:3	TSEL0	R/W	0	DAC1 trigger source selection: 000: TIM6 TRGO 001: TIM8 TRGO 010: TIM7 TRGO 011: TIM5 TRGO 100: TIM2 TRGO 101: TIM4 TRGO 110: External interrupt pin 111: Software trigger
2	TEN0	R/W	0	DAC1 trigger mode enable: 0: Trigger disabled; data written to DACDHR starts conversion and output after one cycle. 1: Trigger enabled; data written to DACDHR starts conversion and output after three cycles (for hardware trigger) / one cycle (for software trigger).
1	BUF0	R/W	0	DAC1 output buffer enable: 0: Disabled 1: Enabled
0	DACEN0	R/W	0	DAC1 enable: 0: Disabled 1: Enabled; a power-up delay is required when switching to the enabled state.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.5.2 Software Trigger Register (DAC_SWTRG)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:2	RSV	-	-	Reserved
1	SWTRG1	W	0x0	Writing 1 to this bit can generate a software trigger for DAC2, and this bit is cleared automatically.
0	SWTRG0	W	0x0	Writing 1 to this bit can generate a software trigger for DAC1, and this bit is cleared automatically.

37.5.3 DAC1 12-bit Right-aligned Data Register (DAC_DHR12R0)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	DHR0	R/W	0x0	Specify 12 bits of data for DAC1

37.5.4 DAC1 12-bit Left-aligned Data Register (DAC_DHR12L0)

Offset address: 0x0C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:4	DHR0	R/W	0x0	Specify 12 bits of data for DAC1
3:0	RSV	-	-	Reserved

37.5.5 DAC1 8-bit Right-aligned Data Register (DAC_DHR8R0)

Offset address: 0x10

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	DHR0	R/W	0x0	Specify 8 bits of data for DAC1

37.5.6 DAC2 12-bit Right-aligned Data Register (DAC_DHR12R1)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	DHR1	R/W	0x0	Specify 12 bits of data for DAC2

37.5.7 DAC2 12-bit Left-aligned Data Register (DAC_DHR12L1)

Offset address: 0x18

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:4	DHR1	R/W	0x0	Specify 12 bits of data for DAC2
3:0	RSV	-	-	Reserved

37.5.8 DAC2 8-bit Right-aligned Data Register (DAC_DHR8R1)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:8	RSV	-	-	Reserved
7:0	DHR1	R/W	0x0	Specify 8 bits of data for DAC2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.5.9 Dual-DAC 12-bit Right-aligned Data Register (DAC_DHR12RD)

Offset address: 0x20

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:28	RSV	-	-	Reserved
27:16	DHR1	R/W	0x0	Specify 12 bits of data for DAC2
15:12	RSV	-	-	Reserved
11:0	DHR0	R/W	0x0	Specify 12 bits of data for DAC1

37.5.10 Dual-DAC 12-bit Left-aligned Data Register (DAC_DHR12LD)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:20	DHR1	R/W	0x0	Specify 12 bits of data for DAC2
19:16	RSV	-	-	Reserved
15:4	DHR0	R/W	0x0	Specify 12 bits of data for DAC1
3:0	RSV	-	-	Reserved

37.5.11 Dual-DAC 8-bit Right-aligned Data Register (DAC_DHR8RD)

Offset address: 0x28

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:16	RSV	-	-	Reserved
15:8	DHR1	R/W	0x0	Specify 8 bits of data for DAC2

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:0	DHR0	R/W	0x0	Specify 8 bits of data for DAC1

37.5.12 DAC1 Output Data Register (DAC_DOR0)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	DOR0	R	0x0	The 12-bit data being output by DAC1 can be read.

37.5.13 DAC2 Output Data Register (DAC_DOR1)

Offset address: 0x30

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:12	RSV	-	-	Reserved
11:0	DOR1	R	0x0	The 12-bit data being output by DAC2 can be read.

37.5.14 Interrupt Status Register (DAC_IS)

Offset address: 0x34

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:30	RSV	-	-	Reserved
29	DMAIT1	R/W1C	0x0	DMA underflow interrupt status of DAC2, can be cleared by writing 1: 0: No interrupt occurred 1: Interrupt occurred
28:14	RSV	-	-	Reserved

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
13	DMAIT0	R/W1C	0x0	DMA underflow interrupt status of DAC1, can be cleared by writing 1: 0: No interrupt occurred 1: Interrupt occurred
12:0	RSV	-	-	Reserved

37.5.15 DAC Clock Setting Register (DAC_CLK)

Offset address: 0x38

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:9	RSV	-	-	Reserved
8	EXT	R/W	0x0	An additional APBCLK cycle is required for a low level.
7:0	DIV	R/W	0x0	Division factor For example, filling in decimal 82 can divide the frequency from 168 down to 1.00. Note: It is not set by default.

37.6 Operation Procedure

The following takes DAC channel 1 configuration as an example. The configuration for channel 2 is the same.

37.6.1 Direct Output of New Values

1. Configure PA4 as an analog interface and enable the DAC clock in the RCM module.
2. Set DAC_CTRL[2] to 0 to disable triggering.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

3. Set DAC_CTRL[0] to 1 to enable DAC1. After enabling the DAC, a power-up delay is required (add a few milliseconds of delay in software).
4. Write the output value to the corresponding DAC_DHR register, and the respective channel will directly output the the voltage corresponding to the value.

37.6.2 Output Value Update Using Trigger Control

1. Configure PA4 as an analog interface and enable the DAC clock in the RCM module.
2. Set DAC_CTRL[2] to 1 to enable triggering.
3. Configure DAC_CTRL[5:3] to select the trigger source. Here, using software trigger as an example, configure it to 0x7.
4. Set DAC_CTRL[0] to 1 to enable DAC1. After enabling the DAC, a power-up delay is required (add a few milliseconds of delay in software).
5. Write the output value to the corresponding DAC_DHR register. Whenever a trigger event occurs, the respective channel will output the voltage corresponding to the value. In this example, DAC_SWTRG[0] is set to 1 to generate a software trigger.

37.6.3 Output Value Update Using DMA Control

1. Configure PA4 as an analog interface and enable the DAC clock in the RCM module.
2. Use TIM2 as the trigger source and configure TIM2 to generate a TRGO signal upon the update event.
3. Set DAC_CTRL[12] to 1 to enable the DMA mode for DAC.
4. Set DAC_CTRL[2] to 1 to enable triggering.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. Configure DAC_CTRL[5:3] to select the trigger source. In this example, TIM2_TRGO is used as the trigger source, and it is set as 0x4.
6. Set DAC_CTRL[0] to 1 to enable DAC1. After enabling the DAC, a power-up delay is required (add a few milliseconds of delay in software).
7. Configure DMA1. For details on DMA controller configuration, please refer to chapter “11 DMA Controller”.
8. Whenever a trigger event occurs, the DAC will generate a DMA request, and the DMA will write the output value into the corresponding DAC_DHR register, causing the respective channel to output the corresponding voltage value. Here, TIM2 generates a TRGO signal upon each update event to trigger DMA transfer.

37.6.4 Noise Wave Generation

1. Configure PA4 as an analog interface and enable the DAC clock in the RCM module.
2. Set DAC_CTRL[7:6] to 0x1 to generate a noise wave.
3. Configure DAC_CTRL[11:8] to select the output amplitude in noise mode.
4. Set DAC_CTRL[2] to 1 to enable triggering.
5. Configure DAC_CTRL[5:3] to select the trigger source.
6. Set the corresponding DAC_DHR register to specify the initial output value.
7. Set DAC_CTRL[0] to 1 to enable DAC1. After enabling the DAC, a power-up delay is required (add a few milliseconds of delay in software).
8. Whenever a trigger event occurs, the corresponding channel will output a noise wave with certain variations.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

37.6.5 Triangle Wave Generation

1. Configure PA4 as an analog interface and enable the DAC clock in the RCM module.
2. Set DAC_CTRL[7:6] to 0x2 to generate a triangle wave.
3. Configure DAC_CTRL[11:8] to select the output amplitude of the triangle wave.
4. Set DAC_CTRL[2] to 1 to enable triggering.
5. Configure DAC_CTRL[5:3] to select the trigger source.
6. Set the corresponding DAC_DHR register to specify the initial output value.
7. Set DAC_CTRL[0] to 1 to enable DAC1. After enabling the DAC, a power-up delay is required (add a few milliseconds of delay in software).

Whenever a trigger event occurs, the corresponding channel will output a triangle wave with certain variations.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

38 Analog Comparator (ACMP)

38.1 Overview

The device embeds three independent analog comparators (ACMP1, ACMP2, ACMP3).

The ACMP module is used to compare the magnitudes of two input analog voltages and output a high or low logic level to the internal chip based on the comparison result, which can generate an interrupt. When the voltage at the “INP” input pin is higher than that at the “INM” input pin, the comparator outputs a high level; when the voltage at the “INP” input pin is lower than that at the “INM” input pin, the comparator outputs a low level.

38.2 Main Features

- The analog comparator output can generate interrupts, and the interrupt generation method is programmable.
- Up to three independent comparators
- Configurable hysteresis voltage
- Input channels can be selected from I/O ports, DAC output channels, internal reference voltage VBG, VDDH, and VREF
- The system can be woken up from sleep mode via interrupt generation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

38.3 System Block Diagram

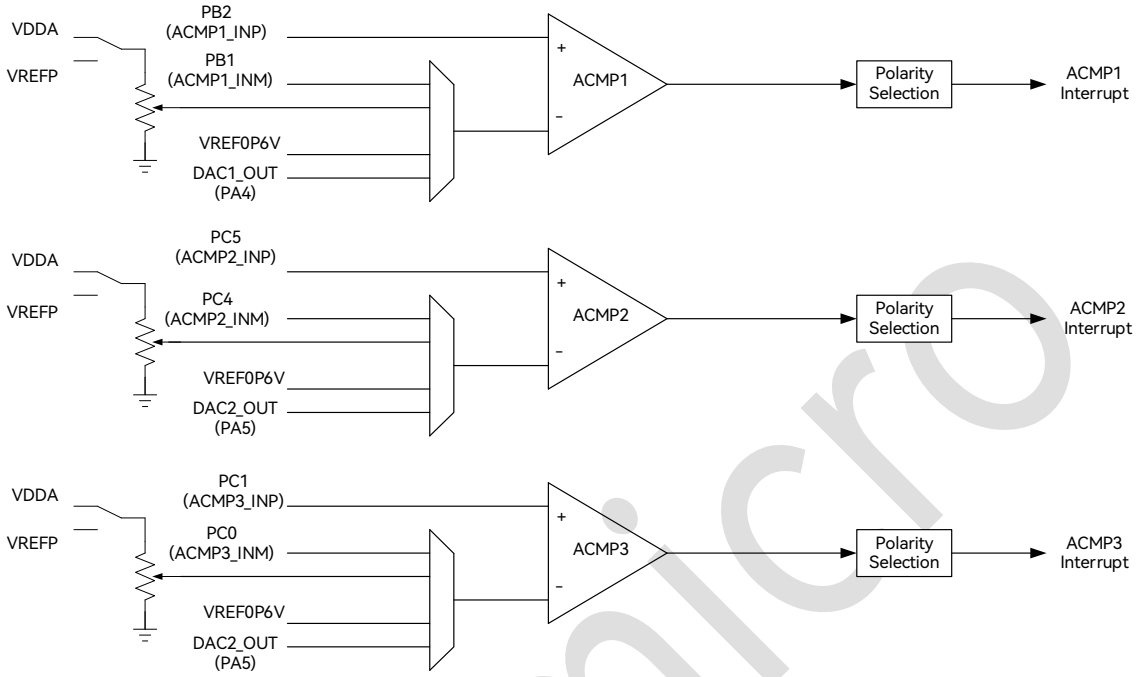

Figure 38-1: ACMP System Block Diagram

38.4 Functional Description

38.4.1 Voltage Comparator

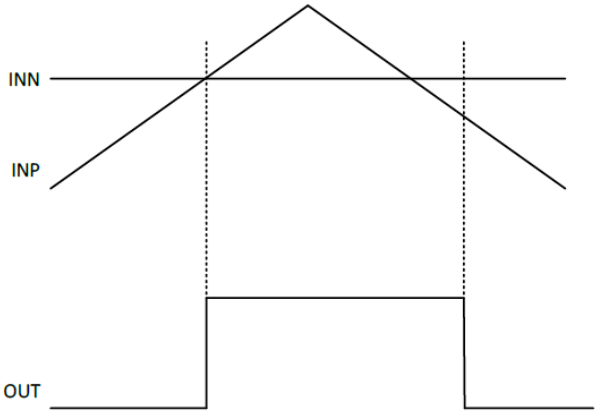

Figure 38-2: Basic Function Diagram of Comparator

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

When configuring the independent comparator function, if the positive input INP voltage is greater than the negative input INN voltage, the comparator outputs high; otherwise, it outputs low.

38.4.2 Comparator Hysteresis Function

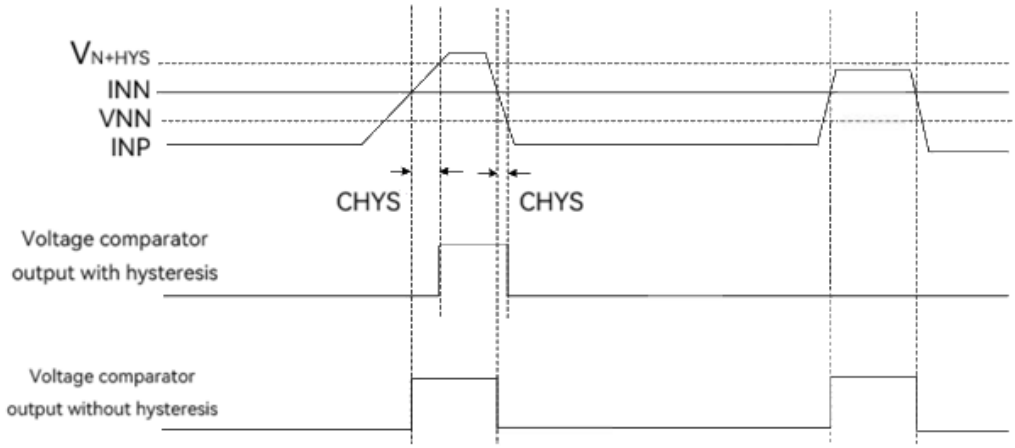

Figure 38-3: Voltage Comparison Function with Hysteresis

The hysteresis function of the comparator is configured by setting ACMP_REG.CHYS. When the hysteresis voltage is set, the comparator outputs high when the positive input INP voltage is greater than the negative input INN voltage plus the hysteresis voltage CHYS. Conversely, the comparator outputs low when the positive input voltage is less than the negative input voltage minus the hysteresis voltage CHYS.

38.5 Register Description

Register base address: 0x4008_0000

The registers are listed below:

Table 38-1: List of ACMP Registers

Offset Address	Register Name	Description
0x00	ACMP_UNLOCK	Write unlock register
0x1C	ACMP_CFG0	Analog comparator 1 register

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Offset Address	Register Name	Description
0x20	ACMP_CFG1	Analog comparator 2 register
0x24	ACMP_CFG2	Analog comparator 3 register

38.5.1 Write Unlock Register (ACMP_UNLOCK)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	UNLOCK	R/W	0	Write unlock register Writing 0xA5A55A5A sets this bit to 1 to unlock the writing to other registers. Writing any other value will relock the register to prevent misoperation.

38.5.2 Analog Comparator 1 Register (ACMP_CFG0)

Offset address: 0x1C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	INTS	R/W1C	0	Interrupt status, can be cleared by writing 1
17:16	INT_CFG	R/W	0	Control interrupt enable: 00: Interrupt disabled 01: Interrupt generated on any edge of OUTPUT 10: Interrupt generated on falling edge of OUTPUT 11: Interrupt generated on rising edge of OUTPUT
15:14	RSV	-	-	Reserved
13	OUTPUT	R	0	Output signal of comparator 1

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
12	CRVINCTRL	R/W	0	Negative input resistor divider input selection: 0: VDDA 1: VREF
11:8	CRVCTRL	R/W	0	Negative input resistor divider ratio: The voltage divider output is (1/6 + CRVCTRL/24) * (V_{DDA} or V_{REF}).
7:6	CNEGSEL	R/W	0	Negative input channel selection: 0: CIN 1: V_{DDA} / V_{REF} 2: VBG (0.6V) 3: DACOUT (DAC1 output)
5:4	CPOSSEL	R/W	0	Positive input channel selection: 0: CIP 0 1: CIP 1 (not supported) 2: CIP 2 (not supported) 3: CIP 3 (not supported) Note: When using this register, this bit shall be fixed at 0 and shall not be modified.
3:2	CHYS	R/W	0	Hysteresis voltage selection: 0: 0 mV 1: 10 mV 2: 20 mV 3: 30 mV
1	CLPM	R/W	0	Low-power mode enable: 0: Low-power mode disabled 1: Low-power mode enabled
0	EN	R/W	0	ACMP1 enable: 0: disabled 1: enabled

38.5.3 Analog Comparator 2 Register (ACMP_CFG1)

Offset address: 0x20

Reset value: 0x0000 0000

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	INTS	R/W1C	0	Interrupt status, can be cleared by writing 1
17:16	INT_CFG	R/W	0	Control interrupt enable: 00: Interrupt disabled 01: Interrupt generated on any edge of OUTPUT 10: Interrupt generated on falling edge of OUTPUT 11: Interrupt generated on rising edge of OUTPUT
15:14	RSV	-	-	Reserved
13	OUTPUT	R	0	Output signal of comparator 2
12	CRVINCTRL	R/W	0	Negative input resistor divider input selection: 0: V_{DDA} 1: V_{REF}
11:8	CRVCTRL	R/W	0	Negative input resistor divider ratio: The voltage divider output is (1/6 + CRVCTRL/24) * (V_{DDA} or V_{REF}).
7:6	CNEGSEL	R/W	0	Negative input channel selection: 0: CIN 1: V_{DDA} / V_{REF} 2: VBG (0.6 V) 3: DACOUT (DAC2 output)
5:4	CPOSSEL	R/W	0	Positive input channel selection: 0: CIP 0 1: CIP 1 (not supported) 2: CIP 2 (not supported) 3: CIP 3 (not supported) Note: When using this register, this bit shall be fixed at 0 and shall not be modified.
3:2	CHYS	R/W	0	Hysteresis voltage selection: 0: 0 mV 1: 10 mV 2: 20 mV

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				3: 30 mV
1	CLPM	R/W	0	Low-power mode enable: 0: Low-power mode disabled 1: Low-power mode enabled
0	EN	R/W	0	ACMP2 enable: 0: Disabled 1: Enabled

38.5.4 Analog Comparator 3 Register (ACMP_CFG2)

Offset address: 0x24

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	INTS	R/W1C	0	Interrupt status, can be cleared by writing 1
17:16	INT_CFG	R/W	0	Control interrupt enable: 00: Interrupt disabled 01: Interrupt generated on any edge of OUTPUT 10: Interrupt generated on falling edge of OUTPUT 11: Interrupt generated on rising edge of OUTPUT
15:14	RSV	-	-	Reserved
13	OUTPUT	R	0	Output signal of comparator 3
12	CRVINCTRL	R/W	0	Negative input resistor divider input selection: 0: V_{DDA} 1: V_{REF}
11:8	CRVCTRL	R/W	0	Negative input resistor divider ratio: The voltage divider output is (1/6 + CRVCTRL/24) * (V_{DDA} or V_{REF}).

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
7:6	CNEGSEL	R/W	0	Negative input channel selection: 0: CIN 1: V_{DDA} / V_{REF} 2: VBG (0.6 V) 3: DACOUT (DAC2 output)
5:4	CPOSSEL	R/W	0	Positive input channel selection: 0: CIP 0 1: CIP 1 (not supported) 2: CIP 2 (not supported) 3: CIP 3 (not supported) Note: When using this register, this bit shall be fixed at 0 and shall not be modified.
3:2	CHYS	R/W	0	Hysteresis voltage selection: 0: 0 mV 1: 10 mV 2: 20 mV 3: 30 mV
1	CLPM	R/W	0	Low-power mode enable: 0: Low-power mode disabled 1: Low-power mode enabled
0	EN	R/W	0	ACMP3 enable: 0: Disabled 1: Enabled

38.6 Operation Procedure

1. Configure the analog GPIO used by the ACMP.
2. Configure the unlock register ACMP_UNLOCK, and write 0xA5A55A5A once to unlock it.
3. Configure the ACMP input signal selection and set whether to enable the hysteresis voltage.
4. Decide whether to configure the ACMP interrupt signal, and clear the interrupt flag in the interrupt service routine.
5. Enable the ACMP module to operate normally.
6. When INP > INN, check whether the comparator output signal is 1.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

39 Operational Amplifier (OPA)

39.1 Overview

This device features two independent operational amplifiers (OPA1 and OPA2) that can be configured as pure operational amplifiers, comparators, unity buffers, or programmable gain amplifiers (PGA). Operational amplifiers are widely used linear integrated circuits, with the most common applications in audio systems.

39.2 Main Features

- Configurable in multiple operating modes: operational amplifier / comparator / unity buffer / PGA
- Operating current: 5 mA
- Operating temperature: -40 ~ 105℃

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

39.3 System Block Diagram

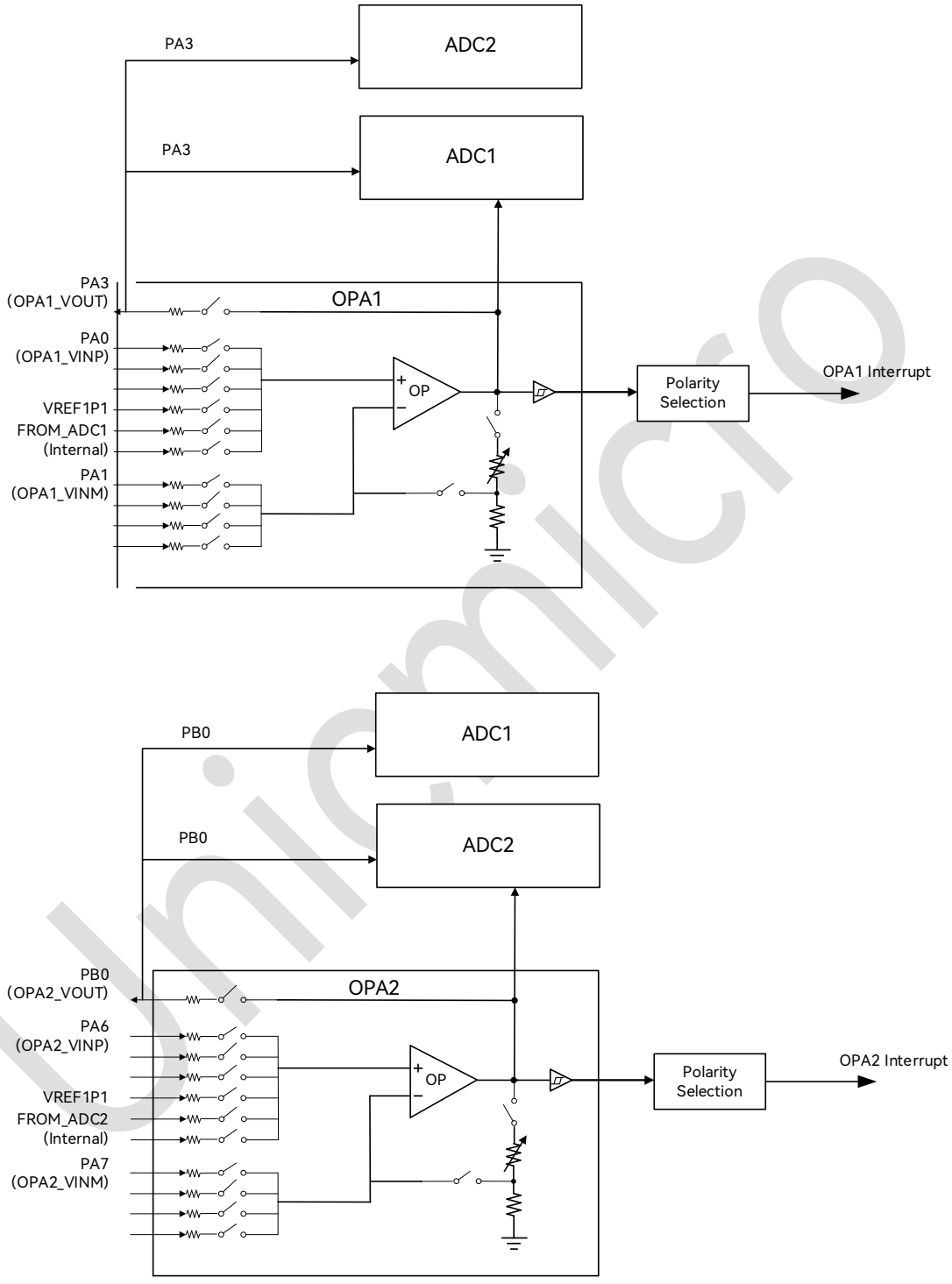

Figure 39-1: OPA System Block Diagram

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

39.4 Register Description

OPA register base address: 0x4008_0000

The registers are listed below:

Table 39-1: List of OPA Registers

Offset Address	Register Name	Description
0x00	OPA_UNLOCK	Write unlock register
0x10	OPA_CFG0	OPA1 setting register
0x14	OPA_CFG1	OPA2 setting register

39.4.1 Write Unlock Register (OPA_UNLOCK)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	UNLOCK	R/W	0	Write unlock register Writing 0xA5A55A5A sets this bit to 1 to unlock the writing to other registers. Writing any other value will relock the register to prevent misoperation.

39.4.2 OPA1 Setting Register (OPA_CFG0)

Offset address: 0x10

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	INT	R/W1C	0	Interrupt status, can be cleared by writing 1
17:16	INT_CFG	R/W	0	Control interrupt enable: 00: Interrupt disabled 01: Interrupt generated on any edge of COMPOUT

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				10: Interrupt generated on falling edge of COMPOUT 11: Interrupt generated on rising edge of COMPOUT
15	RSV	-	-	Reserved
14	COMPOUT	R	0	Output state of OPA as a comparator
13	COMPEN	R/W	0	Comparator function enable: 0: Disabled 1: Enabled
12:10	GAINSEL	R/W	0	Single-ended PGA gain selection signal: 000: 1X 001: 2X 010: 4X 011: 8X 100: 16X 101: 32X 110: 64X 111: Reserved Note: If a gain of 16X/32X/64X is selected, an external feedback resistor is required.
9:7	SELP	R/W	0	OPA positive channel selection signal (SELP[2:0]): 000: Input from pin PA0 001: Reserved 010: Reserved 011: Input from a CORE voltage of 1.1 V 100: Connected from ADC MUX output Others: Reserved
6:4	SELN	R/W	0	OPA negative channel selection signal (SELN[2:0]): 000: Input from pin PA1 001: Analog ground 010: Analog ground 011: Analog ground Others: None of the above (nothing selected)

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	OTPEN	R/W	0	Enable OPA output to IO pin: 0: OPA output not connected to IO pin 1: OPA output connected to IO pin PA3
2	CAPEN	R/W	0	PGA internal feedback capacitor enable signal: 0: Internal feedback capacitor disabled 1: Internal feedback capacitor enabled
1	FBRESEN	R/W	0	PGA internal feedback resistor enable signal: 0: Internal feedback resistor disabled 1: Internal feedback resistor enabled
0	EN	R/W	0	OPA module enable signal: 0: Disabled 1: Enabled

39.4.3 OPA2 Setting Register (OPA_CFG1)

Offset address: 0x14

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:19	RSV	-	-	Reserved
18	INT	R/W1C	0	Interrupt status, can be cleared by writing 1
17:16	INT_CFG	R/W	0	Control interrupt enable: 00: Interrupt disabled 01: Interrupt generated on any edge of COMPOUT 10: Interrupt generated on falling edge of COMPOUT 11: Interrupt generated on rising edge of COMPOUT
15	RSV	-	-	Reserved
14	COMPOUT	R	0	Output state of OPA as a comparator
13	COMPEN	R/W	0	Comparator function enable: 0: Disabled

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				1: Enabled
12:10	GAINSEL	R/W	0	PGA gain selection signal: 000: 1X 001: 2X 010: 4X 011: 8X 100: 16X 101: 32X 110: 64X 111: Reserved Note: If a gain of 16X/32X/64X is selected, an external feedback resistor is required.
9:7	SELP	R/W	0	OPA positive channel selection signal (SELP[2:0]): 000: Input from pin PA6 001: Input from a CORE voltage of 1.1 V 010: Reserved 011: Reserved 100: Connected from ADC MUX output Others: Reserved
6:4	SELN	R/W	0	OPA negative channel selection signal (SELN[2:0]): 000: Input from pin PA7 001: Analog ground 010: Analog ground 011: Analog ground Others: None of the above (nothing selected)
3	OTPEN	R/W	0	Enable OPA output to IO pin: 0: OPA output not connected to IO pin 1: OPA output connected to IO pin (PB0)
2	CAPEN	R/W	0	PGA internal feedback capacitor enable signal: 0: Internal feedback capacitor disabled 1: Internal feedback capacitor enabled
1	FBRESEN	R/W	0	PGA internal feedback resistor enable signal:

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
				0: Internal feedback resistor disabled 1: Internal feedback resistor enabled
0	EN	R/W	0	OPA module enable signal: 0: Disabled 1: Enabled

39.5 Operation Procedure

39.5.1 UNITBUFF Mode

1. Multiplex the OPA-related GPIO for analog function.
2. Configure the unlock register OPA_UNLOCK, and write 0xA5A55A5A once to unlock it.
3. Set OPA_CFGx[1], OPA_CFGx[2] and OPA_CFGx[3] to 1, and set OPA_CFGx[12:10] and OPA_CFGx[13] to 0.
4. Configure the OPA_CFGx register, and set SELN to 0 to select the SELP input signal.
5. Set OPA_CFGx[0] to 1 to enable normal operation of the OPA.

39.5.2 External Feedback Circuit Setup in OPA Mode

1. Multiplex the OPA-related GPIO for analog function.
2. Configure the unlock register OPA_UNLOCK, and write 0xA5A55A5A once to unlock it.
3. Set OPA_CFGx[1] to 0, OPA_CFGx[2] to 0, OPA_CFGx[3] to 1, and OPA_CFGx[13] to 0.
4. Configure the OPA_CFGx register, and set SELN to 0 to select the SELP input signal.
5. Set bit 0 to 1 to enable normal operation of the OPA.
6. Connect the external circuit according to the diagram.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

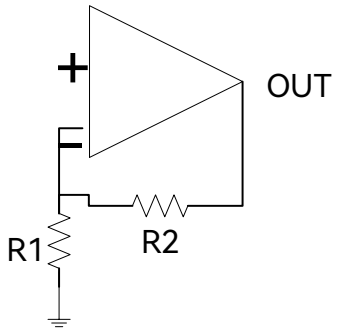

Figure 39-1: External Amplifier Circuit Connection Diagram

Note: Gain = (R1 + R2) / R1

7. Check whether the output voltage is amplified according to the desired gain.

39.5.3 PGA Mode

1. Multiplex the OPA-related GPIO for analog function.
2. Configure the unlock register OPA_UNLOCK, and write 0xA5A55A5A once to unlock it.
3. Set OPA_CFGx[1] to 1, OPA_CFGx[2] to 1, OPA_CFGx[3] to 1, and OPA_CFGx[13] to 0.
4. Select the gain factor by setting OPA_CFGx[12:10].
5. Configure the OPA_CFGx register, and set SELN to 7 to select the SELP input signal.
6. Set OPA_CFGx[0] to 1 to enable normal operation of the OPA.

39.5.4 CMP Mode

1. Multiplex the OPA-related GPIO for analog function.
2. Configure the unlock register OPA_UNLOCK, and write 0xA5A55A5A once to unlock it.
3. Set OPA_CFGx[1] to 0, OPA_CFGx[2] to 0, OPA_CFGx[3] to 0, OPA_CFGx[12:10] to 7, and OPA_CFGx[13] to 1.
4. Configure the OPA_CFGx register to set SELN and SELP input signals.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

5. Configure the interrupt trigger method and decide whether to enable the interrupt [if interrupts are enabled, the interrupt signal must be cleared].

6. Set OPA_CFGx[0] to 1 to enable normal operation of the OPA.

7. Check if OPA_CFGx[14] is set under the specified conditions.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

40 Temperature Sensor (TS)

40.1 Overview

This is a low-power, high-precision temperature sensor that features a built-in 12-bit resolution analog-to-digital converter (ADC) for measuring the ambient temperature (T_A) of the device. The sensor can be placed in power-down mode when not in use.

40.2 Main Features

- Supported temperature range: -40 ~ 105℃
- ADC resolution: 12 bits
- Programmable chopper clock
- Accuracy: ±3℃

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

40.3 Functional Description

40.3.1 Temperature Measurement

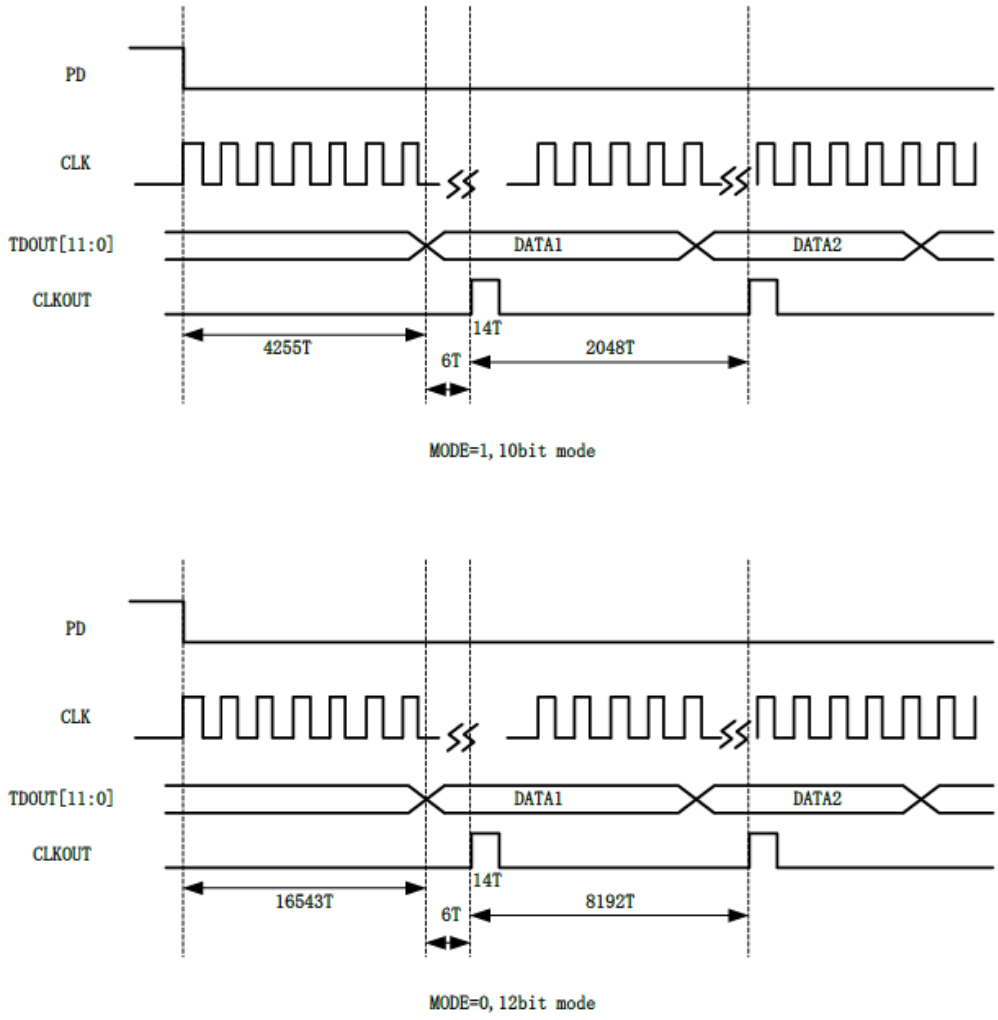

Figure 40-1: Temperature Sensor Conversion Timing Diagram

As shown in Figure 40-1, when the PD signal goes low, the TS starts operating. CLK is the clock for the TS. When MODE = “1”, in high-speed mode, the output data resolution is 10 bits, and the output data rate is f_{CLK} / 2048. When MODE = “0”, in low-speed mode, the output data resolution is 12 bits, and the output data rate is f_{CLK} / 8192.

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

40.4 Register Description

TS register base address: 0x4008_0000

The registers are listed below:

Table 40-1: List of TS Registers

Offset Address	Register Name	Description
0x00	TS_UNLOCK	Write unlock register
0x18	TS_VREFCFG	TSVREF setting register
0x28	TS_CFG	Temperature sensor setting register
0x2C	TS_DATA	Temperature sensor data register

40.4.1 Write Unlock Register (TS_UNLOCK)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:0	UNLOCK	R/W	0	Write unlock register Writing 0xA5A55A5A sets this bit to 1 to unlock the writing to other registers. Writing any other value will relock the register to prevent misoperation.

40.4.2 TSVREF Setting Register (TS_VREFCFG)

Offset address: 0x18

Reset value: 0x0000 0EF1

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
13:4	CHOP_CLK_DIV	R/W	10'hEF	Chopper clock divider setting: Chopper clock frequency = System clock / CHOP_CLK_DIV

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
3	CHOP_CLK_EN	R/W	0	Chopper clock enable signal: 0: Chopper clock disabled 1: Chopper clock enabled
2:1	VREF_SEL	R/W	0	VREF voltage selection: 00: 1.5 V 01: 2.0 V 10: 2.5 V 11: 3.0 V
0	PD	R/W	1	VREF enable signal: 0: Normal operation mode 1: Power-down mode

Note: This register shares a register with VREF. When using both the VREF module and the TS module simultaneously, attention shall be paid to the chopper clock configuration, with a typical setting of 50 kHz.

40.4.3 Temperature Sensor Setting Register (TS_CFG)

Offset address: 0x28

Reset value: 0x0000 0001

Bit	Name	Attribute	Reset Value	Description
31:3	RSV	-	-	Reserved
2	IRQ_EN	R/W	0	An interrupt is triggered when new data is available. Reading TSDATA will clear the interrupt.
1	MODE	R/W	0	Mode selection: 0: Low-speed mode, with an output data resolution of 12 bits and an output data rate of f_{CLK} / 8192 1: High-speed mode, with an output data resolution of 10 bits and an output data rate of f_{CLK} / 2048

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

Bit	Name	Attribute	Reset Value	Description
0	PD	R/W	1	TS enable signal: 0: Normal operation mode 1: Power-down mode

40.4.4 Temperature Sensor Data Register (TS_DATA)

Offset address: 0x2C

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:13	RSV	-	-	Reserved
12	UPDATED	R	0	The first reading of the current data since the last update.
11:0	DATA	R	0	Sensor data

40.5 Operation Procedure

1. Configure the unlock register TS_UNLOCK by writing 0xA5A55A5A once to unlock it.
2. Configure TS_CFG[2] to set the interrupt status and select the interrupt enable switch.
3. Configure TS_CFG[1] to choose the conversion mode.
4. Configure TS_VREFCFG[3] to 1 to enable the chopper clock.
5. Configure TS_VREFCFG[13:4] to set the chopper clock division value, with a typical value of 50 kHz.
6. Wait for TS_DATA[12] to be set to 1, indicating that new data is available.
7. Read TS_DATA[11:0] to obtain the conversion data.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

41 Internal Reference Voltage Source (VREF)

41.1 Overview

The built-in independent voltage reference source (VREF) can output voltages of 1.5 V, 2.0 V, 2.5 V or 3.0 V.

41.2 Register Description

VREF register base address: 0x4008_0000

The registers are listed below:

Table 41-1: List of VREF Registers

Offset Address	Register Name	Description
0x00	VREF_UNLOCK	Write unlock register
0x18	TS_VREFCFG	TSVREF setting register

41.2.1 Write Unlock Register (VREF_UNLOCK)

Offset address: 0x00

Reset value: 0x0000 000

Bit	Name	Attribute	Reset Value	Description
31:0	UNLOCK	R/W	0	**Write unlock register** Writing 0xA5A55A5A sets this bit to 1 to unlock the writing to other registers. Writing any other value will relock the register to prevent misoperation.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

41.2.2 TSVREF Setting Register (TS_VREFCFG)

Offset address: 0x18

Reset value: 0x0000 0EF1

Bit	Name	Attribute	Reset Value	Description
31:14	RSV	R	0	Reserved
13:4	CHOP_CLK_DIV	R/W	10'hEF	Chopper clock divider setting: Chopper clock frequency = System clock / CHOP_CLK_DIV
3	CHOP_CLK_EN	R/W	0	Chopper clock enable signal: 0: Chopper clock disabled 1: Chopper clock enabled
2:1	VREF_SEL	R/W	0	VREF voltage selection: 00: 1.5 V 01: 2.0 V 10: 2.5 V 11: 3.0 V
0	PD	R/W	1	VREF enable signal: 0: Normal operation mode 1: Power-down mode

41.3 Operation Procedure

1. Configure the unlock register VREF_UNLOCK by writing 0xA5A55A5A once to unlock it.
2. Set TS_VREFCFG[3] to enable the chopper clock.
3. Configure TS_VREFCFG[13:4] to set the chopper clock division value.
4. Configure TS_VREFCFG[2:1] to set the VREF output voltage.
5. Configure TS_VREFCFG[0] to enable VREF to operate normally.
6. Check the pin output voltage.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

42 SysTick Timer

42.1 Overview

To support multitasking, the OS needs to perform context switching periodically, which requires hardware resources such as timers to interrupt program execution. When the timer interrupt occurs, the processor will schedule OS tasks in exception handling, and at the same time, it will also carry out OS maintenance. The Cortex-M4 + core is provided with a simple timer called SysTick that is used to generate periodic interrupt requests. SysTick is a 24-bit timer that counts down. When the timer counts down to 0, a programmable value is reloaded and a SysTick exception (exception No. 15) is generated, which causes the execution of SysTick exception handling, a process that is part of the OS.

For systems that do not require an OS, the SysTick timer can also be used for other purposes, such as timing, counting or providing interrupt sources for tasks that require periodic execution. The generation of SysTick exception is controlled, and if exception is prohibited, the SysTick timer can still be used in a polling way, such as checking the current count value or polling the overflow flag.

42.2 Register Description

SysTick register base address: 0xE000_E010

The registers are listed below:

Table 42-1: List of SysTick Registers

Offset Address	Register Name	Description
0x00	SYSTICK_CTRL	SysTick control and status register
0x04	SYSTICK_LOAD	SysTick reload value register
0x08	SYSTICK_VAL	SysTick current valuer register

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

42.2.1 SysTick Control and Status Register (SYSTICK_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:17	RSV	R	0	Reserved
16	COUNTFLAG	R	0	SysTick timer overflow flag: 1: Underflow occurred in SysTick timer 0: No overflow occurred in SysTick timer Reading this register can clear the COUNTFLAG flag.
15:3	RSV	R	0	Reserved
2	CLKSOURCE	R/W	0	SysTick clock source selection: 1: HCLK 0: Invalid
1	TICKINT	R/W	0	SysTick interrupt enable: 1: Enabled 0: Disabled
0	ENABLE	R/W	0	SysTick timer enable: 1: Enabled 0: Disabled

42.2.2 SysTick Reload Value Register (SYSTICK_LOAD)

Offset address: 0x04

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	R	0	Reserved
23:0	RELOAD	R/W	0	SysTick timer reload value

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

42.2.3 SysTick Current Value Register (SYSTICK_VAL)

Offset address: 0x08

Reset value: 0x0000 0000

Bit	Name	Attribute	Reset Value	Description
31:24	RSV	R	0	Reserved
23:0	CURRENT	R/W	0	Read this register to obtain the current count value of SysTick timer; write any value to this register to clear this register and COUNTFLAG.

42.3 Operation Procedure

Since both the reload value and current value of SysTick timer are undefined at reset, to prevent abnormal results, the configuration of SysTick shall follow a certain process:

1. Configure SYSTICK_LOAD[23:0] to set the overflow period of SysTick.
2. Set SYSTICK_CTRL[2] to 1 to select the system clock as the clock source for SysTick.
3. Write any value to SYSTICK_VAL[23:0] to clear the count value and the count flag.
4. If SysTick interrupt is to be used, set SYSTICK_CTRL[1] to 1 to enable the SysTick interrupt.
5. Set SYSTICK_CTRL[0] to 1 to enable SysTick.
6. Query to wait for the timer overflow flag to arrive, then either disable and clear the counter or read SYSTICK_CTRL[16] in the interrupt service routine to clear the overflow flag.

Unicmicro

 Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

43 Debug Support (DBG)

The chip is built around a Cortex®-M4 core which contains hardware extensions for advanced debugging features. The debug extensions allow the core to be stopped either on a given instruction fetch (breakpoint) or data access (watchpoint).

When the core is halted, users can view the internal state of the core and the external state of the system. Upon completion of the query operations, the core and peripherals can be restored to continue executing the corresponding program.

The hardware debugging module of the chip core is available when connected to a debugger (unless disabled).

The following debug interfaces are supported:

- Serial interface (two-wire SWD)
- JTAG debug interface (4-wire or 5-wire JTAG)

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.

44 Revision History

Version	Date	Description
V1.0.0	Feb-01-2026	Initial release.

Unicmicro

Copyright © 2026 Unicmicro (Guangzhou) Co., Ltd.